

Todos os direitos autorais reservados pela **TOTVS S.A.**

Proibida a reprodução total ou parcial, bem como a armazenagem em sistema de recuperação e a transmissão, de qualquer modo ou por qualquer outro meio, seja este eletrônico, mecânico, de fotocópia, de gravação, ou outros, sem prévia autorização por escrito da proprietária.

O desrespeito a essa proibição configura em apropriação indevida dos direitos autorais e patrimoniais da TOTVS.

Conforme artigos 122 e 130 da LEI no. 5.988 de 14 de Dezembro de 1973.

## **TOTVS Business Conector**

### RM – Versão 12.1.10

Habilitar o participante nas funcionalidades do ambiente de forma sistemática e prática, abordando diversas opções oferecidas pelo sistema, tais como: exportação em diversos formatos, importação de diversos formatos, utilização do antigo e novo Conceito para integração, Correspondências, DataServer e WebServices.



## Sumário

1. Objetivo.....	4
2. Introdução.....	4
3. Apresentação.....	6
4. Conceito.....	7
5. Acessando o Conceito do TOTVS Business Conector .....	7
6. Criando um novo conceito (modelo antigo) .....	8
6.1. Identificação.....	8
6.2. Conexão Origem e Conexão Destino .....	9
6.2.1. Conexão Arquivo texto .....	10
6.2.2. Conexão Arquivo XML .....	10
6.2.3. Conexão Banco de dados .....	10
6.2.4. Conexão DataServer (TOTVS RM) .....	10
6.2.5. Conexão MS CRM (Microsoft CRM) .....	11
6.2.6. Conexão WebService.....	11
6.3. Estruturas Origem/Destino.....	15
6.4. Correspondências.....	21
6.5. Execução .....	23
6.6. Execução via agendamento.....	25
7. Transformações.....	27
7.1. Conceito com operação de Lista de decodificação.....	29
7.2. Conceito com operação de Lookup .....	32
7.3. Conceito com operação de Tranformação XSLT .....	34
8. Conexões Entrada X Conexões Destino.....	36
8.1. Conceito com arquivos XML .....	36
8.1.1. Origem banco de dados (SQL) e Destino arquivo XML .....	36
8.1.2. Origem arquivo XML e Destino banco de dados (tabela).....	45
8.2. Conceito com DataServer.....	51
8.2.1. Origem DataServer e Destino arquivo texto.....	51
8.2.2. Origem arquivo XML e Destino DataServer .....	59
9. Conceito (novo conceito ou conceito modelo novo).....	64
9.1. Habilitando o novo conceito .....	65
9.2. Convertendo conceitos antigos para o novo modelo .....	67
9.3. Criando um conceito no novo modelo.....	70
9.4. Identificação (Propriedades e Ligações).....	70
9.5. Parâmetros para execução do conceito.....	71
9.6. Conexão Origem e Conexão Destino .....	73
9.6.1. Conexão/Estrutura de dados origem.....	74
9.6.2. Conexão/Estrutura de dados destino .....	76
9.6.3. Vinculo de parâmetros às fontes de dados (conexões).....	78

9.7. Ligações (Correspondências) .....	79
9.8. Operações (Transformações) .....	81
9.9. Execução .....	85
9.10. Conceito com WebServices .....	88
9.10.1. Origem WebService de terceiros e Data Server .....	88
10. WebServices do TOTVS Business Connect .....	93
10.1. Data Server (wsDataServer) .....	94
10.2. Processos (wsProcess) .....	94
10.3. TOTVS Message - ByYou ESB (wsTOTVSMMessage) .....	95
10.4. TOTVS Educacional (wsEdu) .....	95
10.5. TOTVS Obras e Projetos (wsPrj) .....	95
10.6. TOTVS Gestão de Relacionamento com Cliente (wsCRMAtendimento) .....	95
10.7. Conceitos do TOTVS Business Connect (wsConConceito) .....	97
10.8. Lançamento Financeiro (wsFin) .....	97
10.9. Movimentos no Gestão de Estoque, Compras, Faturamento (wsMovn) .....	97
11. Conclusão .....	98
12. Anexos .....	99
12.1. Como expor os WebServices do TOTVS Business Connect através do Host (recomendado) .....	99
12.2. Como expor os WebServices do TotvsBusinessConnect através do IIS .....	102
12.3. DataServers do TOTVS RM .....	107

## 1. Objetivo

O TOTVS Business Conector é o ETL (Extract | Transform | Load) da TOTVS. É um monitor que permite a integração entre entidades (cadastros/formulários/tabelas/arquivos) distintas para integração do ERP com bases e/ou sistemas legados. Além de permitir o mapeamento em diversos níveis (bases, arquivos, dataservers, webservices), permite a manipulação através de controles de automação para transformar dados entre a origem e o destino.

Este treinamento visa capacitar o participante na compreensão da comunicação e relação necessária entre os sistemas/aplicativos/dados a serem integrados bem como manipular as funcionalidades do TOTVS Business Conector afim de tornar isto possível.

## 2. Introdução

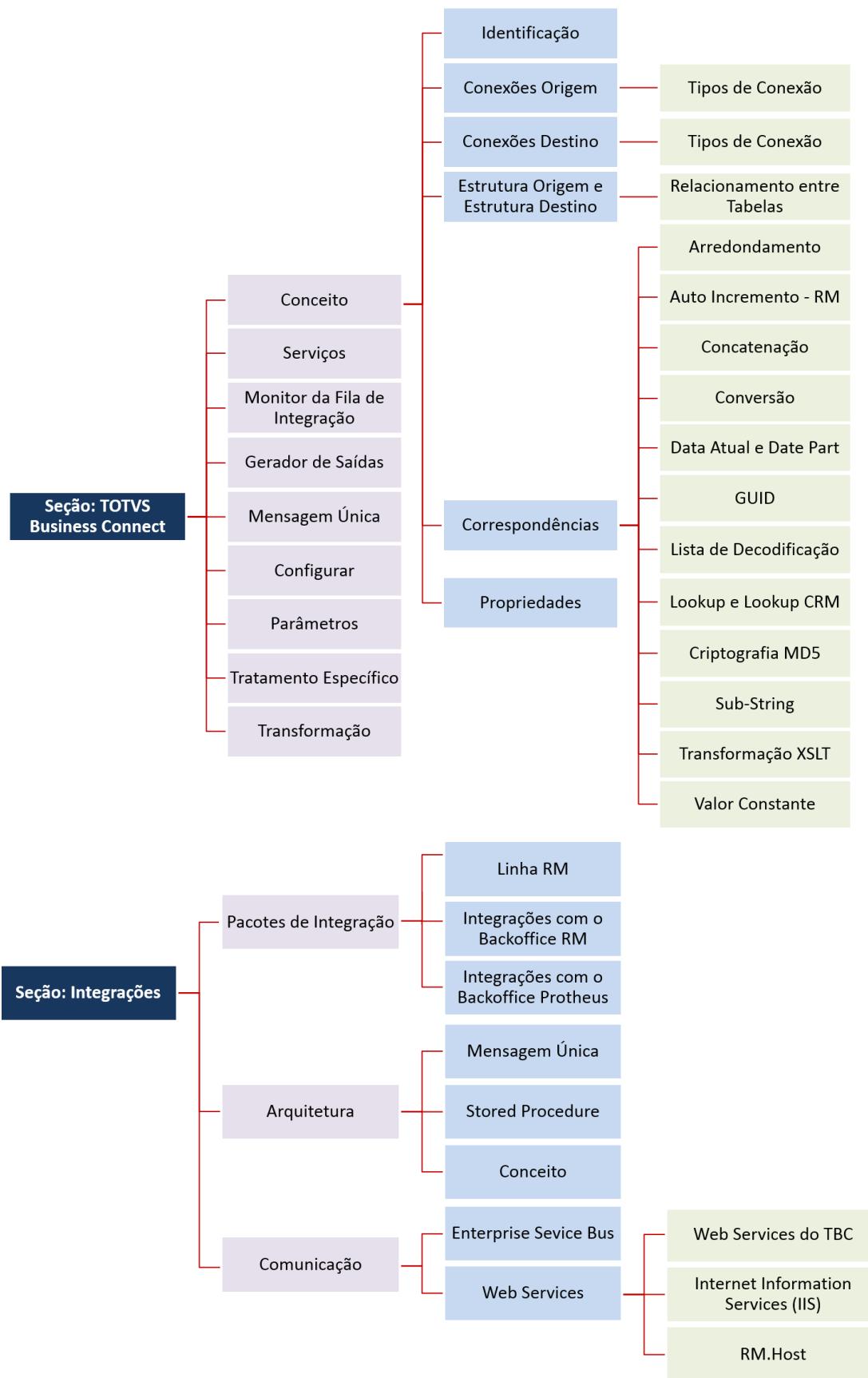
Os sistemas informatizados podem ser povoados a partir de dados de diferentes origens onde a maioria tende a ser bancos de dados relacionais ou flat files (arquivos textos, planilhas, etc.) mas podem existir outros tipos de fontes também. Uma ferramenta de integração precisa ser capaz de se comunicar com bases de dados, ler diversos formatos de arquivos utilizados por toda a organização bem como possibilitar a troca de informações através de diversas interfaces.

Uma ferramenta ETL (Extract | Transform | Load) objetiva trabalhar desde a extração de dados de fontes externas, transformação para atender às necessidades de negócios e carga dos dados nos sistemas (conceitualmente Data Warehouse) então trata-se de uma ferramenta ideal para fazer todo tipo de trabalho de importação, exportação, transformação de dados para outros ambientes de dados e/ou para outras necessidades a serem endereçadas.

Os custos com ferramentas de ETL podem ser expressivos mas não podemos negar que, para um negócio que integrações entre sistemas, BI, resultados on-line, etc., é indispensável uma ferramenta que possibilite a gestão de todo este tráfego de dados. Entretanto avaliar os ganhos de uma ferramenta de ETL antes mesmo de adquiri-la já é algo bem mais complicado.

O TOTVS Business Conector (TBC) é um conjunto de ferramentas definições, recursos/funcionalidades para facilitar as integrações sejam elas dentro do TOTVS RM, sistemas/aplicativos da família TOTVS e, principalmente, com sistemas terceiros sejam eles legados ou fornecidos por outras empresas de software (basta apenas que estes forneçam interfaces de entrada/saída de dados).

# TOTVS Business Conector



Com uma gama de recursos e uma interface visual intuitiva, o TBC consegue tornar o trabalho de integrar soluções em uma tarefa muito simples.

Dentre as várias possibilidades de conexão de dados que a ferramenta disponibiliza, podemos destacar o acesso nativo aos objetos do TOTVS RM, permitindo a gravação de dados que passam pelas mesmas regras de negócio pelas quais passam a informação incluída via cadastro dos produtos TOTVS RM.

### 3. Apresentação

Este material didático foi desenvolvido seguindo uma ordem que facilita o aprendizado. Mostraremos a sequência em que desenvolvemos conceitos (extração, transformação, importação) da maneira mais simples e ao decorrer das lições incrementando recursos mais complexos.

Neste treinamento iremos trabalhar com o recurso chamado “Conceito” integrante do TBC. É através deste recurso que iremos configurar origem, transformação e destino dos dados bem como executá-lo que por fim efetivará a extração, transformação e gravação de dados no destino.

#### Antes de iniciar

Antes de trafegar dados via TBC - Conceito (criar/executar um conceito) temos de nos atentar para um detalhe muito importante: Nunca inicie a montagem de um conceito se você não estiver seguro de que sabe exatamente a origem dos dados, os relacionamentos (mapeamento) entre as informações, as transformações necessárias a aplicar nos dados e o destino dos dados. Esclareça todas as dúvidas antes de iniciar a montagem do conceito, desta forma evitaremos retrabalhos. Após ter conhecimento de todos os detalhes da origem/destino, mapeamentos dos dados e informações a serem trabalhadas, iremos criar o conceito provendo tais informações.

## 4. Conceito

O Conceito é o recurso através do qual é possível importar, transformar e exportar dados de uma conexão origem para uma conexão destino. Também guarda quais campos de cada sistema estão envolvidos na integração, e quais as correspondências (mapeamento, relacionamento) entre os atributos de origem e destino.

Um conceito é uma entidade estruturada que armazena as informações de uma integração específica e, basicamente, é composto por: conexões origem, conexão destino, estrutura origem e estrutura destino, correspondências, transformações (manipular os dados da origem antes de enviá-los ao destino).

Ainda pode-se realizar o agendamento/execução de conceitos, isto é, configura-se um conceito, o grava e então pode-se agendar a sua execução (seja um uma determinada data e/ou através de recorrência). Para isso, selecione os conceitos desejados na visão de Conceito e execute o processo "Processar Conceito". Será aberta uma tela para configurar o agendamento do processo. Após a parametrização, os conceitos estarão agendados e serão executados conforme configurados.

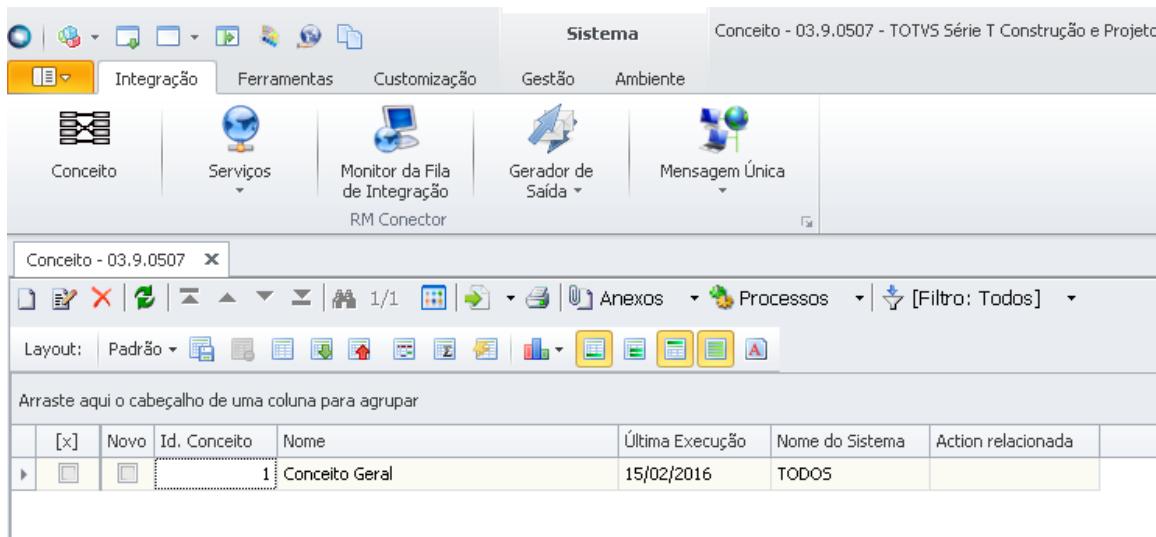
## 5. Acessando o Conceito do TOTVS Business Conector

O Conceito está disponível no módulo de “Integrações” do TOTVS RM e possui permissão controlada pelo perfil de segurança então antes de acessá-lo é necessário conceder permissões desta funcionalidade no perfil do usuário que irão trabalhar com tal recurso.

O acesso ao conceito é realizado através de “Integrações | Integração | Conceito”.



Ao acessá-lo será apresentado uma visão (de acordo com o filtro selecionado) com os conceitos já cadastrados.



## 6. Criando um novo conceito (modelo antigo)

Apresentaremos um exemplo da criação de conceitos montando um conceito simples para extrair dados de uma tabela (banco de dados) e gerar um arquivo com a listagem dos registros desta tabela (arquivo texto).

Para a criação de um novo conceito clique, a partir da visão de conceitos, no ícone de inclusão posicionado no canto superior esquerdo da visão.

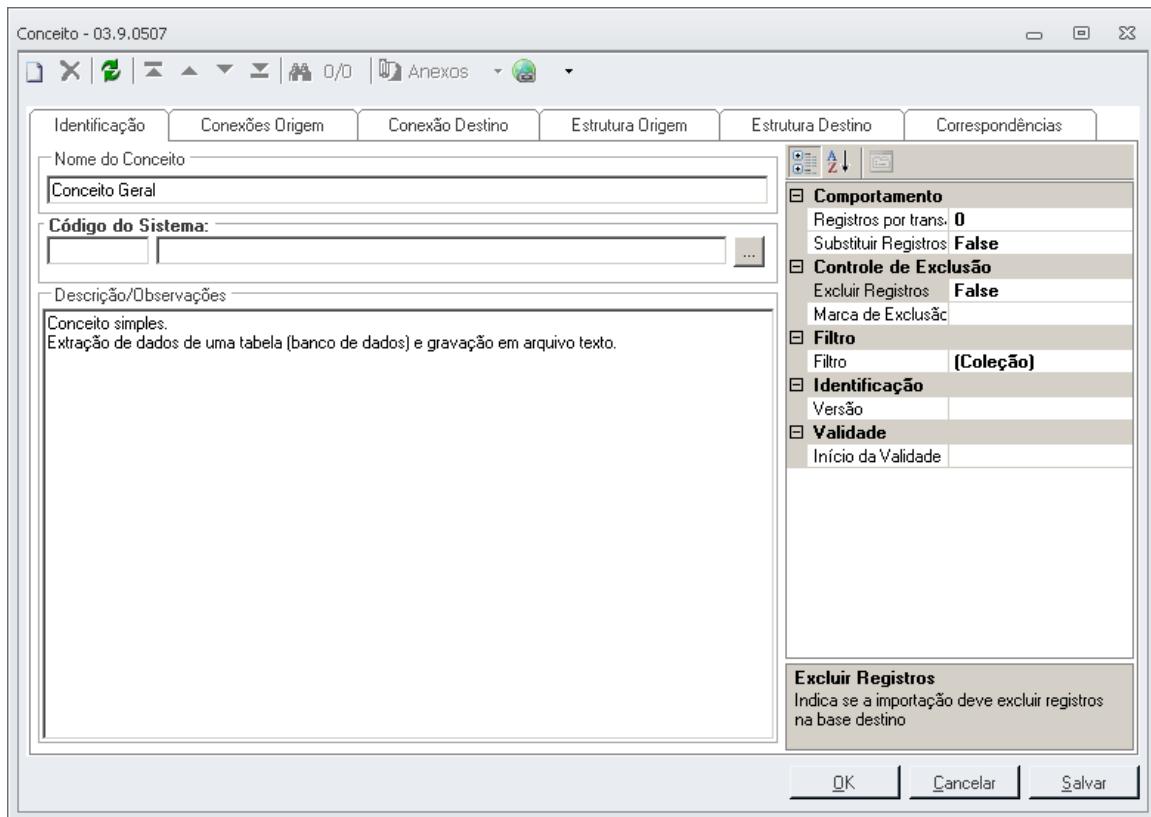
A janela de edição do conceito será apresentada ao usuário para realizar as configurações necessárias. Observe que teremos campos de preenchimento obrigatórios e opcionais.

### 6.1. Identificação

Na aba Identificação temos diversas propriedades/campos com o objetivo de configurar e identificar o conceito que está sendo criado.

- **Nome do Conceito:** nome do conceito que está sendo criado. O nome facilita a identificação do conceito a partir da tela de visão;
- **Código do Sistema:** É possível relacionar um sistema a um determinado conceito, por exemplo, o conceito que trabalha rotinas financeiras pode ser relacionado ao sistema financeiro;
- **Descrição/Observações:** Pode-se informar uma breve descrição sobre o conceito criado;
- **Carregar:** permite importar um conceito a partir de um arquivo no formato \*.icp;
- **Gravar:** permite exportar o conceito para um arquivo no formato \*.icp;
- **Registros por transação:** determina quantos registros da tabela raiz devem ser processados antes de se atualizar o banco destino;
- **Substituir Registros:** indica se a importação deve atualizar registros existentes ao invés de somente incluir novos registros;
- **Excluir Registros:** indica se a importação deve excluir registros na base destino;
- **Marca de Exclusão:** valor a ser pesquisado em campos de controle de exclusão para se apagar um registro (estas configurações terão impacto apenas na Conexão Origem);
- **Filtro:** filtro incremental do processo de importação (esta configuração terá impacto apenas na Conexão Origem);
- **Versão:** versão do conceito;
- **Início da Validade:** data de início da validade do conceito.

Em nosso exemplo inicial informaremos apenas “Nome do Conceito”, “Descrição/Observações”.



## 6.2. Conexão Origem e Conexão Destino

As Conexões Origem/Destino são criadas as conexões com a aplicação origem ou destino. Ao criar uma conexão, é necessário informar o nome e o tipo para depois proceder com o preenchimento de suas características/propriedades. A conexão de origem/destino pode ser de um dos tipos abaixo:

- Arquivo Texto;
- Arquivo XML;
- Banco de Dados;
- Data Server (TOTVS RM);
- MS CRM (Microsoft CRM);
- Web Service (apenas em conexões destino).

### OBSERVAÇÕES:

1. SQLs (sentenças personalizadas) só podem ser utilizadas em conexões de banco de dados na origem.
2. Quando o RM.exe estiver sendo executado utilizando o RM.Host.Service, é necessário realizar algumas configurações para poder fazer uso do recurso conceito com Conexão (Origem ou Destino) do tipo Arquivo XML ou Arquivo Texto.  
A configuração permite carregar os arquivos (de origem ou destino) a partir de outro diretório que não o "C:", como por exemplo, da Área de Trabalho. Para isso é necessário alterar a propriedade de Logon do Host no Serviço do Windows. Acesse "Serviços | RM.Host.Service | Logon" e marque a opção "Permitir que o serviço interaja com a área de trabalho", ou defina uma conta de rede que tenha permissão para acessar a área de trabalho especificada.

A seguir detalharemos as propriedades e significados de cada um dos tipos de conexões.

## 6.2.1. Conexão Arquivo texto

- **Nome:** Nome para identificar a estrutura que está sendo criada. É aconselhável preencher com o nome da tabela, para facilitar a identificação.
- **Continuar execução em caso de erros:** O processamento continua caso algum erro seja apresentado.
- **Tipo de Arquivo:** Arquivo separado por delimitador ou Arquivo de tamanho fixo. Quando o tamanho dos campos for fixo (sem delimitador), é necessário ao informar o conjunto de tabelas e criar os campos, configurar o parâmetro "Tamanho" referente ao campo que está sendo criado.
- **Delimitador:** Caracteres ou conjunto de caracteres que marca o limite entre dois campos. Caso o caractere seja de tabulação, deve se digitar 'TAB'.
- **Arquivo:** Nome do arquivo que deve alimentar a conexão.

## 6.2.2. Conexão Arquivo XML

- **Texto XSD:** Informe o texto schema no formato XSD. Ao clicar no lookup para informar o texto XSD é aberta uma visão na qual é possível digitar o texto schema, ou abrir o arquivo XSD para que o texto seja carregado.
- **Arquivo XML:** Informe o caminho completo e o nome do arquivo XML.
- **Tabela Principal:** Selecione a tabela principal, apenas para conexão de origem.
- **Nó Raiz:** Caso a consulta de origem tenha como retorno múltiplo registros, informe neste campo o nó raiz.

## 6.2.3. Conexão Banco de dados

- **Nome:** Nome para identificar a estrutura de banco de dados que está sendo criada.
- **Flag's:**
  - **Permitir atualização dos registros:** Atualiza registros se necessário.
  - **Continuar execução em caso de erros:** Permite que o processo continue sendo executado em caso de erros.
- **Aba Conexão**
  - **Provedor:** Informe o provedor de acesso ao banco de dados.
  - **Suporta Transações:** Informe se o banco de dados suporta transações.
  - **Testar:** Testa se as informações do banco de dados estão corretas.
  - **Conexão:** Informe os dados do banco de dados e o servidor.
- **Aba Tabelas**
  - Pode ser utilizada uma tabela ou uma sentença SQL.
  - Quando for uma sentença, O nome deve sempre começar com 'qry\_ '.
- **Aba Parâmetros**
  - **Sem Filtro:** Não realiza nenhuma seleção prévia dos registros.
  - **Parâmetro:** Utiliza o parâmetro para realizar seleção dos registros.
  - **Fixo:** Utiliza filtro fixo para selecionar os registros.

## 6.2.4. Conexão DataServer (TOTVS RM)

- **Nome:** Nome para identificar a estrutura do Data Server que está sendo criada.
- **Flag's:**
  - **Permitir atualização dos registros:** Atualiza registros se necessário.

- **Continuar execução em caso de erros:** Permite que o processo continue sendo executado em caso de erros.
- **Aba Conexão**
  - **Servidor:** Informe o nome do DataServer a ser utilizado pela conexão.
  - **Owner Data:** Informe o owner data.
  - **Usar conexão padrão:** utiliza a conexão em ativo.
  - **Alias:** Informe o alias do CORPORE que deve representar o processo de importação. Este Alias já deve ser previamente cadastro (no Alias Manager ou no módulo de Globais).
  - **Usuário:** Informe nome do usuário do CORPORE que deve representar o processo de importação.
  - **Senha:** Informe a senha de autenticação ao CORPORE.
  - **Usar contexto padrão:** Informe se o Data Server irá utilizar a conexão corrente, atribuindo o valor "True". Caso contrário (valor igual a "False"), os campos Usuário CORPORE, Senha do CORPORE e Alias CORPORE devem ser preenchidos.
  - **Contexto:** Permite selecionar manualmente a Chapa do Funcionário, Coligada, Filial, Código Externo, Sistema, Unidade, Usuário, Usuário de Serviço, Data, Exercício Fiscal entre outros.
- **Aba Parâmetros**
  - **Sem Filtro:** Não realiza nenhuma seleção prévia dos registros.
  - **Parâmetro:** Utiliza o parâmetro para realizar seleção dos registros.
  - **Fixo:** Utiliza filtro fixo para selecionar os registros.

## 6.2.5. Conexão MS CRM (Microsoft CRM)

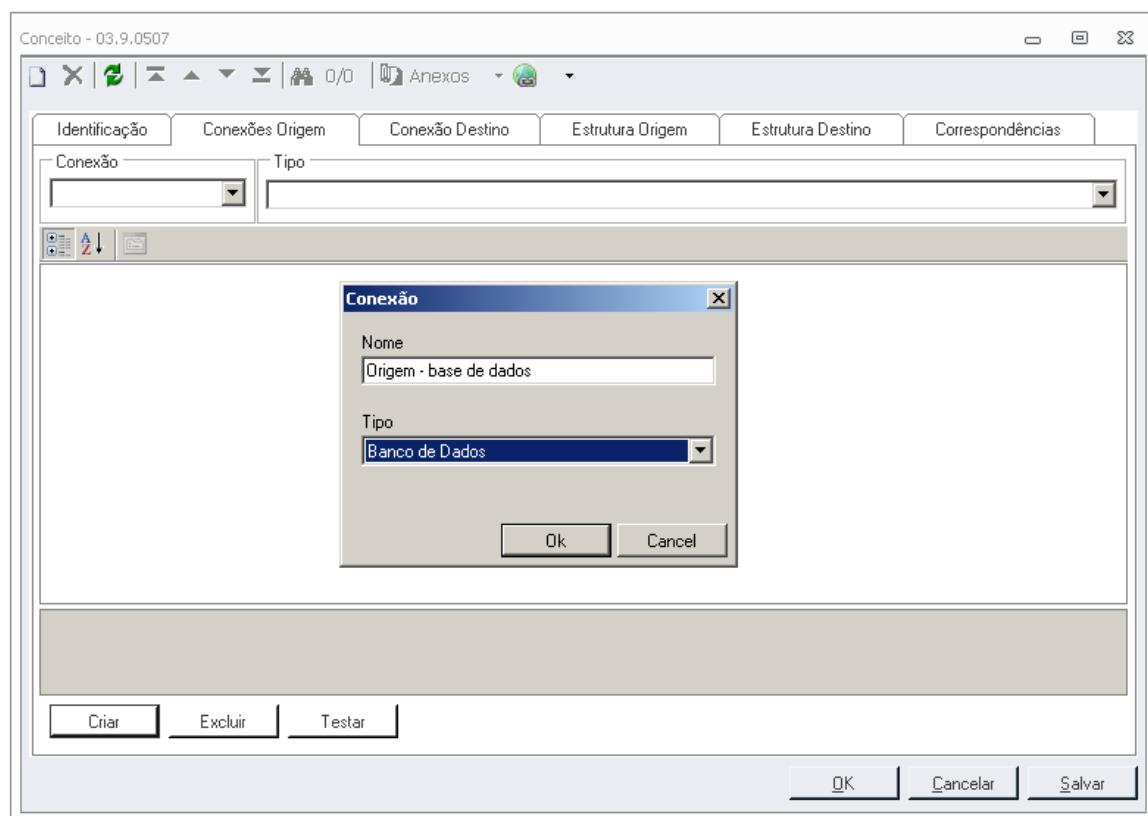
- **Server URL:** Informe o endereço da conexão, por exemplo: `http://crmserver`
- **Domínio:** Informe o nome do domínio.
- **Usuário:** Informe o nome do usuário do CRM.
- **Senha:** Informe a senha do usuário CRM.
- **Organização:** Informe o nome da organização para integração.
- **Entidade CRM:** Informe o nome da entidade CRM.
- **ID CRM:** Informe o nome do identificador de registro da entidade do CRM.

## 6.2.6. Conexão WebService

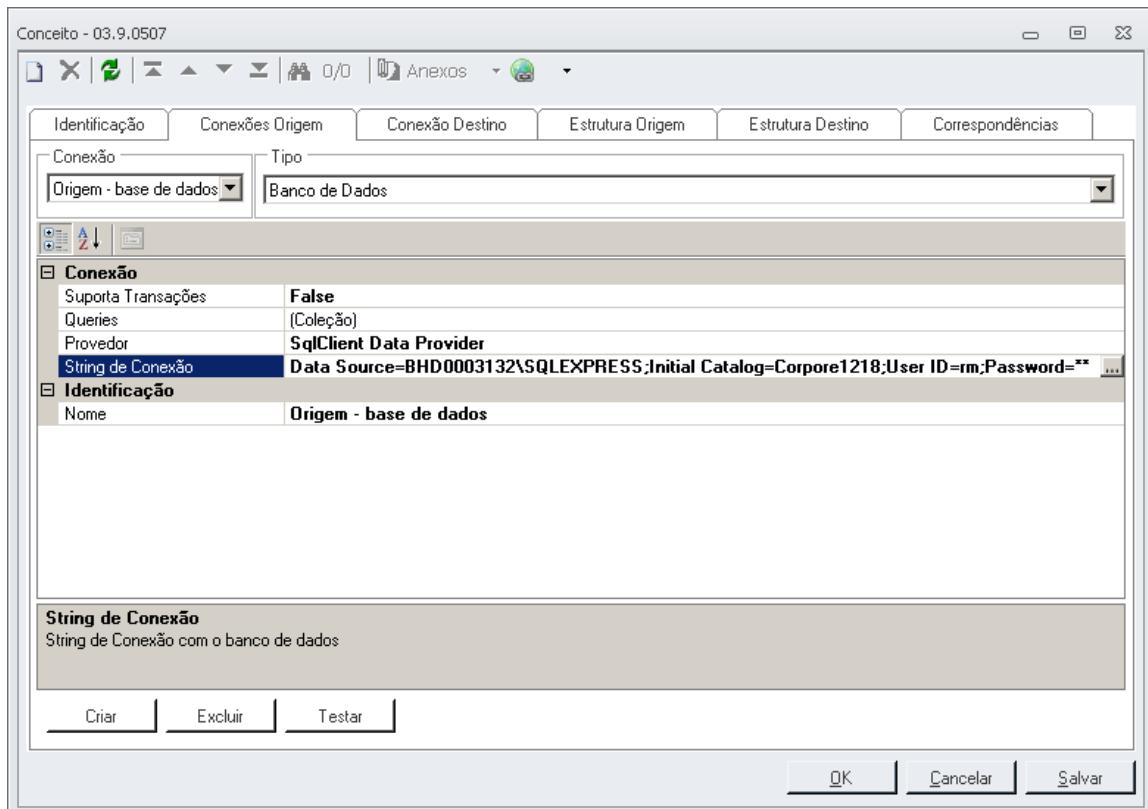
- **Nome:** Nome para identificar a estrutura Web Service que está sendo criada.
- **Flag's:**
  - **Continuar execução em caso de erros:** O processamento continua caso algum erro seja apresentado.
- **Aba Conexão:**
  - **Endereço WSDL:** Informe o endereço do Web Service.  
Exemplo: <http://localhost/ToTvcBusinessConnect/wsProcess.asmx?wsdl>
  - **Protocolo:** Selecione o tipo de protocolo para autenticação.
  - **Usar autenticação padrão:** Informe o nome do servidor do proxy caso seja necessário para acessar o Web Service.
  - **Usuário:** Informe o usuário de domínio caso seja necessário autenticar o Web Service.
  - **Senha:** Informe a senha de domínio caso seja necessário autenticar o Web Service.
  - **Domínio:** Informe o nome do domínio caso seja necessário para autenticar o Web Service.
  - **Usar proxy padrão:** informe se será utilizado o proxy padrão ou não.
  - **Proxy:** Informe o nome de servidor do proxy caso seja necessário acessar o Web Service.
  - **Usar autenticação do proxy padrão:** informe se será utilizado autenticação padrão ou não.

- **Usuário:** Informe o usuário do proxy caso seja necessário acessar o Web Service.
  - **Senha:** Informe a senha do proxy caso seja necessário acessar o Web Service.
  - **Domínio:** Informe o nome do domínio do proxy caso seja necessário acessar o Web Service.
  - **Método:** Informe o método do Web Service, caso seja necessário.
- **Aba Parâmetros**
    - **Sem Filtro:** Não realiza nenhuma seleção prévia dos registros.
    - **Parâmetro:** Utiliza o parâmetro para realizar seleção dos registros.
    - **Fixo:** Utiliza filtro fixo para selecionar os registros.

Na aba Conexões Origem, crie uma conexão (botão “Criar” posicionado no canto inferior esquerdo da interface de edição do conceito) informando seu nome e escolhendo o tipo. A Conexão Origem pode ser dos tipos: Arquivo Texto, Arquivo XML, Banco de Dados, Data Server, MS CRM. Em nosso exemplo utilizaremos o tipo “Banco de dados”.



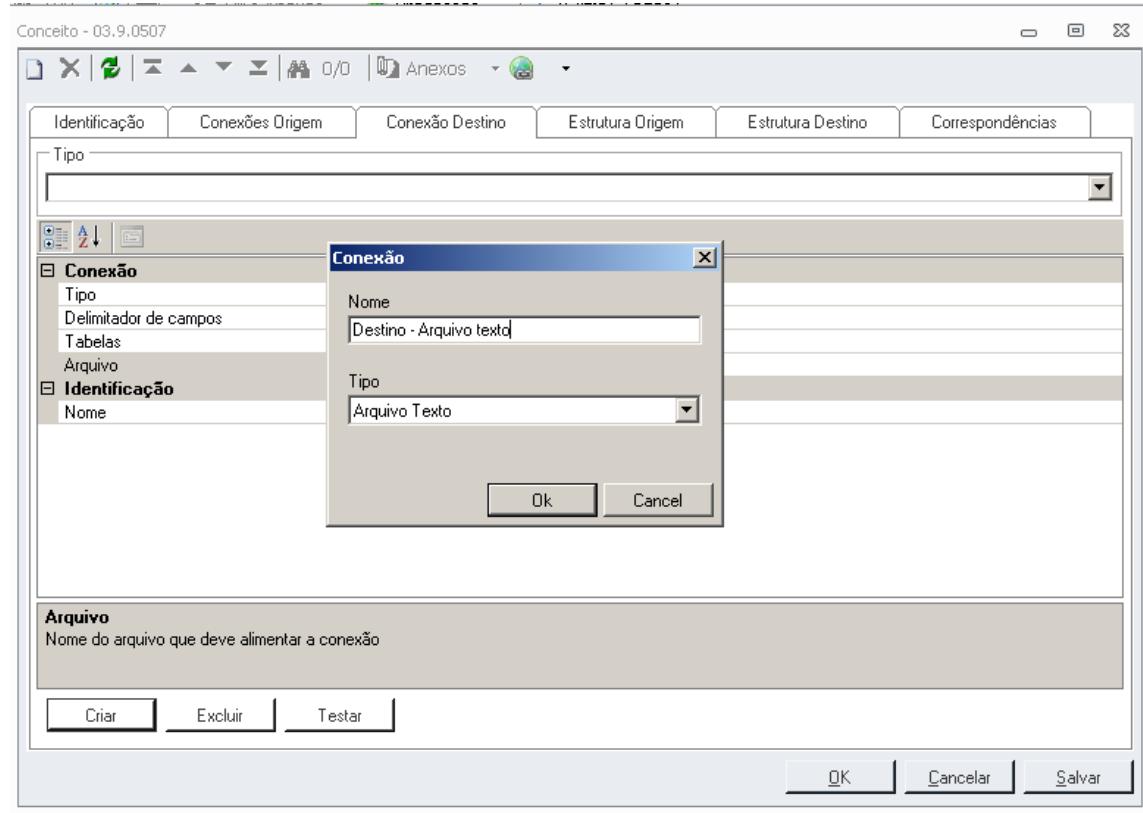
Em seguida à definição do nome e tipo (após clicar em OK), ainda na aba Conexões Origem, serão apresentadas as propriedades/parametrizações da conexão de acordo com o tipo selecionado para que seja configurado a conexão de origem. Considerando que a origem de dados será um banco de dados, devemos configurar o provedor (através de uma lista de provedores) e montar a string de conexão (como facilitador pode-se clicar no botão “...” [reticências] que conduzirá o usuário ao uma interface com as propriedades de strings de conexão).



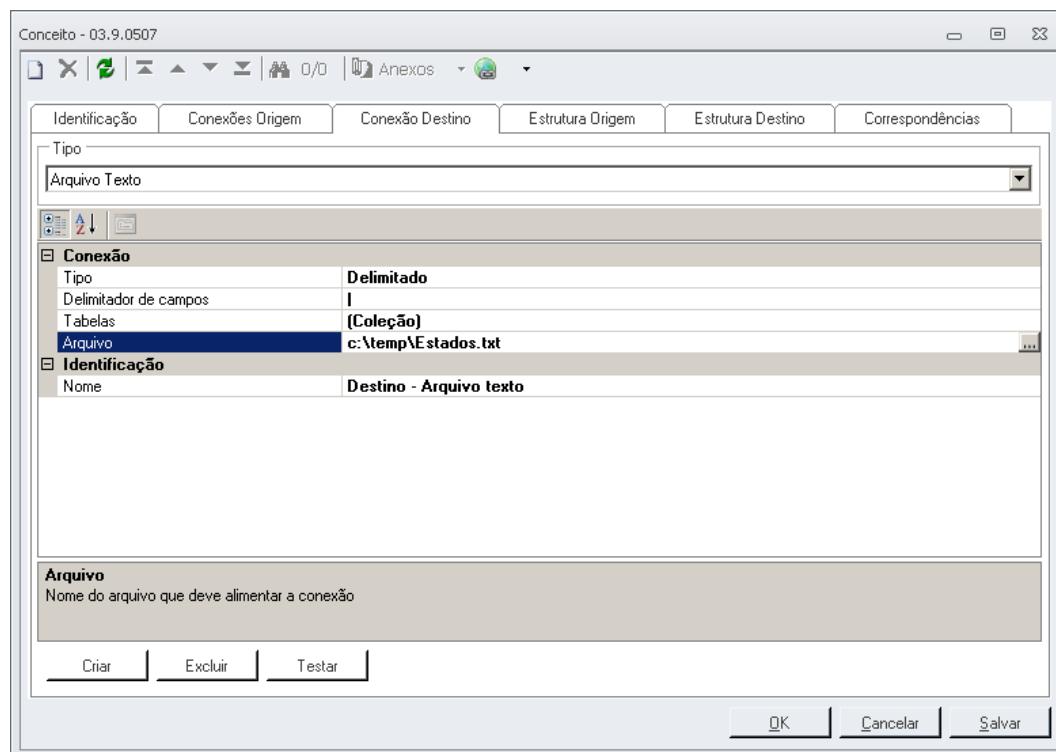
Uma vez definida a conexão de dados de origem o próximo passo é definir a conexão destino que irá receber os dados enviados pela de origem. Para tal acesse a aba Conexão Destino.

A configuração da conexão de destino funciona da mesma forma que na conexão de origem. É necessário criar uma conexão, informar seu nome/tipo e então irão surgir as propriedades/parâmetros referentes ao tipo escolhido para que os mesmos sejam configurados (para a conexão destino, além dos tipos mencionados na Conexão Origem, também está disponível o tipo WebService).

Em nosso exemplo, em que utilizaremos como destino um arquivo texto, devemos escolher como tipo a opção “Arquivo texto” e clicar no botão OK.



Entre os parâmetros informe para o atributo “Tipo” o valor “Delimitado”, para o atributo “Delimitador de campos” o valor “|” (pipe ou outro a sua escolha) e para o atributo “Arquivo” o caminho e nome do arquivo destino a ser gerado.



### 6.3. Estruturas Origem/Destino

Como já definimos/parametrizamos as conexões origem e destino o próximo passo é definir quais informações serão recuperadas/extraidas e para onde vamos exportá-las. Estas definições são realizadas através de estruturas origem e destino representadas pelas abas Estrutura Origem e Estrutura Destino respectivamente.

Nas abas Estrutura Origem e Estrutura Destino estão disponíveis “Tabelas” das conexões (Origem e Destino) definidas nas abas anteriores. Elas deverão ser indicadas, bem como as propriedades de cada campo.

A visão da Estrutura (Origem e Destino) é dividida em:

- **Tabelas Disponíveis:**  
Todas as tabelas definidas na Conexão deverão ser apresentadas neste campo.
- **Tabelas de Importação:**  
As tabelas selecionadas no campo Tabelas Disponíveis e que forem adicionadas à estrutura, podem ser visualizadas no campo Tabelas de Importação.

Estão disponíveis os botões:

- **Adicionar:** Para adicionar uma tabela à estrutura, selecione a tabela desejada no campo Tabelas Disponíveis, e ao clicar no botão Adicionar será aberta uma janela onde é possível definir quais os campos da tabela serão adicionados
- **Remover:** Remove um campo da tabela ou a tabela que foi adicionada ao campo Tabelas de Importação, de acordo com a seleção.
- **Limpar:** Limpa todas as Tabelas de Importação incluídas, deixando em branco para que uma nova seja criada.
- **Carregar:** Permite importar uma Estrutura a partir de um arquivo no formato \*.tst.
- **Gravar:** Permite exportar uma Estrutura para um arquivo no formato \*.tst.
- **Atualizar:** Quando alguma alteração for feita via banco e um novo campo for incluído pra tabela já utilizada, deve-se clicar no ícone Atualizar para que seja carregado esse novo campo nas opções.

- **Campos de Configurações (Tabelas e Campos)**

- **Tabelas:**  
Ao selecionar uma tabela, no quadro Tabelas de Importação, na parte inferior da visão são mostrados os campos para configuração da tabela selecionada.

Alguns campos aparecem para todos os tipos de conexão:

- **Filtro Estático:** Filtro da tabela para importação.
- **Detalhes:** Tabelas selecionadas com detalhes da tabela ativa.
- **Mestre:** Tabela mestre da tabela ativa.
- **Alias:** Alias atribuído à tabela para o processo de importação.
- **Campos:** Campos selecionados da tabela.
- **Conexão:** Campo somente leitura, que informa a conexão de origem da tabela.
- **Nome:** Nome da tabela na fonte de dados. O nome deve sempre começar com "qry\_".
- **Tipo da Tabela:** Campo somente leitura que informa se a tabela é fonte ou origem de dados na importação.

Quando a Conexão criada (Origem ou Destino) for do tipo Banco de Dados, além dos campos citados, também está disponível nos campos da Estrutura (Origem ou Destino) o campo:

- **SQL:** Informe a sentença SQL.

- **Campos:**

Ao selecionar um campo da tabela em Tabelas de Importação, são apresentados na parte inferior da visão as opções para configuração do campo.

As opções são iguais para todos os campos, independentemente do tipo da conexão. São elas:

- **Entrada:** Ponto de entrada da operação ou campo.
- **Controle de Exclusão:** Indica se o campo deve ser utilizado para controle de exclusão. Este parâmetro deve trabalhar em conjunto com o parâmetro Controle de Exclusão da aba Identificação. Ao informar no parâmetro da aba Identificação qual campo será utilizado como controle, é importante na aba Estrutura selecionar o campo informado e configurar este parâmetro como True. Caso contrário o conceito que utilizar este parâmetro não irá funcionar corretamente.
- **Auto Incremento:** Informe se o campo é um campo de auto incremento gerenciado pelo banco.
- **Chave Primária:** Informe se o campo é parte da chave primária da tabela.
- **Obrigatório:** Informe se o campo é obrigatório ou se pode receber valores nulos.
- **Somente Leitura:** Informe se o campo permite ou não gravação de dados.
- **Valor default:** Informe o valor assumido pelo campo quando nenhum outro valor for atribuído.
- **Alias:** Informe o alias atribuído ao campo no processo de importação.
- **Descrição:** Informe notas ou observações acerca do campo.
- **Nome:** Informe o nome do campo na fonte de dados.
- **Tabela:** Campo somente leitura, onde é possível visualizar o nome da tabela à qual o campo pertence.
- **Campo PK Referência:** Determina o nome do campo PK referenciado por este campo.
- **Campo Referência:** Determina o nome do campo referenciado por este campo.
- **Entidade Referência:** Determina o nome da entidade referenciada por este campo.

#### ■ Relacionamento entre Tabelas

Nas Estruturas Origem e Destino é possível adicionar mais de uma tabela para montar as estruturas das conexões criadas.

A primeira tabela adicionada é considerada como a tabela principal, e será reconhecida normalmente no momento de realizar as correspondências, diferente das outras tabelas que não serão apresentadas. Para que as demais tabelas sejam identificadas é necessário realizar o relacionamento entre as mesmas com a tabela principal.

Ao clicar no lookup Detalhes, uma visão é aberta para que se possa realizar este relacionamento. No lado esquerdo da visão são mostrados os relacionamentos existentes, e abaixo estão disponíveis os botões Add e Remove, para adicionar um relacionamento ou remover o relacionamento selecionado, respectivamente.

No lado direito estão disponíveis os campos para configurar o relacionamento. Os campos são:

- Relacionamento
  - **Campos:**  
Vincular o campo da tabela com o campo da tabela mestre.

Ao clicar no lookup Campos para vincular o campo da tabela com o da tabela mestre, uma visão é aberta para informar os mesmos. No lado esquerdo da visão são mostrados os campos que já foram criadas as correspondências.

Abaixo estão disponíveis os botões Add e Remove, para adicionar um campo ou remover o campo selecionado, respectivamente. Do lado direito estão disponíveis os campos:

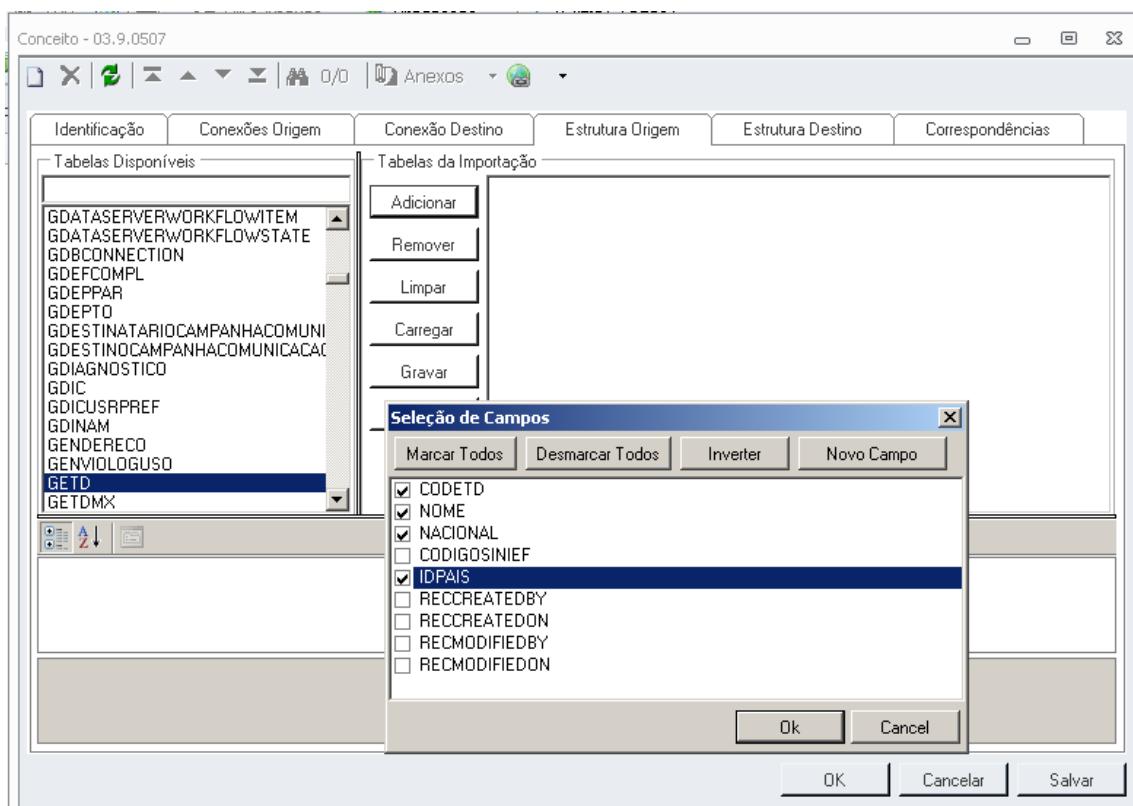
- **Detalhe:** Campo da tabela que deverá fazer correspondência com o campo da tabela mestre.
- **Mestre:** Campo da tabela mestre.
- **Cardinalidade:** Informe se o relacionamento será de um para muitos, ou de um para um.

- Tabela

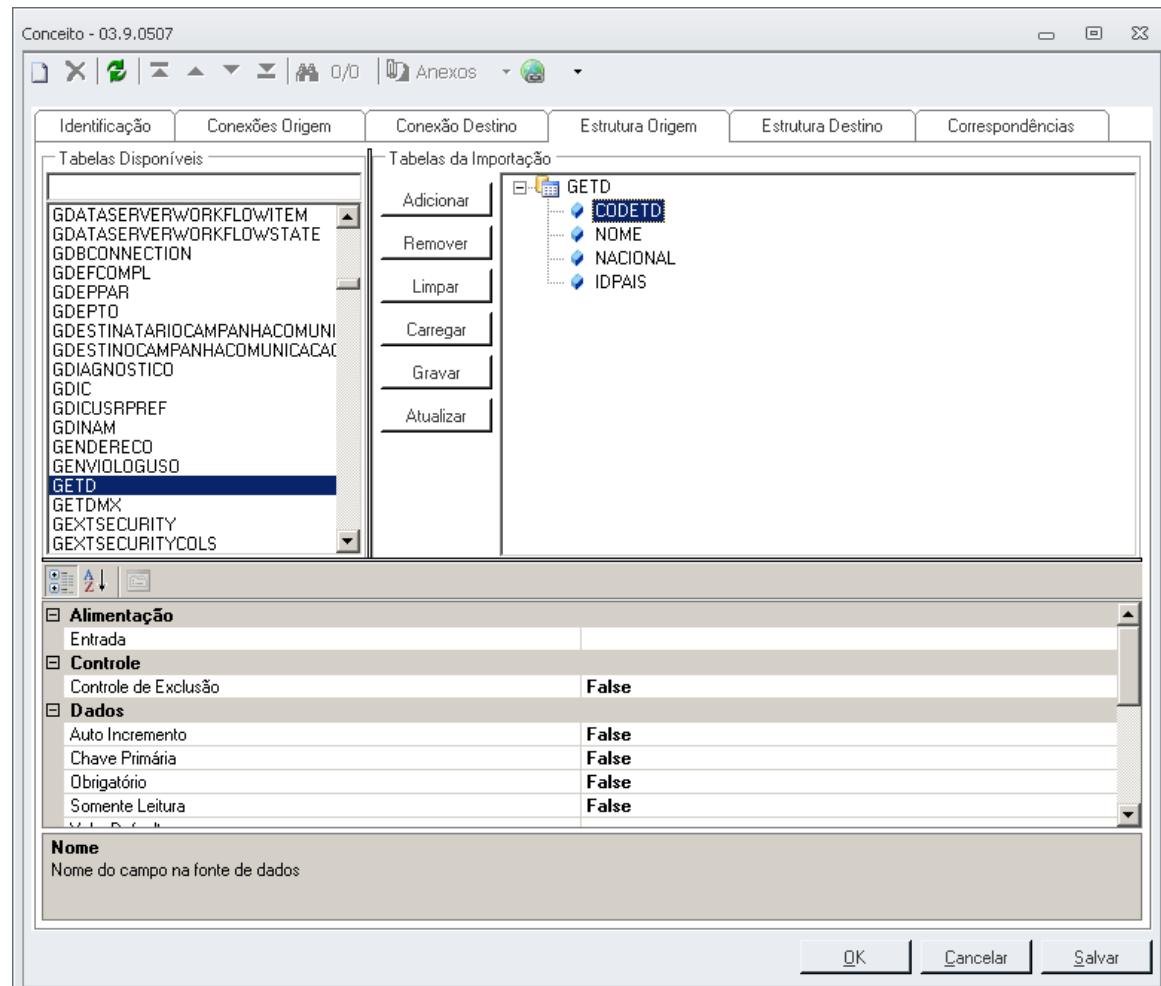
- **Detalhe:** Informe a tabela que deverá ter um relacionamento com a tabela mestre.
- **Nome:** Campo do tipo somente leitura, onde consta o nome da tabela mestre do relacionamento que está sendo criado
- **Nota:** Para que a configuração possa ser realizada corretamente, é necessário primeiro informar a tabela no campo Detalhe, para depois informar os campos e a cardinalidade em Relacionamento.

*OBSERVAÇÃO: Ao criar mais de uma Conexão Origem, pode-se apontar na aba Estrutura Origem os recursos da segunda conexão, acrescentando as tabelas na estrutura (<http://TOTVSconnect.blogspot.com.br/2011/07/como-fazer-conceito-com-relacionamento.html>).*

Seguindo a ideia de nosso exemplo, a origem será dada pelos registros da tabela de estados (GETD) do banco de dados que configuramos na Conexão Origem. Ao acessar a aba Estrutura Origem será apresentado, do lado esquerdo, a lista de tabelas de tal banco de dados. Devemos selecionar a tabela que utilizaremos como fonte de dados e clicar no botão “Adicionar”. Será apresentado a lista de colunas de tal tabela onde devemos marcar as colunas das quais desejamos exportar seus valores.



Ao realizar esta ação tais informações são transportadas para a tela anterior no campo “Tabelas da Importação” e apresentada de forma estruturada (árvore). Observe que na parte debaixo da tela são apresentadas as propriedades de cada coluna (não detalharemos modificações destas propriedades neste momento pois nosso exemplo é inicial, básico, apenas para contextualizar o funcionamento de um conceito).

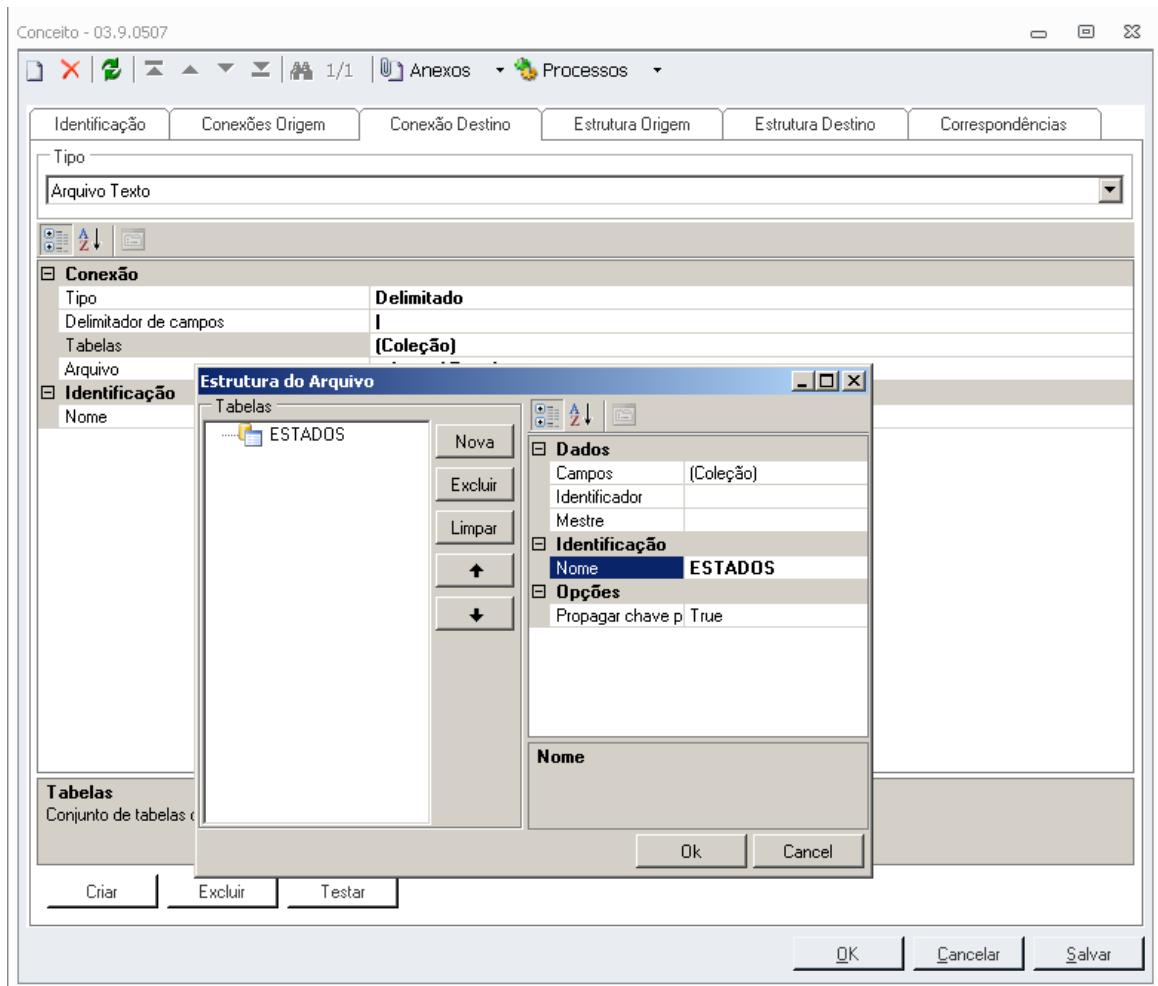


**OBSERVAÇÃO:** a explicação a seguir poderia ter sido apresentada no momento da configuração da conexão destino mas será apresentada propositalmente neste passo pois já temos a definição/estrutura dos dados da origem a serem exportados facilitando assim o entendimento.

Como iremos exportar, da origem, os campos CODETD, NOME, NACIONAL e IDPAIS precisamos ter, no destino, uma estrutura similar com estes campos para receber tais dados.

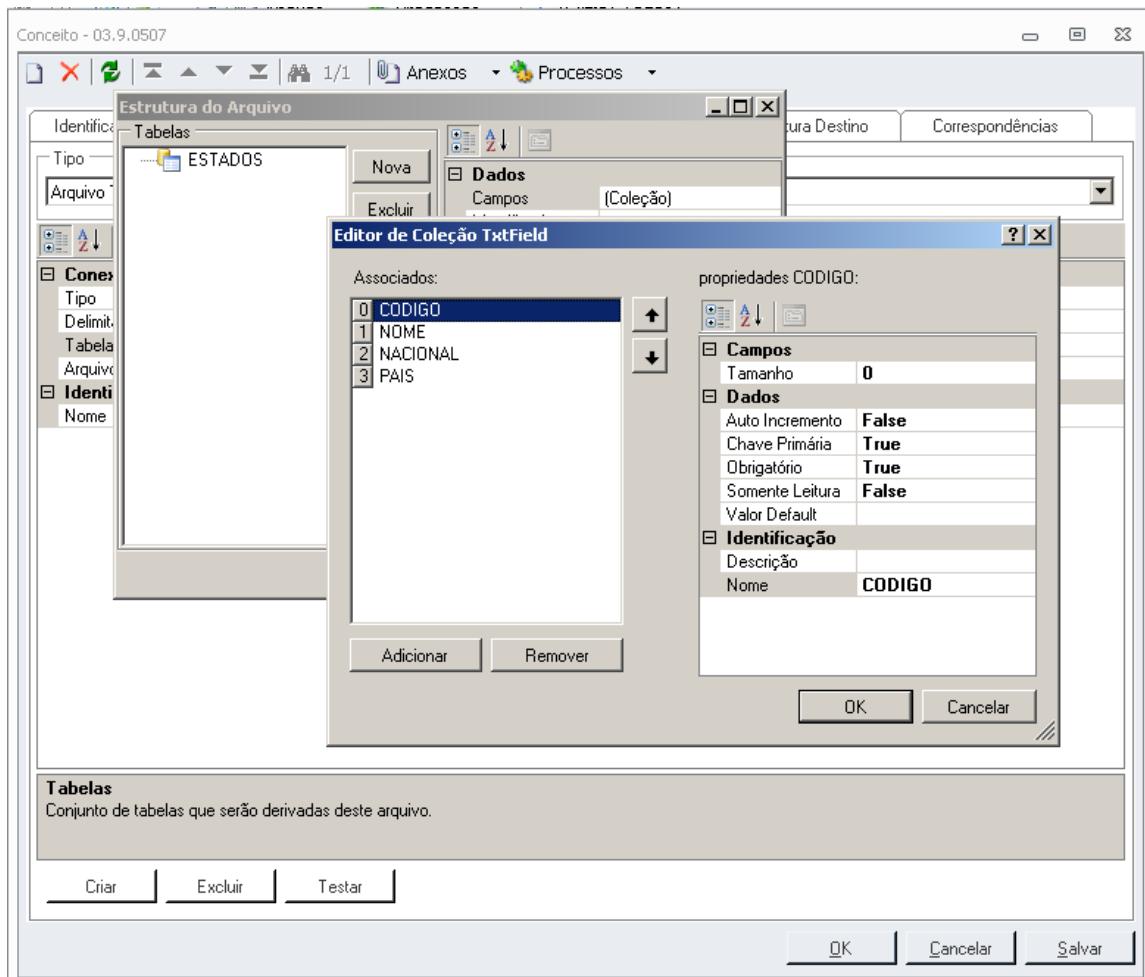
O destino será um arquivo texto e este, em geral, não possui uma estrutura de dados definida (como possui uma tabela de banco de dados, um arquivo XML, etc.) mas o conceito precisa desta estrutura (para mapeamentos e transformações) e permite que a criemos na configuração da Conexão Destino através do atributo Tabelas. Para a definição da estrutura destino voltaremos um pouco até a configuração da conexão destino.

Na aba Conexão Destino, acesse o atributo Tabelas. Clique em Nova para incluir uma tabela e informe um nome.



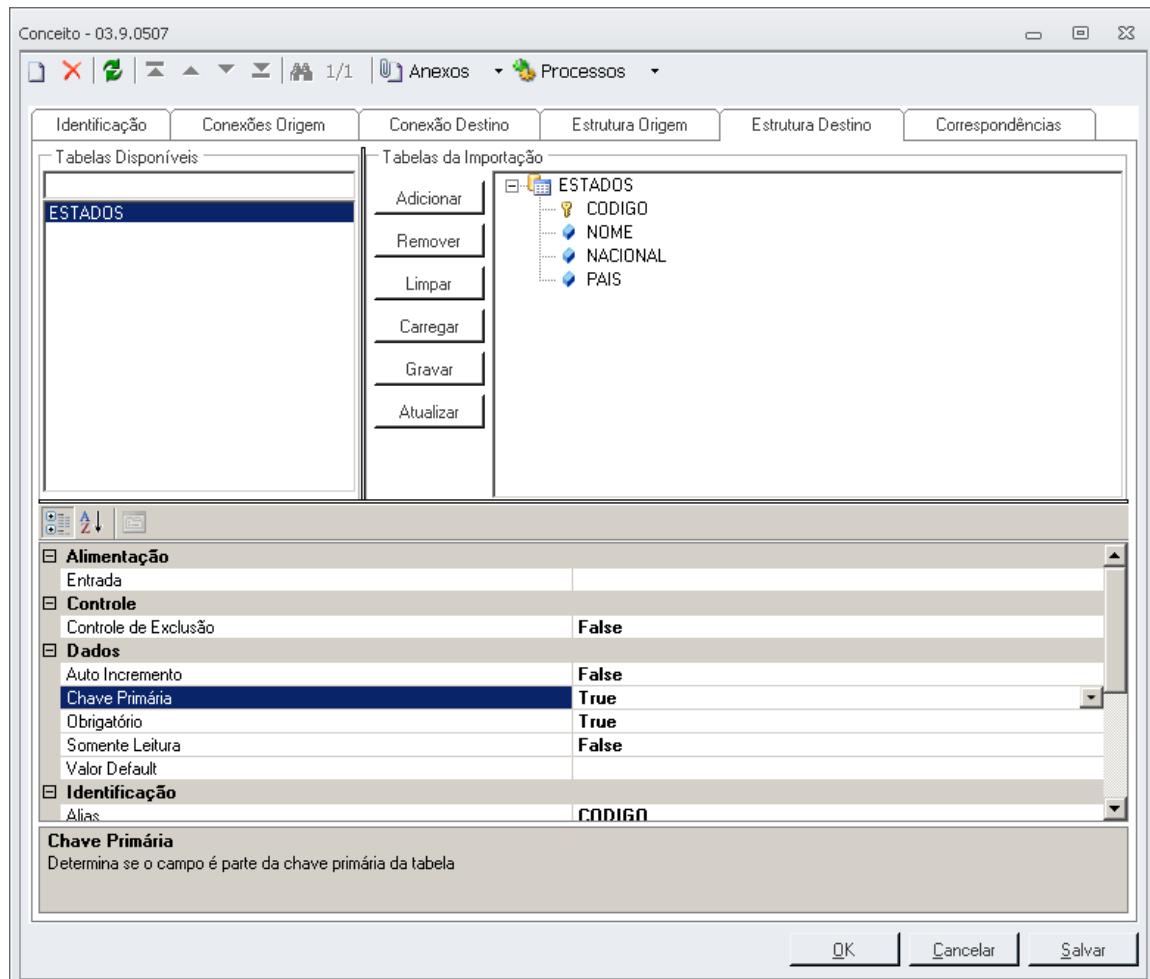
Em seguida, com tal tabela selecionada, acesse seu atributo Campos.

Adicione tantos campos quantos forem os campos da origem selecionados para a exportação (como você poderá notar, os nomes dos campos não precisam, obrigatoriamente, ter os mesmos nomes dos campos origem).



Agora o destino já possui definido uma estrutura de dados e então podemos retornar a definição/configuração da estrutura de dados destino na aba estrutura Destino.

A sua definição/configuração é similar a realizada na Estrutura Origem: selecionar a tabela (no campo Tabelas disponíveis temos agora a estrutura que montamos na Conexão Destino) e adicionar/configurar as colunas.



## 6.4. Correspondências

Após configurar a Estrutura Origem e a Estrutura Destino é necessário realizar as correspondências entre os campos da Origem e os campos do Destino na aba Correspondências.

Nesta aba são apresentadas as tabelas e campos da Estrutura Origem e de Destino para que seja feita a conexão entre elas. Através de uma estrutura visual dos campos origem e destino, é possível visualizar o relacionamento entre os mesmos de acordo com as configurações realizadas.

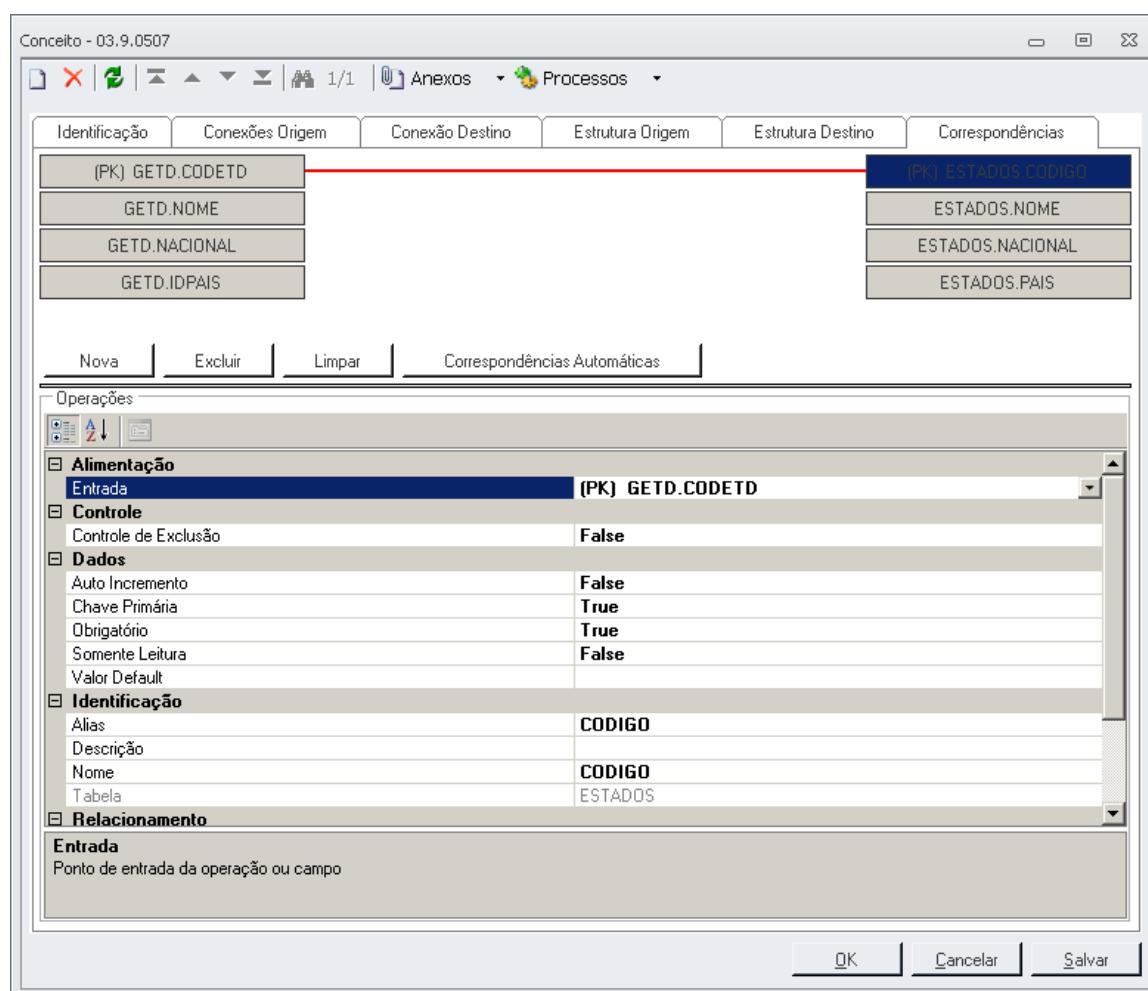
Ao selecionar um campo, suas configurações aparecem na parte inferior da tela. Além de ser possível alterar as configurações dos campos e efetuar o relacionamento entre os mesmos, também estão disponíveis as opções:

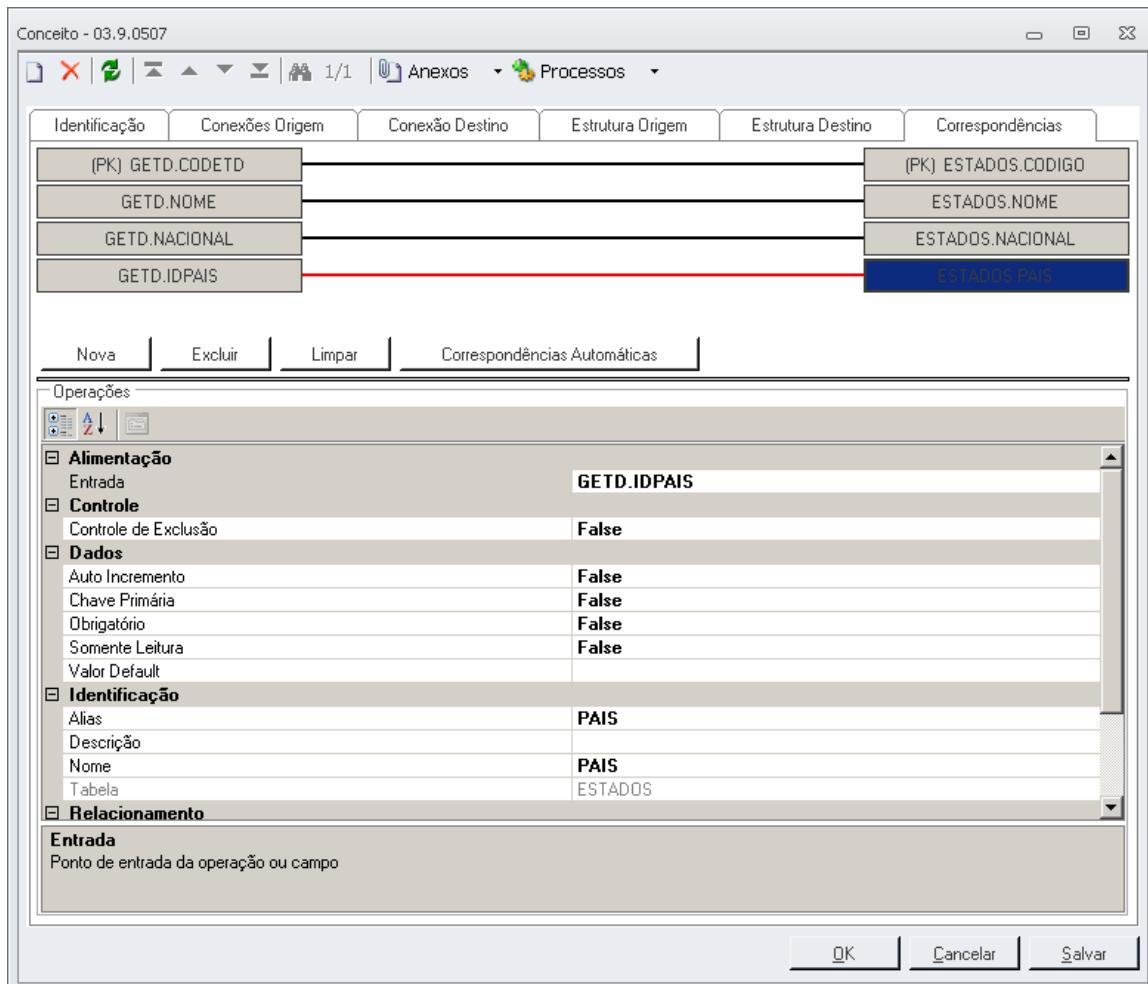
- **Nova:** Uma nova conexão é criada para interligar as tabelas da Estrutura de Origem com as de Destino. Podendo ser das seguintes operações (serão mais detalhadas na seção de Transformações):
  - Arredondamento
  - Auto Incremento – TOTVS RM
  - Concatenação
  - Conversão
  - Data Atual e Date Part
  - GUID
  - Lista de Decodificação

- Lookup e Lookup CRM
- Criptografia MD5
- Sub-String
- Transformação XSLT
- Valor Constante

- **Excluir:** Exclui as conexões selecionadas.
- **Limpar:** Limpa todas as conexões feitas e deixa em branco para que uma nova seja criada.
- **Correspondências Automáticas:** Efetua automaticamente as correspondências entre os campos. Primeiro, é feita a correspondência entre os campos que possuem mesmo nome na conexão origem e na conexão destino. Após, são considerados os campos que não foram relacionados, e feita uma análise de similaridade entre os nomes dos campos na conexão origem e na conexão destino. Caso a correspondência seja efetuada com menos de 70% de similaridade entre os campos, um alerta será exibido para avisar que alguns campos podem não estar com correspondências corretas.

Pode-se realizar a correspondência de forma automática, clicando no botão Correspondências Automáticas, ou selecionando o campo no Destino e informando no parâmetro Entrada qual campo da origem será entrada do campo selecionado.

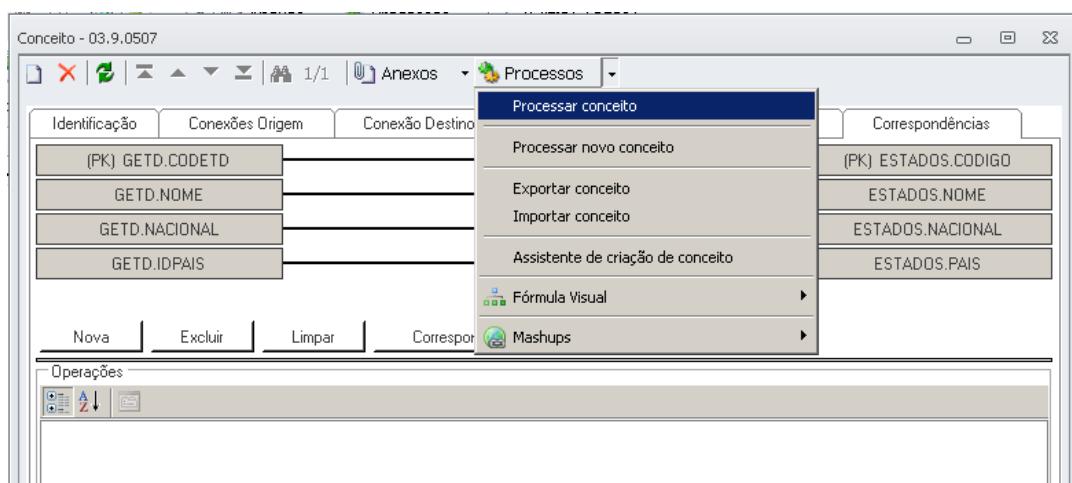


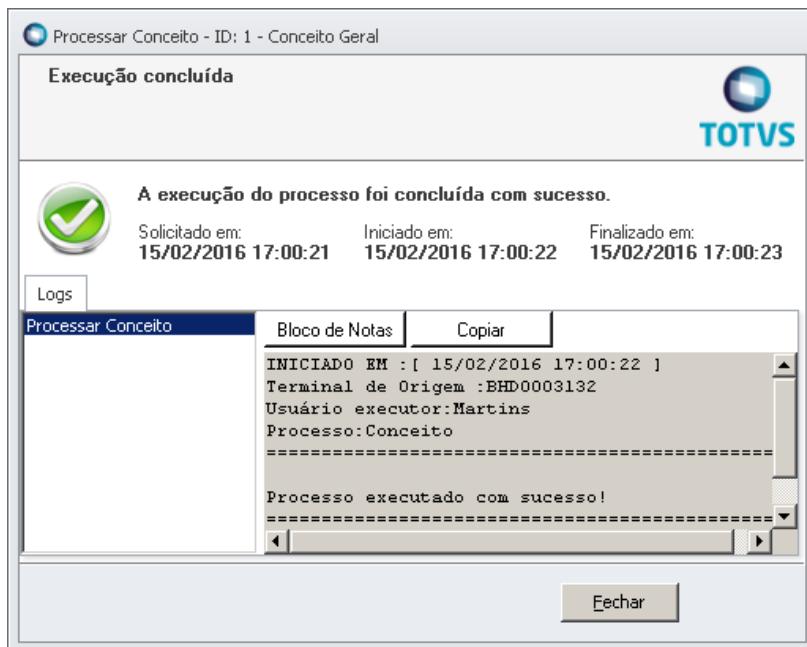
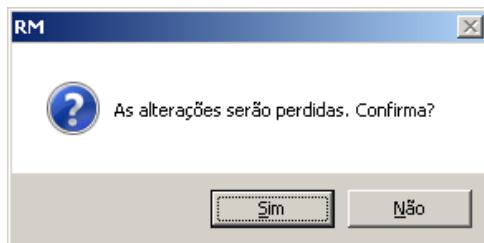


Salve o conceito (gravações parciais podem ser realizadas a qualquer momento).

## 6.5. Execução

Para executá-lo basta acessar o menu de Processos | Processar Conceito para que o conceito seja executado.





```
Estados.txt - Bloco de notas
Arquivo Editar Formatar Exibir Ajuda
-- Outro|N|
AC|Acre|S|1
AL|Alagoas|S|1
AM|Amazonas|S|1
AP|Amapá|S|1
BA|Bahia|S|1
CE|Ceará|S|1
DF|Distrito Federal|S|1
ES|Espírito Santo|S|1
GO|Goiás|S|1
MA|Maranhão|S|1
MG|Minas Gerais|S|1
MS|Mato Grosso do Sul|S|1
MT|Mato Grosso|S|1
PA|Pará|S|1
PB|Paraíba|S|1
PE|Pernambuco|S|1
PI|Piauí|S|1
PR|Paraná|S|1
RJ|Rio de Janeiro|S|1
RN|Rio Grande do Norte|S|1
RO|Rondônia|S|1
RR|Roraima|S|1
RS|Rio Grande do Sul|S|1
SC|Santa Catarina|S|1
SE|Sergipe|S|1
SP|São Paulo|S|1
TO|Tocantins|S|1
US|Estados Unidos|N|9
```

### Exercício

Acesse o cadastro de Conceito e crie o seguinte registro:

#### 1. Origem Banco de Dados → Destino Arquito Texto

Neste exercício, iremos exportar os dados da tabela de Disciplinas (SDISCIPLINA) do Banco de Dados para um Arquivo Texto.

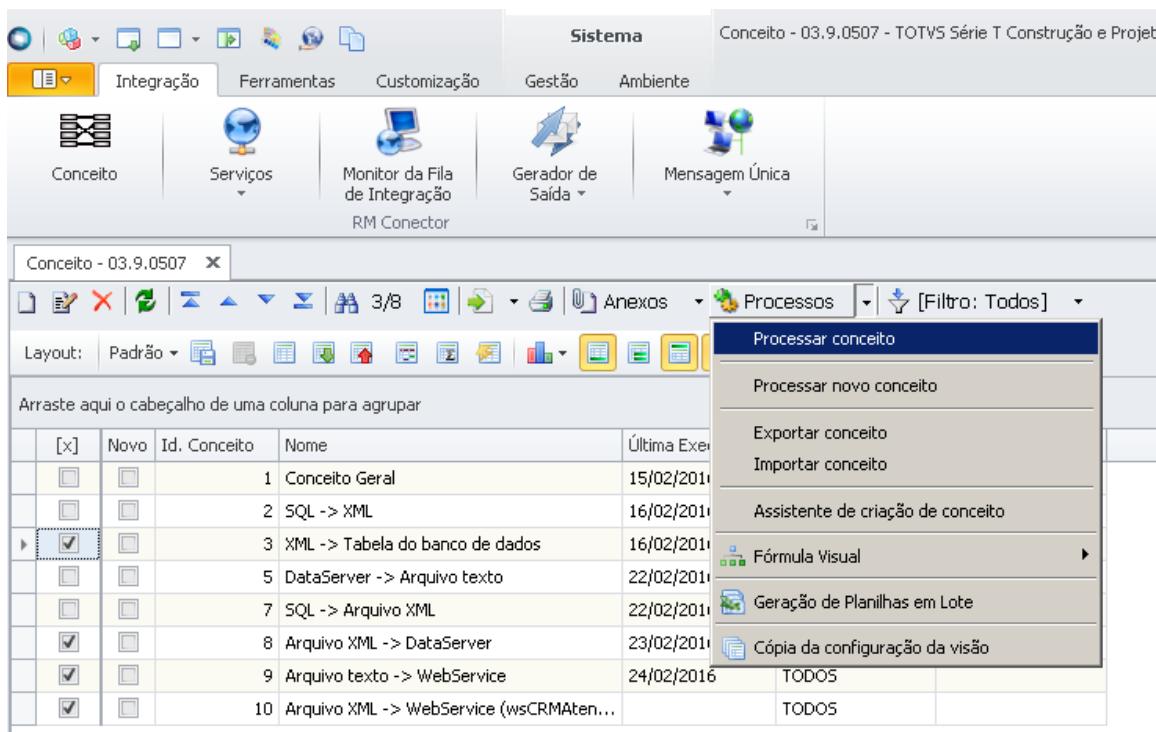
Exporte os campos: **CODCOLIGADA, CODDISC, NOME e CH**

## 6.6. Execução via agendamento

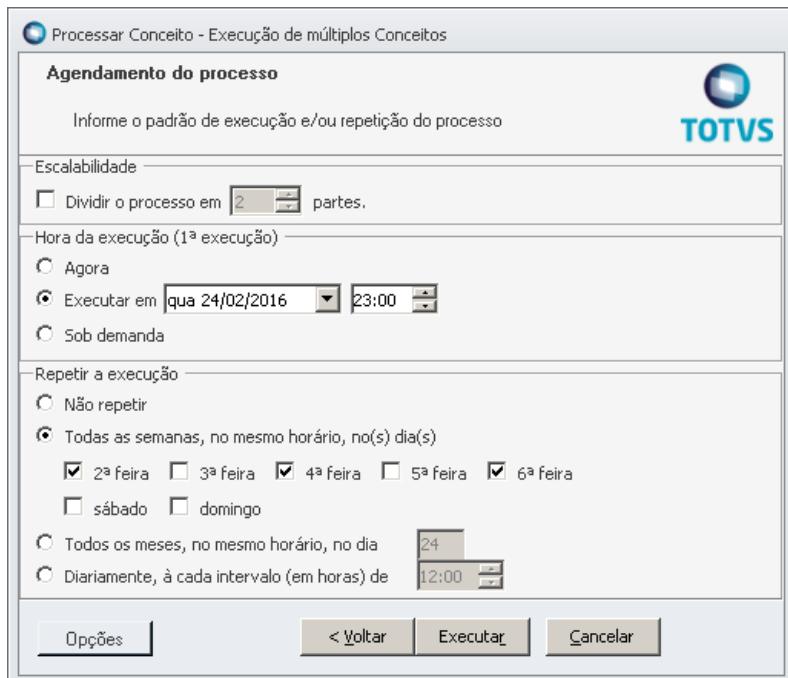
A execução do conceito é realizada por um processo padrão do TOTVS RM então é possível a sua execução via agendamento bem como agendamento com recorrência.

Também é possível efetuar o agendamento/execução de múltiplos de conceitos. Para isso, selecione os conceitos desejados na visão de Conceito e execute o processo "Processar Conceito". Será aberta uma tela para configurar o agendamento do processo. Após a parametrização, os conceitos estarão agendados e serão executados conforme configurado.

**OBSERVAÇÃO:** A recorrência e o agendamento de processos estarão disponíveis para uso somente se o JobServer estiver configurado para execução em 3 Camadas, no arquivo RM.Host.Service.exe.config.



The screenshot shows the TOTVS Business Conector application window. At the top, there's a toolbar with various icons for integration, tools, customization, management, and environment. Below the toolbar, a navigation bar includes 'Sistema' and 'Conceito - 03.9.0507 - TOTVS Série T Construção e Projeto'. Underneath the navigation bar is a row of icons for 'Conceito', 'Serviços', 'Monitor da Fila de Integração', 'Gerador de Saída', and 'Mensagem Única'. The main area is titled 'Conceito - 03.9.0507' and contains a table of concepts. The table has columns for 'Novo', 'Id. Conceito', 'Nome', 'Última Exec.', and 'Status'. Several rows are listed, including 'Conceito Geral', 'SQL -> XML', 'XML -> Tabela do banco de dados', 'DataServer -> Arquivo texto', 'SQL -> Arquivo XML', 'Arquivo XML -> DataServer', 'Arquivo texto -> WebService', and 'Arquivo XML -> WebService (wsCRMAten...'. On the right side of the table, a context menu is open under the 'Processos' button, showing options like 'Processar conceito', 'Processar novo conceito', 'Exportar conceito', 'Importar conceito', 'Assistente de criação de conceito', 'Fórmula Visual', 'Geração de Planilhas em Lote', and 'Cópia da configuração da visão'. The 'Processar conceito' option is highlighted.



**Agendamento do processo**

Informe o padrão de execução e/ou repetição do processo

**Escalabilidade**

Dividir o processo em  partes.

**Hora da execução (1ª execução)**

Agora  
 Executar em

Sob demanda

**Repetir a execução**

Não repetir  
 Todas as semanas, no mesmo horário, no(s) dia(s)  
 2ª feira  3ª feira  4ª feira  5ª feira  6ª feira  
 sábado  domingo

Todos os meses, no mesmo horário, no dia   
 Diariamente, à cada intervalo (em horas) de

**Opções** | < Voltar | Executar | Cancelar |

Conceitualmente, os Jobs são os processos na linha TOTVS RM, e cada um possui informações únicas que vão desde a data da execução do Job, até a estação e o usuário que o executou. O Job é criado no momento da execução de um processo e é registrado no Gerenciador de Jobs.

Os Jobs são executados pelo JobServer que é o mecanismo de execução dos processos. O avanço no uso do JobServer vem da possibilidade de utilizar a Arquitetura 3 Camadas onde funcionalidades da linha RM, bancos de dados e estações de trabalho cliente trabalham fisicamente em separado (em geral uma estação diferente para cada camada).

### Execução Assíncrona de Processos

Alguns processos, que possibilitam e fazem sentido tal tipo de execução, serão executados de forma assíncrona, ou seja, você poderá realizar outras tarefas enquanto o processo é executado. Todos os Jobs que você definir executar dessa forma, serão processados do servidor de aplicação, o que garante melhor desempenho da execução dos processos.

### Agendamento de Processos

Caso você não queira executar o processo imediatamente, ou queira agendar para uma data específica, na linha TOTVS RM é possível agendar para um mês/dia/horário, e ainda configurar essa execução com uma periodicidade pré-definida.

Todos os processos são passíveis de agendamento, mas esta decisão é tomada durante o desenvolvimento e confecção dos processos. A tela de agendamento é apresentada na penúltima etapa do wizard de processos.

### Recorrência de Processos

Para realizar execuções recorrentes basta configurar a repetição do Job, parametrizando execuções semanais, (quais dias da semana serão executados), mensais (qual dia do mês) ou mesmo diariamente (qual horário do dia).

## 7. Transformações

Em uma integração, ou até mesmo ao gerar uma saída (destino), nos deparamos com algumas restrições ou particularidades nos campos destino. Estas particularidades demandam um tratamento no valor do campo origem para então ser aplicado no campo destino.

Em algumas ocasiões precisamos de tratamentos simples – limite/tamanho no tamanho de um campo, tratamentos de datas, arredondamentos, DE/PARA's fixos – mas também podem existir transformações podem ser mais complexas a serem trabalhadas – tipo de dados, geração de identificadores, criptografias, DE/PARA's variáveis (chave estrangeira, mestre-detalhe, lookup).

A ferramenta do conceito do TBC disponibiliza recursos para que determinadas situações possam ser tratadas gerando, como valor destino, o dado da maneira como o campo destino exige.

A este recurso denominamos Transformações e o mesmo está disponível, principalmente (e mais facilmente visualizável), através da aba “Correspondências” do conceito. Em geral as operações são criadas (botão “Nova” na interface de “Correspondências”) – devidamente configuradas com base em uma entrada/origem – e em seguida “ligadas” ao campo destino.

As operações suportadas e seus parâmetros são os seguintes:

- Arredondamento  
Recebe um valor como entrada e este é convertido para um decimal de acordo com as definições do parâmetro "Tipo de Arredondamento".
  - Entrada: um dos campos da origem no qual será aplicada a operação
  - Tipo: Método de arredondamento
    - Arredondar: Arredonda para um valor para o inteiro mais próximo
    - ParaCima: Arredonda o número decimal para o inteiro mais alto
    - ParaBaixo: Arredonda o número decimal para o inteiro mais baixo
    - TruncarDecimais: Remove a parte decimal do número
- Auto incremento - TOTVS RM  
Retorna um valor auto incremento baseado em um valor previamente gravado nas tabelas de auto incremento do TOTVS RM utilizando como referência a coluna informada entre os parâmetros
  - Código do sistema: Informe de qual sistema será buscada a informação. Sendo que deve ser informado a letra do sistema.
  - Coluna: Informe o nome referência do auto incremento que será utilizado para realizar o auto incremento.
  - Driver Name: Informe se o driver é Sql ou Oracle.
  - Nome do Auto Incremento: Informe a coligada.

Exemplo supondo que o valor do IDCONCEITO na tabela é 15:

Código do Sistema: G  
Coluna: IDCONCEITO  
Driver Name: SqlServer  
Nome do Auto Incremento (coligada): 1  
Retorno da operação: 16.

- Concatenação  
Recebe duas entradas e resulta em uma saída com o resultado da operação definida pelos parâmetros "Tipo de dados".
  - Entrada: Ponto de entrada da operação ou campo.

- Entrada Secundária: Valor a ser concatenado à entrada principal.
- Concatenador: Caracteres que serão incluídos entre as entradas principal e secundária, funcionando como um separador de dados.
- Tipo de Dados: Tipos de dados resultante da operação, que pode ser Texto, Inteiro, Decimal, Data.

Se o campo "Tipo de Dados" estiver definido como texto os dados serão concatenados. Mas essa operação também pode executar funções de soma de valores do tipo Inteiro e Decimal.

Caso esse campo esteja definido como inteiro as entradas serão convertidas para inteiro e então será realizada a soma dos valores (o mesmo vale para o tipo decimal). Neste caso, o campo Concatenador deve ser deixado em branco ou preenchido com um valor numérico que será acrescentado ao total da soma das duas entradas.

#### ▪ Conversão

Transforma o dado de entrada para o tipo de dado informado na parametrização.

- Entrada: Ponto de entrada da operação ou campo.
- Tipo de Dados: Tipos de dados resultante da operação, que pode ser: Texto (converte para String), Inteiro (converte para int64), Decimal (converte para Double), Data (converte para Datetime)

#### ▪ Data Atual

Retorna a data atual do sistema no formato definido nos parâmetros

- Formato de Data: Define qual o formato de data será utilizado. Os formatos podem ser: Mês Dia Ano, Dia Mês Ano, Ano Mês Dia, Ano Dia Mês
- Hora: Define se será adicionada à data, as horas, minutos e segundos.
- Separador: Define qual o caractere será utilizado para separar as partes da data.

#### ▪ Date Part

Retorna uma parte da data que é recebida como entrada. A parte da data a ser retornada é definida na parametrização.

- Entrada: Ponto de entrada da operação ou campo.
- Parte: A parte da data que deseja ser retornada. Pode ser: Dia, Mês, Ano, Dia do ano, Dia da semana, Hora, Minuto, Segundo

#### ▪ GUID - Globally Unique Identifier

Gera e retorna um identificador UUID (Universal Unique Identifier).

- Entrada: campo que deverá receber o identificador

#### ▪ Lista de Decodificação

Valores de origem são convertidos para um valor de destino específico através de uma lista de conversão. A lista de conversão contém o valor origem e qual valor destino ele representa e deve ser informado nos parâmetros desta operação.

- Valor Padrão: Valor a ser utilizado caso o valor de origem não corresponda a nenhum dos valores mapeados na propriedade Valores.
- Valores: É uma lista de pares de valores a serem utilizados para a conversão (DE/PARA). Tal lista é composta por pares com os seguintes valores:
  - Código: Informe o valor do campo na origem.
  - Valor: Informe o valor que o campo deverá ter no destino.

#### ▪ Lookup

A operação Lookup recebe um valor e retorna, de acordo com as parametrizações realizadas, outro valor.

Os parâmetros para configurar a operação Lookup são:

- Entrada: Ponto de entrada da operação ou campo.

- Campo Chave: Informe o campo da tabela que deve ter seu valor comparado com o campo informado na Entrada.
  - Campo Valor: Informe o campo que deve ter seu valor retornado pelo lookup.
  - Nome da Conexão: Informe o nome da conexão de origem utilizada.
  - Nome da Tabela ou Query: Informe o nome da tabela ou query que será utilizada para o lookup (tabela ou query cujo campo foi informado na propriedade "CampoValor").
- Criptografia MD5  
Recebe a entrada, criptografa os dados e os envia para o campo destino.
    - Entrada: Ponto de entrada da operação ou campo.
  - Sub-String  
Recebe uma entrada e retorna somente uma Sub-String (parte do texto) determinada pelos parâmetros. Inicia de uma posição, determinada pelo campo Início, e possui um tamanho específico, retornando o número de caracteres informado no parâmetro Quantidade. A posição inicial é sempre 0 (zero) e a quantidade tem de estar dentro dos limites da string de entrada.
    - Entrada: Ponto de entrada da operação ou campo.
    - Início: Índice da primeira letra da Sub-String.
    - Quantidade: Número de caracteres da Sub-String.
  - Transformação XSLT  
A operação de Transformação XSLT transforma arquivos XML em outros formatos XML.  
Os parâmetros para configurar esta operação são:
    - Entrada: Ponto de entrada da operação ou campo.
    - Inclui Cabeçalho: Inclui nos dados transformados o cabeçalho XML.
    - XSLT: Informe o XSLT de transformação.
  - Valor Constante  
Retorna o valor definido na configuração da operação.
    - Alimentação: Valor constante da operação.

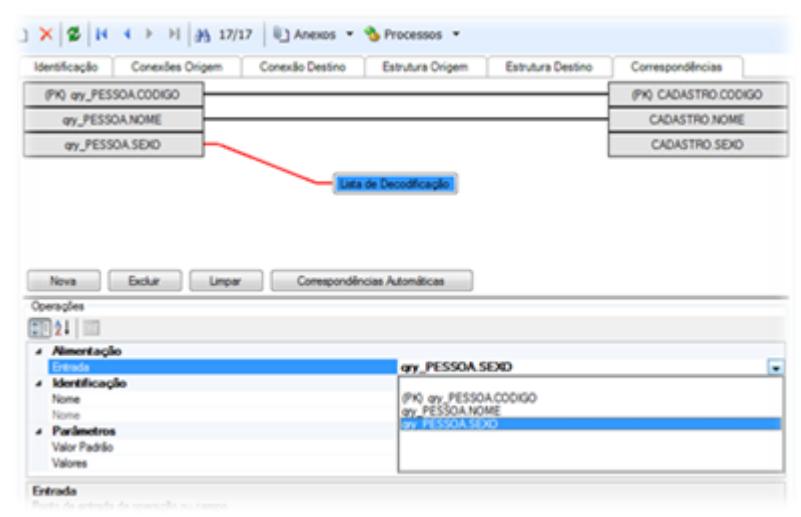
## 7.1. Conceito com operação de Lista de decodificação

Na operação Lista de Decodificação, valores de origem são convertidos para um valor de destino específico através de uma lista de conversão.

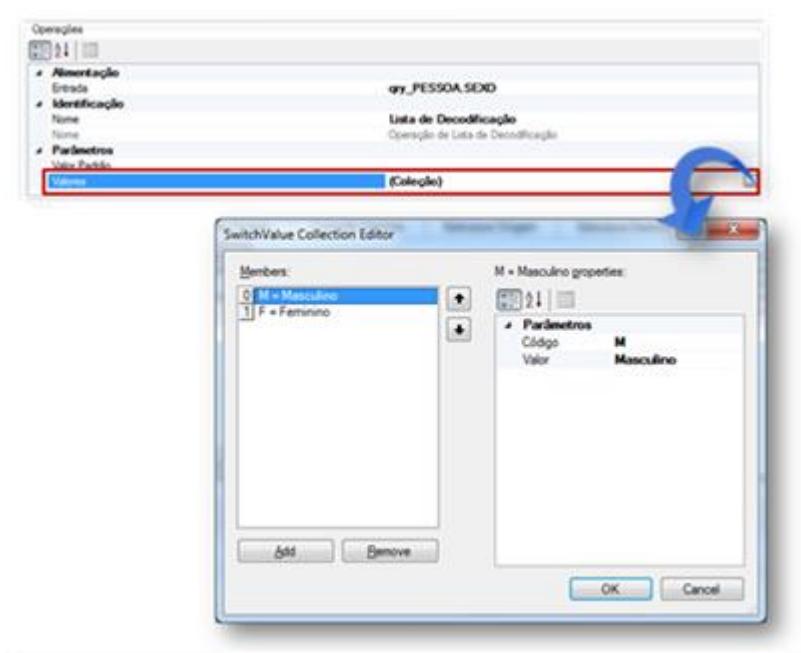
Após criar e configurar um conceito, clique na aba Correspondências e crie uma nova operação do tipo Lista de Decodificação.

Configure os campos referentes à conexão.

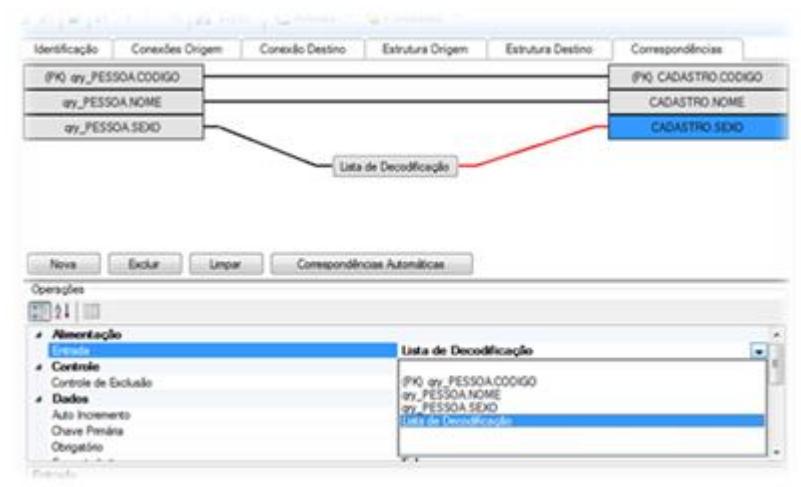
- Entrada: Informe a entrada da operação (Exemplo: qry\_PESSOA.SEXO)
- Valor Padrão: Informe o valor a ser utilizado caso o valor de origem não corresponda a nenhum dos valores mapeados na propriedade Valores.



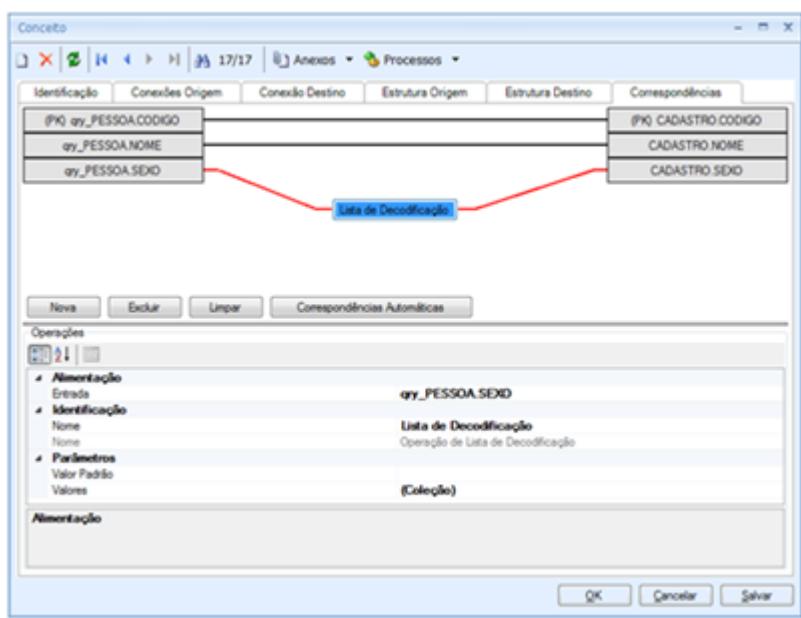
- Valores: Ao clicar no lookup do campo, é aberta uma visão onde deve ser criada a lista de conversão, informando os valores de origem e seus respectivos valores no destino.
- Clique no botão Add para adicionar uma relação de valores origem e destino, e depois configure os campos.
- Código: Informe o valor do campo na origem.
  - Valor: Informe o valor que o campo deverá ter no destino.



Selecione o campo destino que deve receber o retorno da operação (Exemplo: CADASTRO.SEXO)



Após realizar os procedimentos anteriores, a operação Lista de Decodificação está configurada.

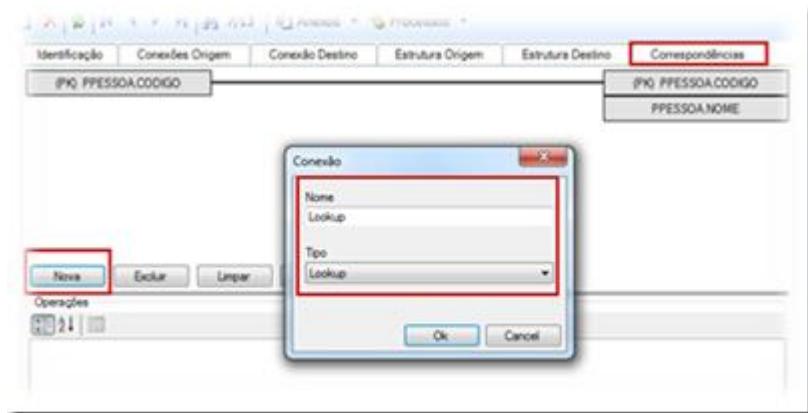


Com base nos exemplos, a operação Lista de Decodificação irá receber como entrada o valor do campo “qry\_PESSOA.SEXO” que poderá ter como valor ‘M’ ou ‘F’.

A operação irá buscar o valor recebido como entrada no campo Código da lista de conversão, identificar o valor correspondente no campo Valor (“Masculino” para entrada ‘M’, “Feminino” para entrada ‘F’), e retornar o conteúdo deste campo para o destino, campo CADASTRO.SEXO.

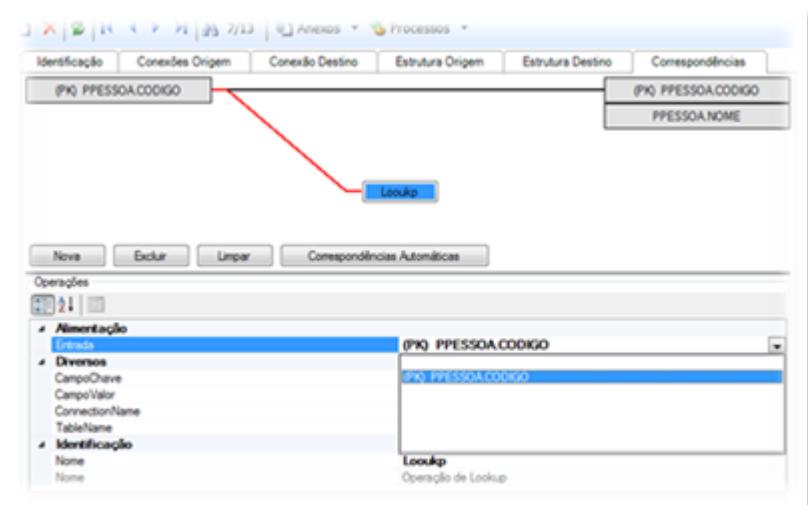
## 7.2. Conceito com operação de Lookup

Após criar e configurar um conceito, clique na aba “Correspondências” e crie uma nova operação do tipo Lookup.



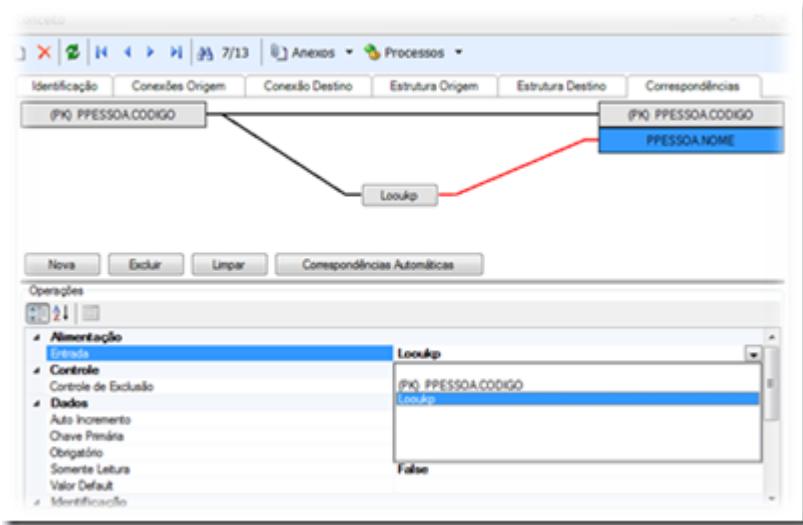
Configure os campos referentes à conexão:

- Entrada: Ponto de entrada da operação ou campo (Exemplo: (PK) PESSOA.CODIGO)
- CampoChave: Informe o campo que deve ter seu valor retornado pelo lookup (Exemplo: PRIMEIRONOME)
- CampoValor: Informe o campo da tabela que deve ter seu valor comparado com o campo informado na Entrada (Exemplo: CODIGO)
- ConnectionName: Informe o nome da conexão de origem utilizada (Exemplo: Banco de Dados [nome da Conexão Origem])
- TableName: Informe o nome da tabela que será utilizada para o lookup ([tabela cujo campo foi informado em CampoValor], Exemplo: PESSOAAMX)

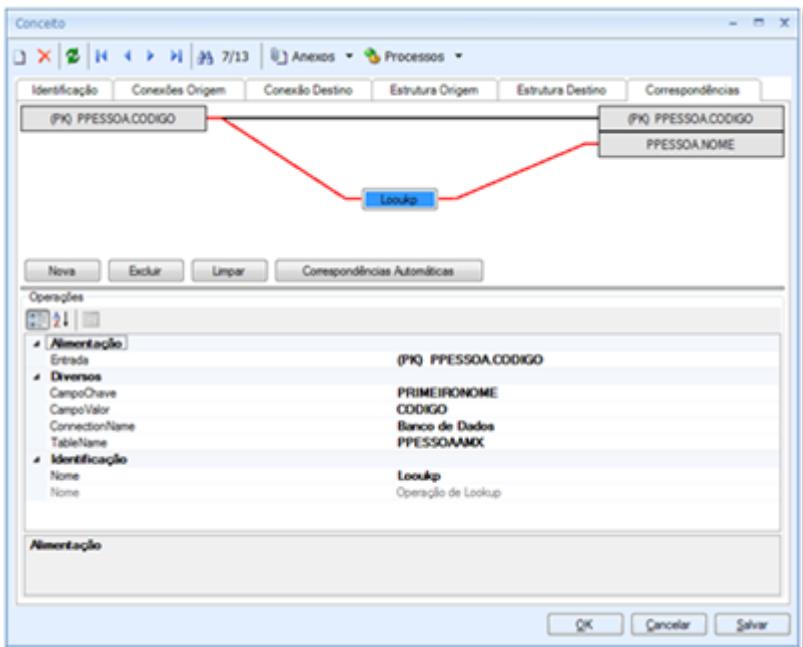


Selecione o campo destino que deve receber o retorno da operação (Exemplo: PESSOA.NOME)

## TOTVS Business Conector



Após realizar os procedimentos anteriores, a operação Lookup está configurada.



Com base nos valores fornecidos nos exemplos, a operação Lookup irá buscar na tabela PESSOAMX, disponibilizada na conexão Banco de Dados, o campo CODIGO que contenha o mesmo valor que o campo (PK) PESSOA.CODIGO, e retornar o valor do campo PRIMEIRONOME correspondente ao campo CODIGO para no campo Destino, PESSOA.NOME.

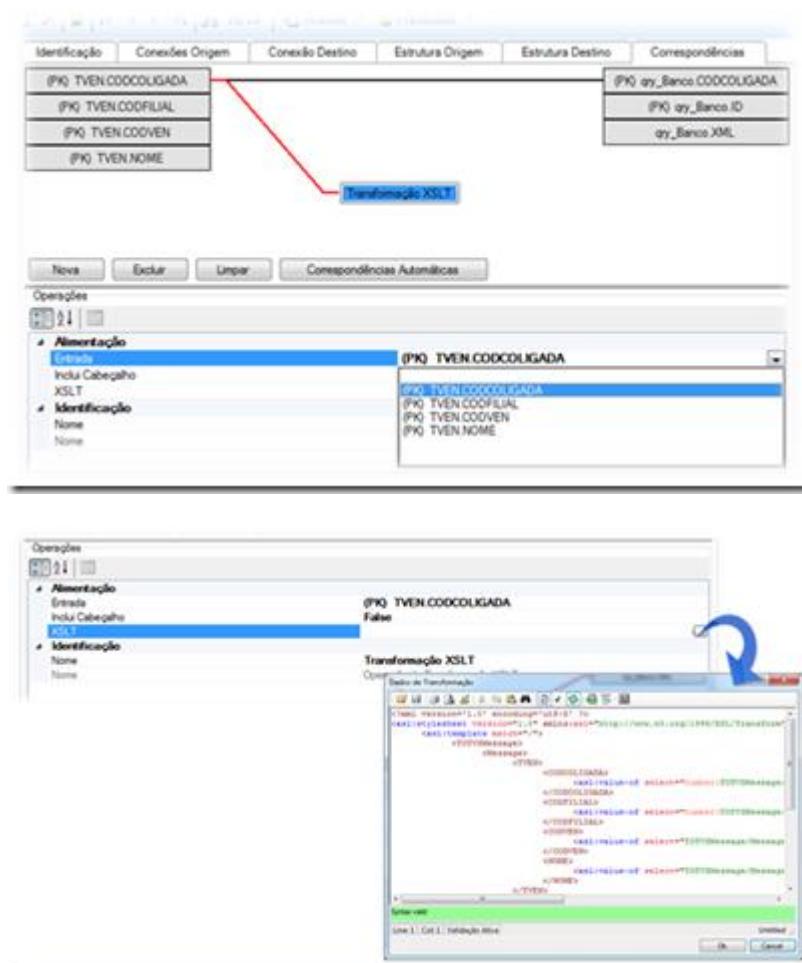
## 7.3. Conceito com operação de Transformação XSLT

A operação de Transformação XSLT transforma arquivos XML em outros formatos XML. Neste caso a Conexão Origem deve ser do tipo Arquivo XML.

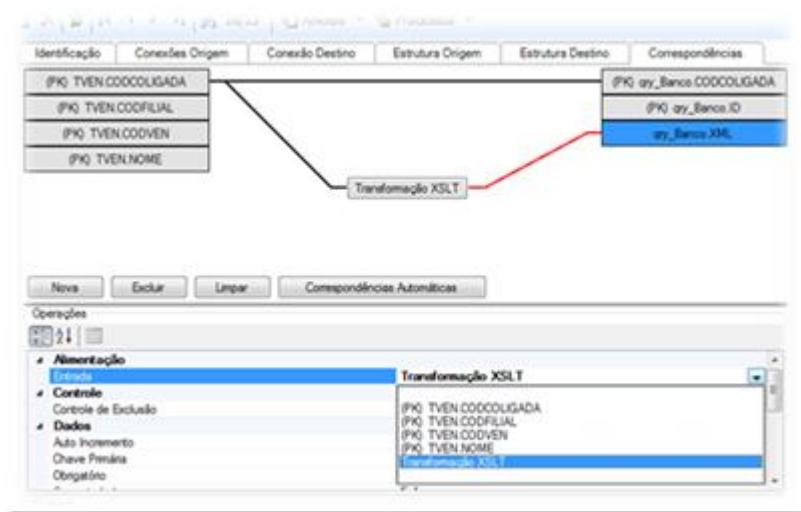
Após criar e configurar um conceito, clique na aba “Correspondências” e crie uma nova operação do tipo Transformação XSLT.

Configure os campos referentes à conexão.

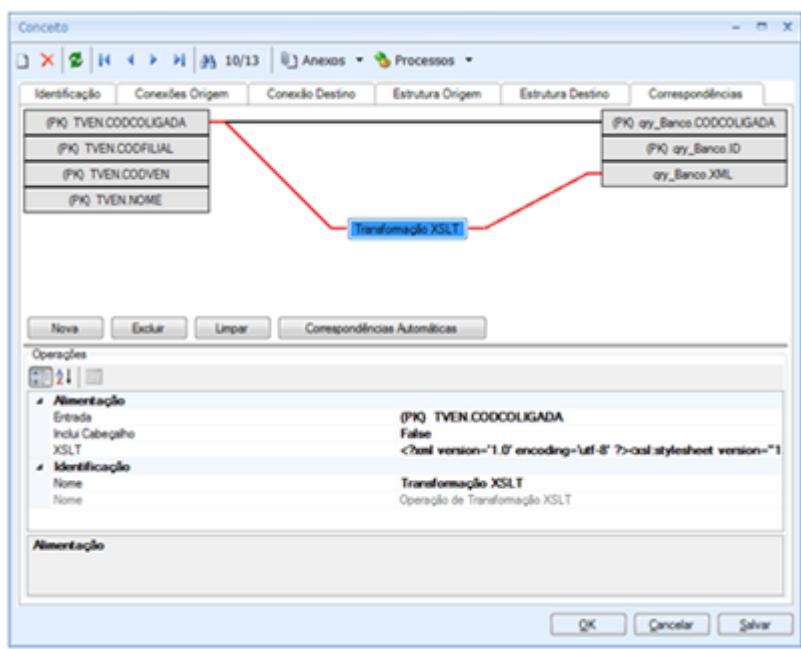
- Entrada: Ponto de entrada da operação ou campo (Exemplo: (PK) TVEN.CODCOLIGADA)
  - Inclui Cabeçalho: Inclui nos dados transformados o cabeçalho XML.
  - XSLT: Informe o XSLT de transformação.



Selecione o campo destino que deve receber o retorno da operação (Exemplo: qry\_Banco.XML)



Após realizar os procedimentos anteriores, a operação Transformação XSLT está configurada.



Com base nos valores fornecidos nos exemplos, a operação Transformação XSLT irá receber um arquivo XML a partir da entrada “(PK) TVEN.CODCOLIGADA” (conforme configurado na Conexão Origem do tipo Arquivo XML), transformar o mesmo com base no XSLT de transformação informado, e retorna-lo para o campo destino “qry\_Banco.XML”.

## 8. Conexões Entrada X Conexões Destino

Segundo apresentado, o Conceito pode trabalhar com alguns tipos pré-definidos para conexão que são: Arquivo Texto, Arquivo XML, Banco de Dados, Data Servers, MS CRM, WebServices (sendo este último apenas na conexão destino). Com isto abre-se a possibilidade da combinação de conexões origem X conexões destino utilizando-se os tipos citados na criação dos conceitos atendendo assim uma ampla variedade de situações/contextos do ambiente (sistemas) do cliente.

Uma vez conhecido a estrutura base do conceito, como é criado, configurado e executado passaremos agora a apresentar algumas combinações possíveis onde, para os próximos exemplos, nos focaremos nas particularidades, itens de interesse e pontos de atenção relativas às variações nos tipos de conexão de origem/destino.

### 8.1. Conceito com arquivos XML

XML é uma linguagem para a criação de documentos com dados organizados hierarquicamente, tais como textos, banco de dados. Possibilitando a separação entre formato e conteúdo (reutilização com diferentes propósitos) é classificada como extensível porque permite definir seus elementos (arquivos XSD).

#### 8.1.1. Origem banco de dados (SQL) e Destino arquivo XML

Neste exemplo apresentaremos um exemplo onde os dados serão extraídos (portanto origem) de uma base de dados através de uma consulta SQL e como saída (portanto destino) um arquivo XML com os valores referentes às consultas SQL de origem.

Para a recuperação dos dados utilizaremos um SQL que “pega” registro da tabela de cliente/fornecedor do TOTVS RM que é similar a abaixo:

```
SELECT CODCOLIGADA, CODCFO, NOME, PAGREC, ATIVO, PESSOAFISOUJUR, CGCCFO, CIDADE, TELEFONE, IDCFO,
TIPOCONTRIBUINTEINSS
FROM FCFO (NOLOCK)
WHERE CGCCFO IS NOT NULL
AND CIDADE IS NOT NULL
AND TELEFONE IS NOT NULL
```

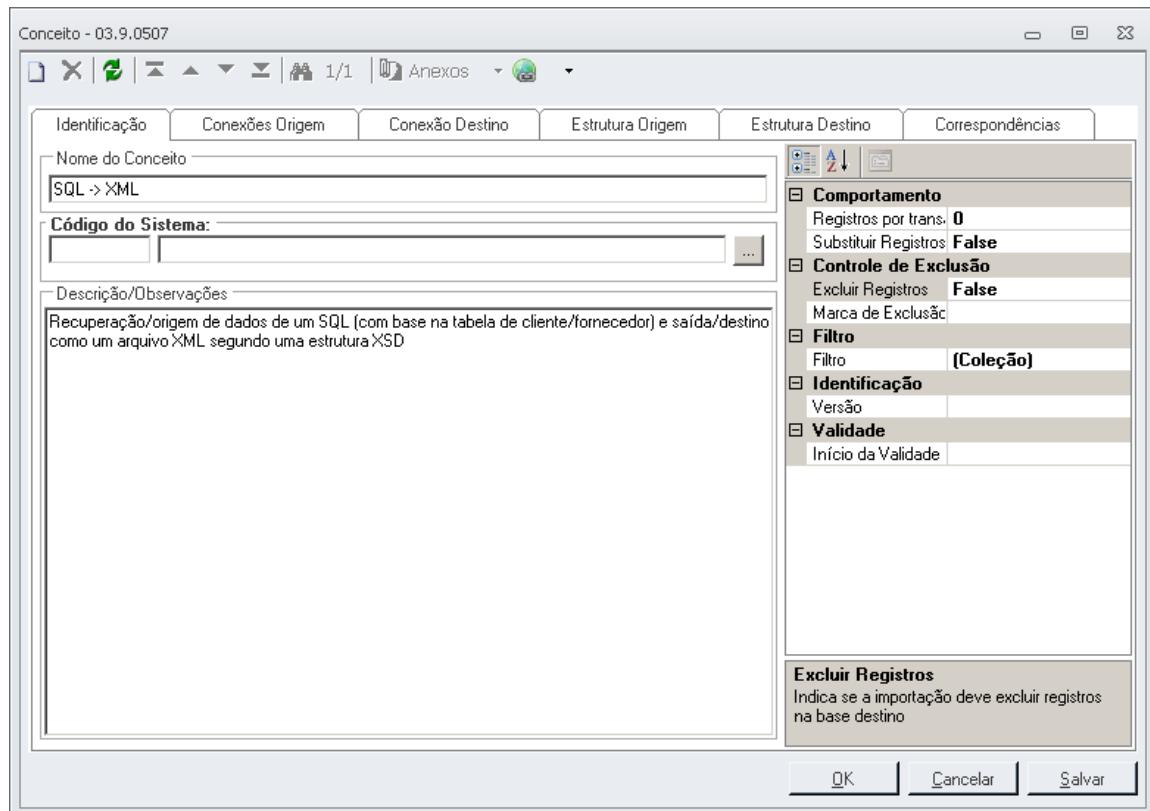
Como a saída será um XML, será necessário informar um XSD (XML Schema Definition) que definirá o “padrão” que XML seguirá. Este XSD poderá ser informado diretamente no conceito em atributo próprio para tal ou carregar o XSD através de um arquivo previamente montado com tal schema. O XSD a ser utilizado neste exemplo é o apresentado abaixo:

```
<?xml version="1.0" encoding="utf-8"?>
<xss:schema xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" xmlns:msprop="urn:schemas-microsoft-
com:xml-msprop" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xss:element name="FCFO">
    <xss:complexType>
      <xss:all>
        <xss:element name="CODCOLIGADA"></xss:element>
        <xss:element name="CODCFO"></xss:element>
        <xss:element name="NOME"></xss:element>
        <xss:element name="PAGREC"></xss:element>
        <xss:element name="ATIVO"></xss:element>
        <xss:element name="PESSOAFISOUJUR"></xss:element>
        <xss:element name="CGCCFO"></xss:element>
        <xss:element name="CIDADE"></xss:element>
        <xss:element name="TELEFONE"></xss:element>
        <xss:element name="IDCFO"></xss:element>
        <xss:element name="TIPOCONTRIBUINTEINSS"></xss:element>
      </xss:all>
    </xss:complexType>
  </xss:element>
</xss:schema>
```

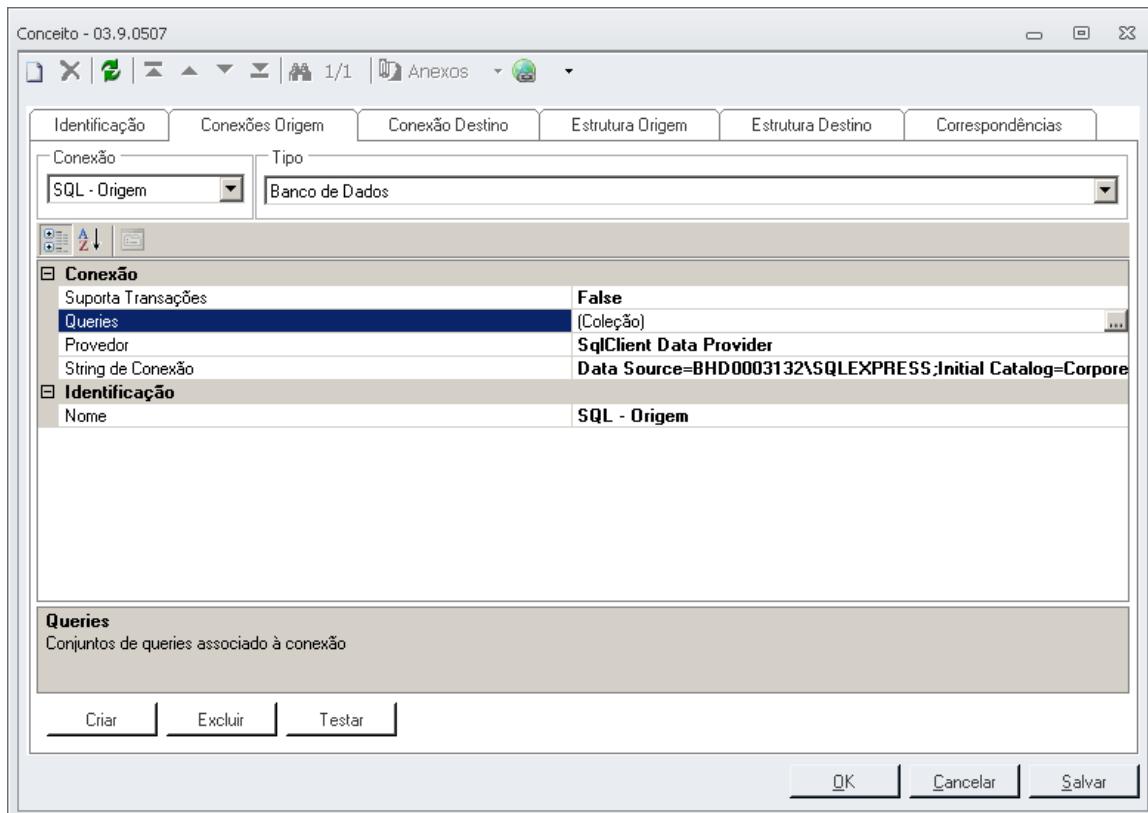
```
</xs:element>
</xs:schema>
```

Agora podemos criar o conceito com as seguintes características (valores dos atributos):

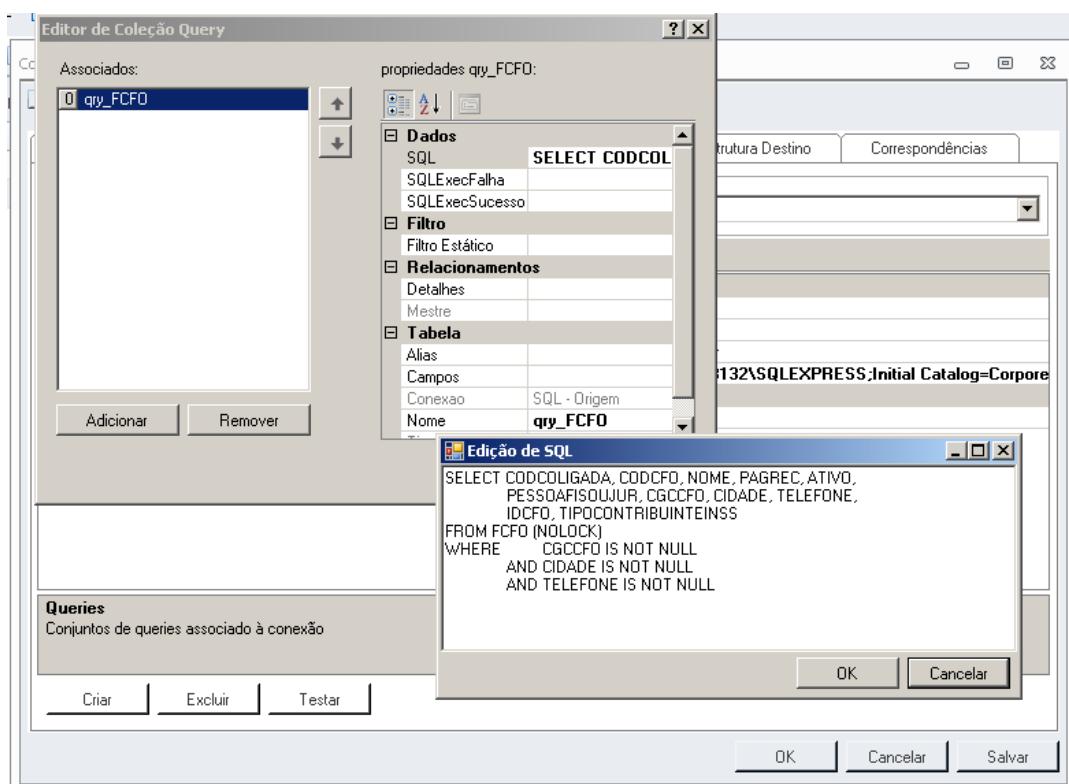
- Identificação
  - Nome: SQL -> XML (ou a gosto do participante);
  - Descrição: Recuperação/origem de dados de um SQL (com base na tabela de cliente/fornecedor) e saída/destino como um arquivo XML segundo uma estrutura XSD (ou a gosto do participante);



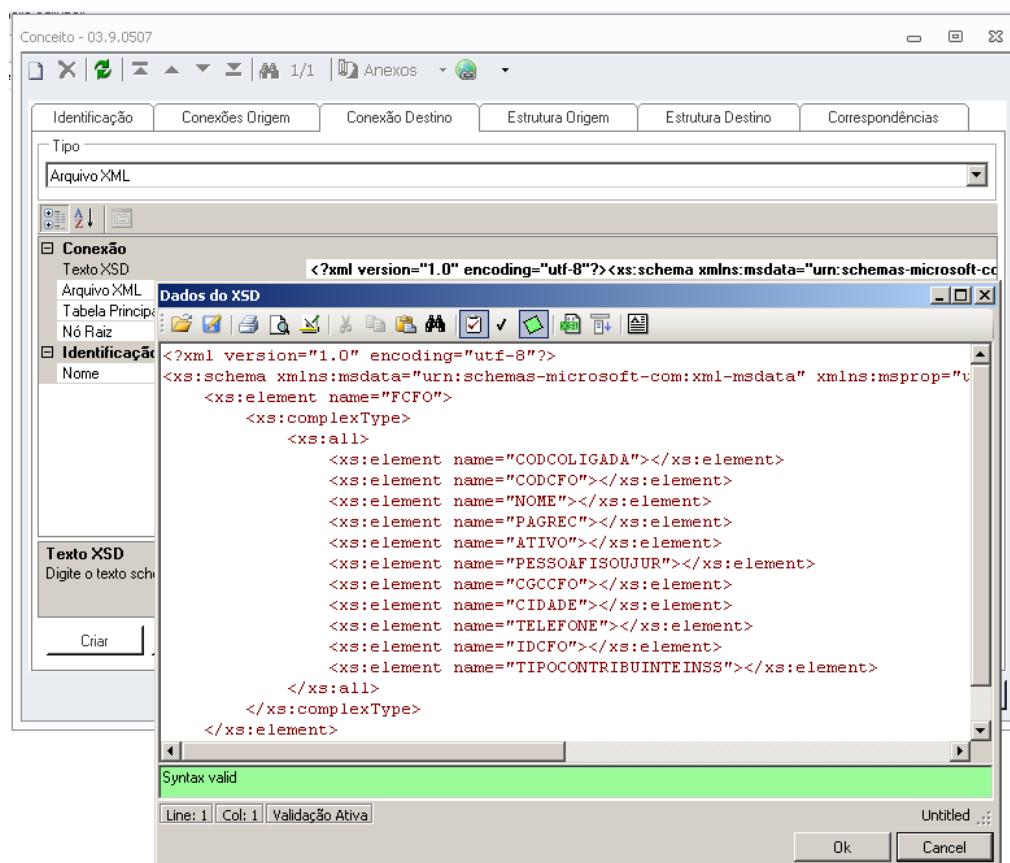
- Conexão Origem
  - Nome: SQL – Origem (ou a gosto do participante);
  - Tipo: Banco de dados;
  - Provedor: SqlClient Data Provider;
  - String de Conexão: Montar a string de conexão desejada (informar o servidor e banco de dados origem conforme realizado em exemplos/prints anteriores);



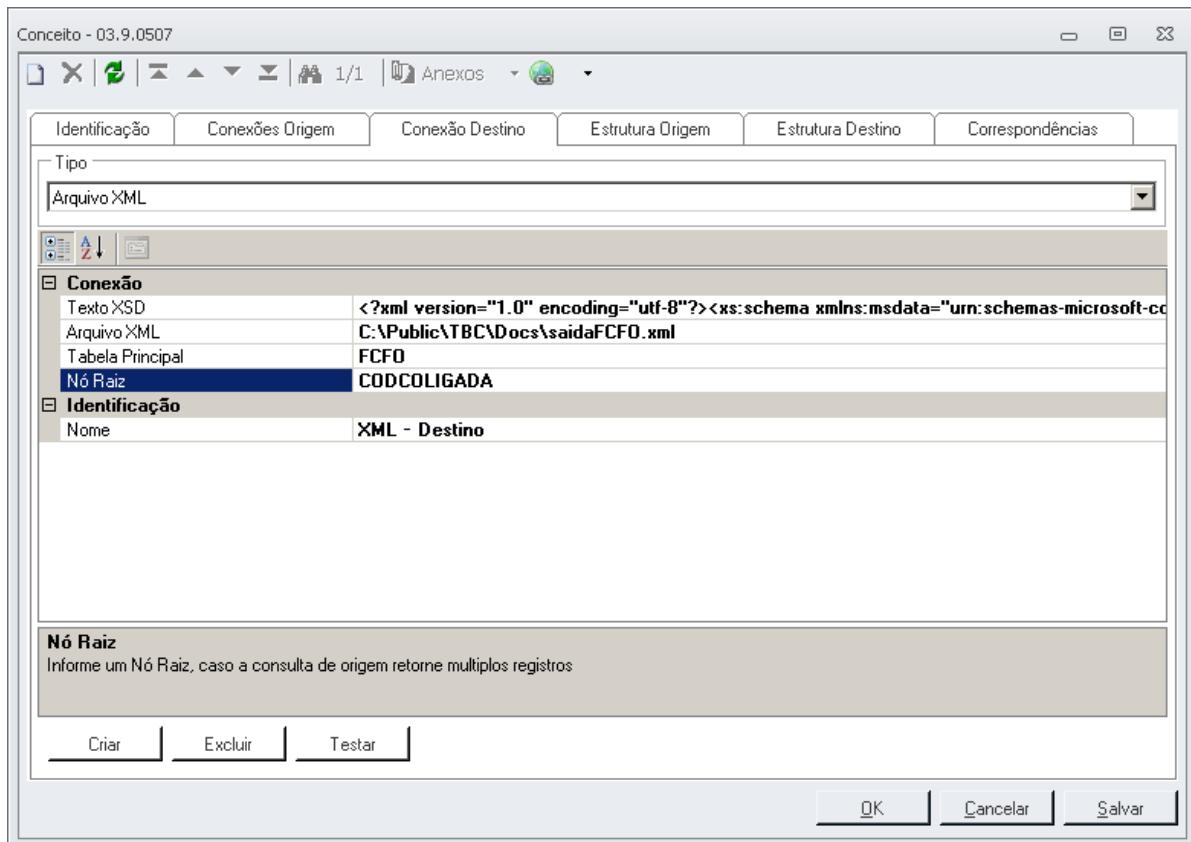
- Queries: Adicione uma query com o SQL conforme citado anteriormente (mais acima) e dê um nome a esta query (exemplo: qry\_FCFO);



- Conexão Destino
  - Nome XML – Destino (ou a gosto do participante);
  - Tipo: Arquivo XML;
  - Texto XSD: Informe o texto schema no formato XSD. Ao clicar no lookup para informar o texto XSD é aberta uma visão na qual é possível digitar o texto schema, ou abrir o arquivo XSD para que o texto seja carregado;
  - Arquivo XML: Informe o caminho completo e o nome do arquivo XML;
  - Tabela Principal: Selecione a tabela principal, apenas para conexão de origem;
  - Nó Raiz: Caso a consulta de origem tenha como retorno múltiplo registros, informe neste campo o nó raiz. Em nosso exemplo será CODCOLIGADA;

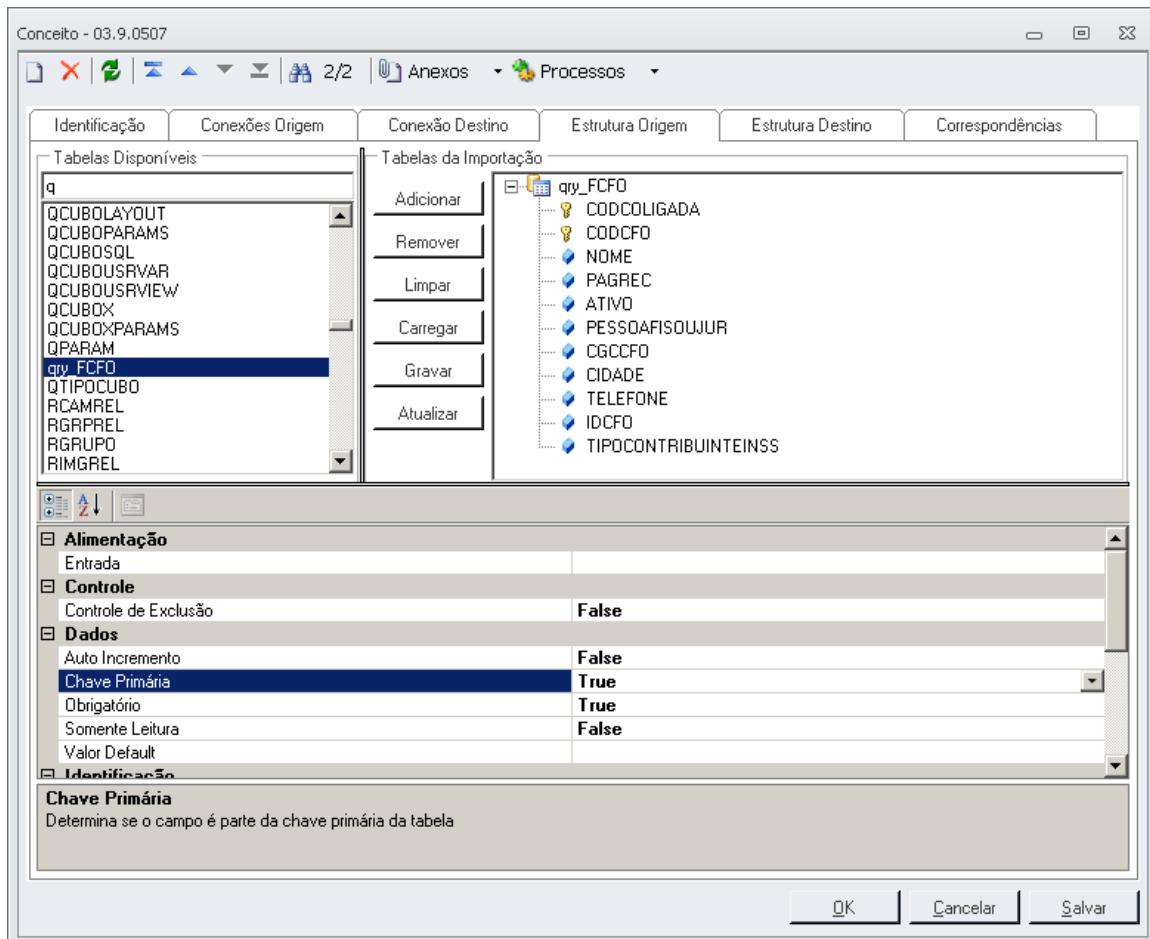


**OBSERVAÇÃO:** Note acima que o esquema (XSD) foi criado com a Tabela FCFO e como atributos/TAGs os mesmos campos/colunas do SQL de origem desta forma as Correspondências ocorrem sem inconsistências.

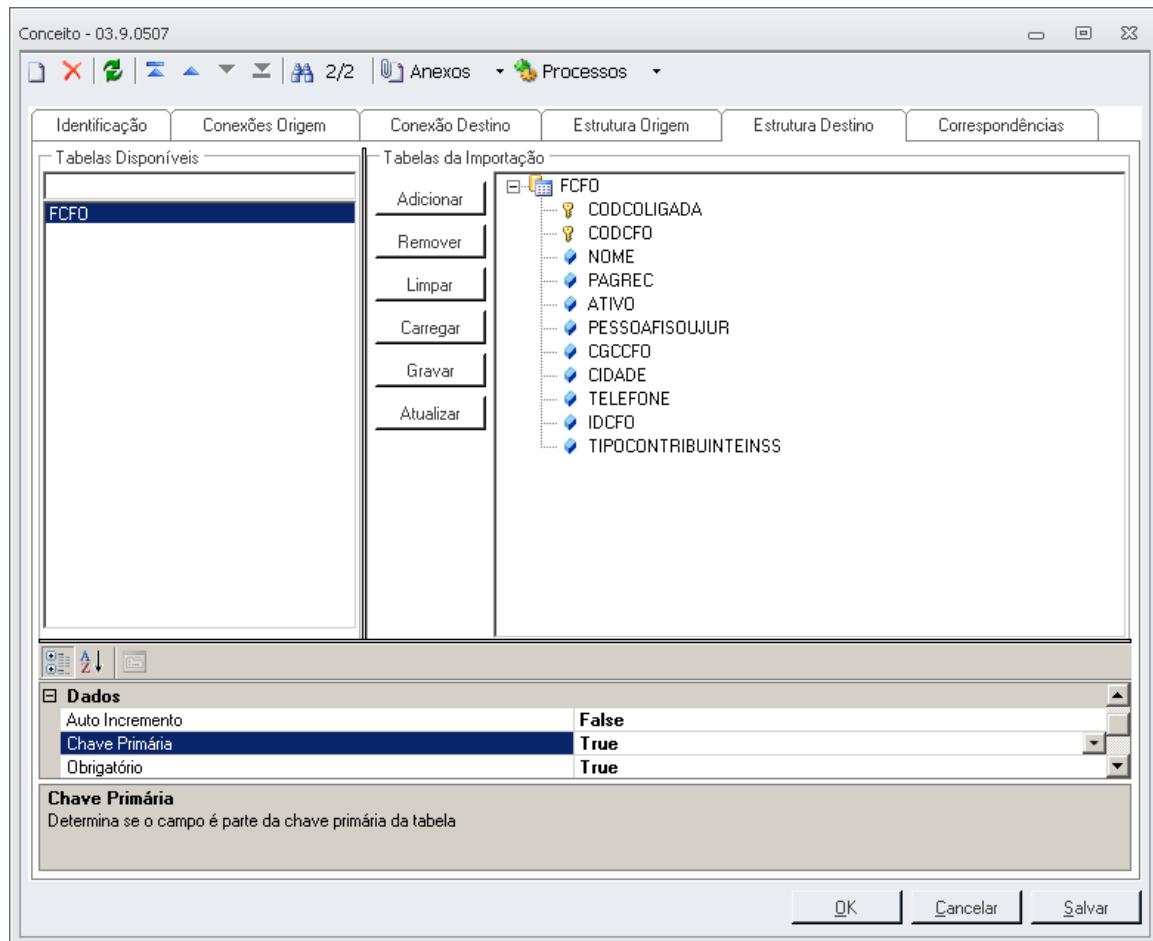


O passo seguinte, assim como o conceito geral (feito como exemplo base), é a definição das Estruturas Origem e Destino.

Como estrutura origem, em tabelas disponíveis, selecionaremos a query que criamos (origem) e adicionaremos os campos/colunas desta query. Após a adição destas, selecione as que são a chave primária (CODCOLIGADA E CODCFO) e, na parte debaixo das colunas onde são apresentados os atributos da tabela/coluna, altere para “True” o atributo “Chave Primária”.



Para Estrutura Destino, em tabelas disponíveis, será apresentado o arquivo XML que criarmos na conexão destino. Similar ao realizado para estrutura origem, em tabelas disponíveis, selecione nosso arquivo XML (FCFO ou o nome criado pelo participante), adicione os campos e marque como “True” os que serão chave primária (em nosso exemplo CODCOLIGADA e CODCFO).



O próximo item a ser configurado é a correspondência entre os campos da origem e os campos do destino, isto é, um mapeamento para indicar ao sistema que o valor de um determinado campo da origem será atribuído a qual campo no destino.

Acesse a aba Correspondências e efetue as ligações entre os campos origem x destino ou clique nem Correspondências automáticas e o conceito os fará automaticamente considerando os critérios internos (podem ser alteradas/ajustadas pelo usuário).

## TOTVS Business Conector



Conceito - 03.9.0507

2/2 | Anexos | Processos

Identificação	Conexões Origem	Conexão Destino	Estrutura Origem	Estrutura Destino	Correspondências
(PK) qry_FCFO.CODCOLIGADA					(PK) FCFO.CODCOLIGADA
(PK) qry_FCFO.CODCFO					(PK) FCFO.CODCFO
qry_FCFO.NOME					FCFO.NOME
qry_FCFO.PAGREC					FCFO.PAGREC
qry_FCFO.ATIVO					FCFO.ATIVO
qry_FCFO.PESSOAFISOUJUR					FCFO.PESSOAFISOUJUR
qry_FCFO.CGCCFO					FCFO.CGCCFO
qry_FCFO.CIDADE					FCFO.CIDADE
qry_FCFO.TELEFONE					FCFO.TELEFONE
qry_FCFO.IDCFO					FCFO.IDCFO
qry_FCFO.TIPOCONTRIBUINTEINSS					FCFO.TIPOCONTRIBUINTEINSS

Nova | Excluir | Limpar | Correspondências Automáticas

Operações

OK | Cancelar | Salvar

Após proceder com as correspondências e estar seguro que as demais configurações/parametrizações estão corretas (podem ser ajustadas a qualquer momento) basta salvar o conceito e em seguida executá-lo.

Integração | Ferramentas | Customização | Gestão | Ambiente

Conceito - 03.9.0507 - TOTVS Série T Construção e F

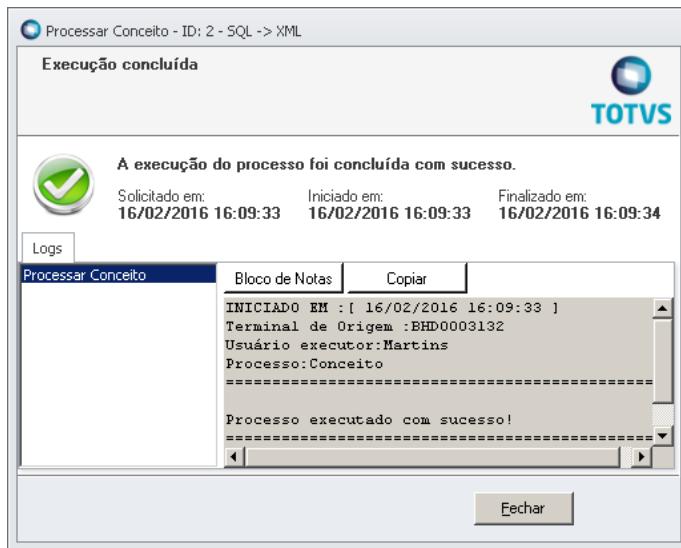
Página Inicial - 03.9.0507 | Conceito - 03.9.0507

Layout: Padrão | Arraste aqui o cabeçalho de uma coluna para agrupar

[x]	Novo	Id. Conceito	Nome	Última Exec.
<input type="checkbox"/>	<input type="checkbox"/>	1	Conceito Geral	15/02/201
<input checked="" type="checkbox"/>	<input type="checkbox"/>	2	SQL -> XML	16/02/201

Processos [Filtro: Todos]

- Processar conceito
- Processar novo conceito
- Exportar conceito
- Importar conceito
- Assistente de criação de conceito
- Fórmula Visual
- Geração de Planilhas em Lote
- Cópia da configuração da visão



E como resultado temos o arquivo XML com os registros exportados

```
C:\Public\TBC\Docs\saídaFCFO.xml - Notepad++ [Administrator]
Arquivo Editar Localizar Visualizar Formatar Linguagens Configurações Macro Executar Plugins Janelas ?
File Open saídaFCFO.xml
1 <?xml version="1.0" encoding="utf-8"?>
2 <CODCOLIGADA>
3   <FCFO>
4     <CODCOLIGADA>1</CODCOLIGADA>
5     <CODCFO>C00000</CODCFO>
6     <NOME>JORNAL FOLHA DE SÃO PAULO</NOME>
7     <PAGREC>3</PAGREC>
8     <ATIVO>1</ATIVO>
9     <PESSOAFLISOUJUR>J</PESSOAFLISOUJUR>
10    <CGCCFO>21.719.356/0001-50</CGCCFO>
11    <CIDADE>Manaus</CIDADE>
12    <TELEFONE>3215-8475</TELEFONE>
13    <IDCFO>7</IDCFO>
14    <TIPOCONTRIBUINTEINSS>0</TIPOCONTRIBUINTEINSS>
15  </FCFO>
16 </FCFO>
17   <FCFO>
18     <CODCOLIGADA>1</CODCOLIGADA>
19     <CODCFO>C00001</CODCFO>
20     <NOME>MÓVEIS MODULAR LTDA</NOME>
21     <PAGREC>1</PAGREC>
22     <ATIVO>1</ATIVO>
23     <PESSOAFLISOUJUR>K</PESSOAFLISOUJUR>
24     <CGCCFO>17.157.686/0001-21</CGCCFO>
25     <CIDADE>Belo Horizonte</CIDADE>
26     <TELEFONE>222-2010</TELEFONE>
27     <IDCFO>8</IDCFO>
28     <TIPOCONTRIBUINTEINSS>0</TIPOCONTRIBUINTEINSS>
29   </FCFO>
30   <FCFO>
31     <CODCOLIGADA>1</CODCOLIGADA>
32     <CODCFO>C00002</CODCFO>
33     <NOME>GRUPO BMG S/A</NOME>
34     <PAGREC>3</PAGREC>
35     <ATIVO>1</ATIVO>
36     <PESSOAFLISOUJUR>L</PESSOAFLISOUJUR>
37     <CGCCFO>22.678.197/0001-55</CGCCFO>
38     <CIDADE>Betim</CIDADE>
39     <TELEFONE>(31) 3334-4030</TELEFONE>
40     <IDCFO>9</IDCFO>
41     <TIPOCONTRIBUINTEINSS>0</TIPOCONTRIBUINTEINSS>
42 </FCFO>
```

### **Exercício**

Acesse o cadastro de Conceito e crie o seguinte registro:

#### **1. Origem Banco de Dados (SQL) → Destino Arquito XML**

Neste exercício, iremos exportar os dados da tabela de Funcionários (PFUNC) do Banco de Dados, utilizando um consulta SQL para um arquivo XML.

Campos: PFUNC.CODCOLIGADA (PK), PFUNC.CHAPA (PK), PSECAO.DESCRICAO SECAO, PFUNCAO.NOME FUNCAO, PFUNC.SALARIO.

**Observação:** Utilize o arquivo XSD “PFUNC” que se encontra dentro da pasta “Arquivos treinamento TBC\Novo Conceito”.

### **8.1.2. Origem arquivo XML e Destino banco de dados (tabela)**

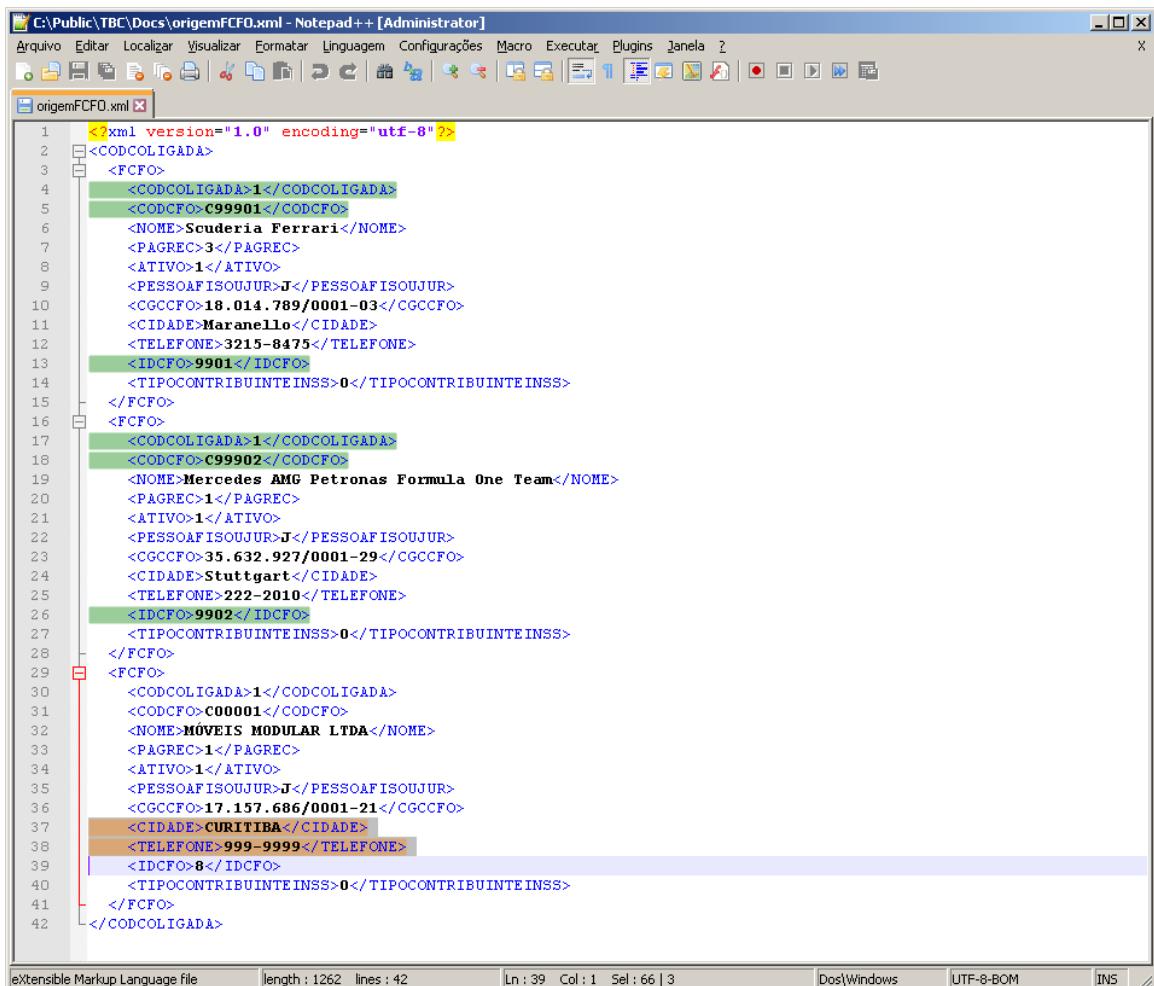
Através do Conceito é possível efetuar inclusões e/ou alterações diretamente em tabelas de uma base de dados. Neste contexto devemos nos atentar às regras (constraints, foreign Keys, domínios, etc.) da tabela e/ou campos destino.

No exemplo anterior (origem SQL e destino arquivo XML) geramos um arquivo XML sendo uma lista dos registros da tabela de cliente/fornecedor (através do SQL que elaboramos).

O exemplo a seguir mostra como, a partir de um arquivo XML, podemos inserir e atualizar registros em uma tabela da base do TOTVS RM (mas poderia ser uma tabela qualquer de uma base de dados qualquer). Por questões de facilidade e agilidade iremos aproveitar do exemplo anterior o XSD (schema), estrutura do arquivo XML gerado e alguns registros deste arquivo.

A tabela destino a ser adotada para este conceito, onde iremos incluir/atualizar registros, é a de cliente/fornecedor (FCFO).

A origem dos dados será um arquivo XML, o denominaremos “origemFCFO.xml” (ou outro nome de sua preferência), que conterá dois registros novos e um registro já existentes na base de dados (copiaremos do arquivo XML gerado anteriormente). A estrutura deste arquivo deverá estar de acordo com o XSD (podemos ver a estrutura do arquivo XML gerado anteriormente) e seu conteúdo será similar ao seguinte:



```

1 <?xml version="1.0" encoding="utf-8"?>
2 <CODCOLIGADA>
3   <FCFO>
4     <CODCOLIGADA>1</CODCOLIGADA>
5     <CODCFO>C99901</CODCFO>
6     <NOME>Scuderia Ferrari</NOME>
7     <PAGREC>3</PAGREC>
8     <ATIVO>1</ATIVO>
9     <PESSOAISOUJUR>J</PESSOAISOUJUR>
10    <CGCCFO>18.014.789/0001-03</CGCCFO>
11    <CIDADE>Maranello</CIDADE>
12    <TELEFONE>3215-8475</TELEFONE>
13    <IDCFO>9901</IDCFO>
14    <TIPOCONTRIBUINTEINSS>0</TIPOCONTRIBUINTEINSS>
15  </FCFO>
16  <FCFO>
17    <CODCOLIGADA>1</CODCOLIGADA>
18    <CODCFO>C99902</CODCFO>
19    <NOME>Mercedes AMG Petronas Formula One Team</NOME>
20    <PAGREC>1</PAGREC>
21    <ATIVO>1</ATIVO>
22    <PESSOAISOUJUR>J</PESSOAISOUJUR>
23    <CGCCFO>35.632.927/0001-29</CGCCFO>
24    <CIDADE>Stuttgart</CIDADE>
25    <TELEFONE>222-2010</TELEFONE>
26    <IDCFO>9902</IDCFO>
27    <TIPOCONTRIBUINTEINSS>0</TIPOCONTRIBUINTEINSS>
28  </FCFO>
29  <FCFO>
30    <CODCOLIGADA>1</CODCOLIGADA>
31    <CODCFO>C00001</CODCFO>
32    <NOME>MÓVEIS MODULAR LTDA</NOME>
33    <PAGREC>1</PAGREC>
34    <ATIVO>1</ATIVO>
35    <PESSOAISOUJUR>J</PESSOAISOUJUR>
36    <CGCCFO>17.157.686/0001-21</CGCCFO>
37    <CIDADE>CURITIBA</CIDADE>
38    <TELEFONE>999-9999</TELEFONE>
39    <IDCFO>8</IDCFO>
40    <TIPOCONTRIBUINTEINSS>0</TIPOCONTRIBUINTEINSS>
41  </FCFO>
42 </CODCOLIGADA>

```

extensible Markup Language file length : 1262 lines : 42 Ln : 39 Col : 1 Sel : 66 | 3 Dos\Windows UTF-8-BOM IN5

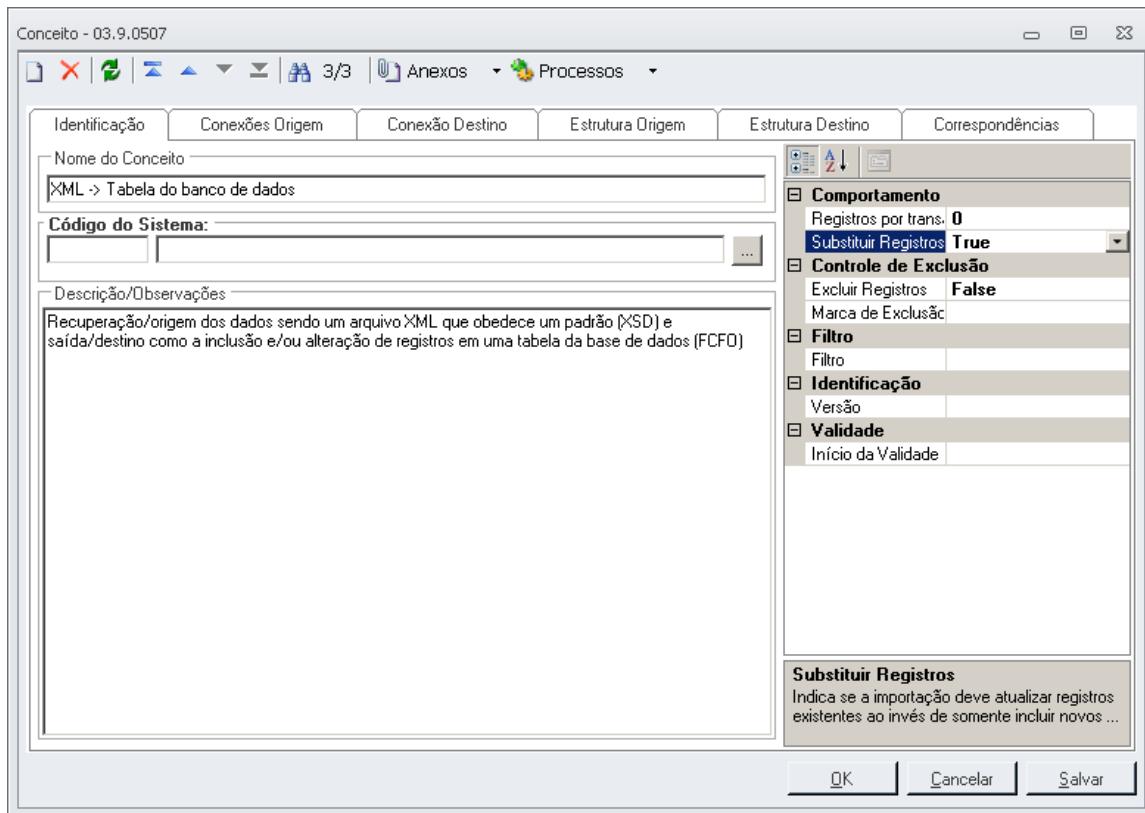
## ATENÇÃO:

1. Para os novos registros, os dois primeiros, atentem-se para os valores da chave primária CODCOLIGADA; CODCFO pois tais pares de valores devem ser chaves que ainda não existam na tabela, caso contrário o conceito entenderá que é para atualizar o registro correspondente a tal chave primária;
2. Para os novos registros, os dois primeiros, atentem-se para o valor do campo IDCFO pois se trata de, na tabela FCFO, um campo com constraint “Unique” então seu valor não poderá ser um que já exista entre os registros da tabela;
3. Para o registro a ser atualizado, o terceiro, a chave primária CODCOLIGADA; CODCFO deve conter valores do registro que se deseja atualizar;
4. Para o registro a ser atualizado, o terceiro, os campos que alteramos, e que serão atualizados, na tabela serão CIDADE e o TELEFONE.

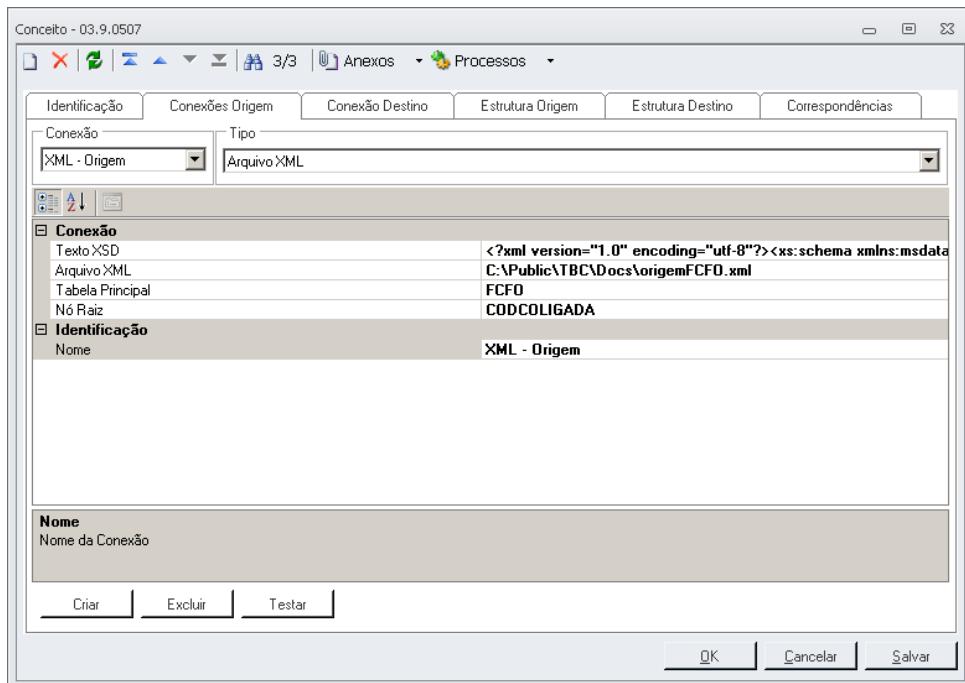
Após estas definições vamos a criação do conceito:

- Identificação
  - Nome: XML -> Tabela do banco de dados (ou a gosto do participante);
  - Descrição: Recuperação/origem dos dados sendo um arquivo XML que obedece um padrão (XSD) e saída/destino como a inclusão e/ou alteração de registros em uma tabela da base de dados (FCFO) (ou a gosto do participante);
  - Substituir Registros (propriedade do conceito apresentado no lado direito da interface)

Como neste exemplo iremos incluir e atualizar registros esta propriedade deve ser alterada (uma vez que o default é “False” para “True”;

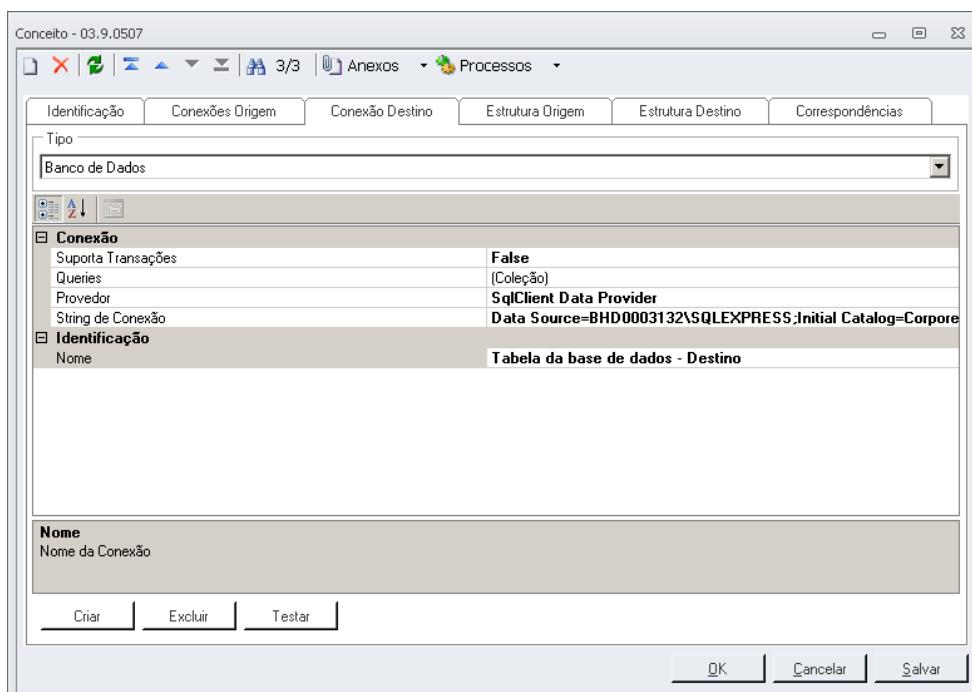


- **Conexão Origem**
  - Nome XML – Origem (ou a gosto do participante);
  - Tipo: Arquivo XML;
  - Texto XSD: Informe o texto schema no formato XSD. Ao clicar no lookup para informar o texto XSD é aberta uma visão na qual é possível digitar o texto schema, ou abrir o arquivo XSD para que o texto seja carregado;
  - Arquivo XML: Informe o caminho completo e o nome do arquivo XML;
  - Tabela Principal: Selecione a tabela principal, apenas para conexão de origem;
  - Nô Raiz: Caso a consulta de origem tenha como retorno múltiplo registros, informe neste campo o nó raiz. Em nosso exemplo será CODCOLIGADA;



- Conexão Destino

- Nome: SQL – Destino (ou a gosto do participante);
- Tipo: Banco de dados;
- Provedor: SqlClient Data Provider;
- String de Conexão: Montar a string de conexão desejada (informar o servidor e banco de dados origem conforme realizado em exemplos/prints anteriores).

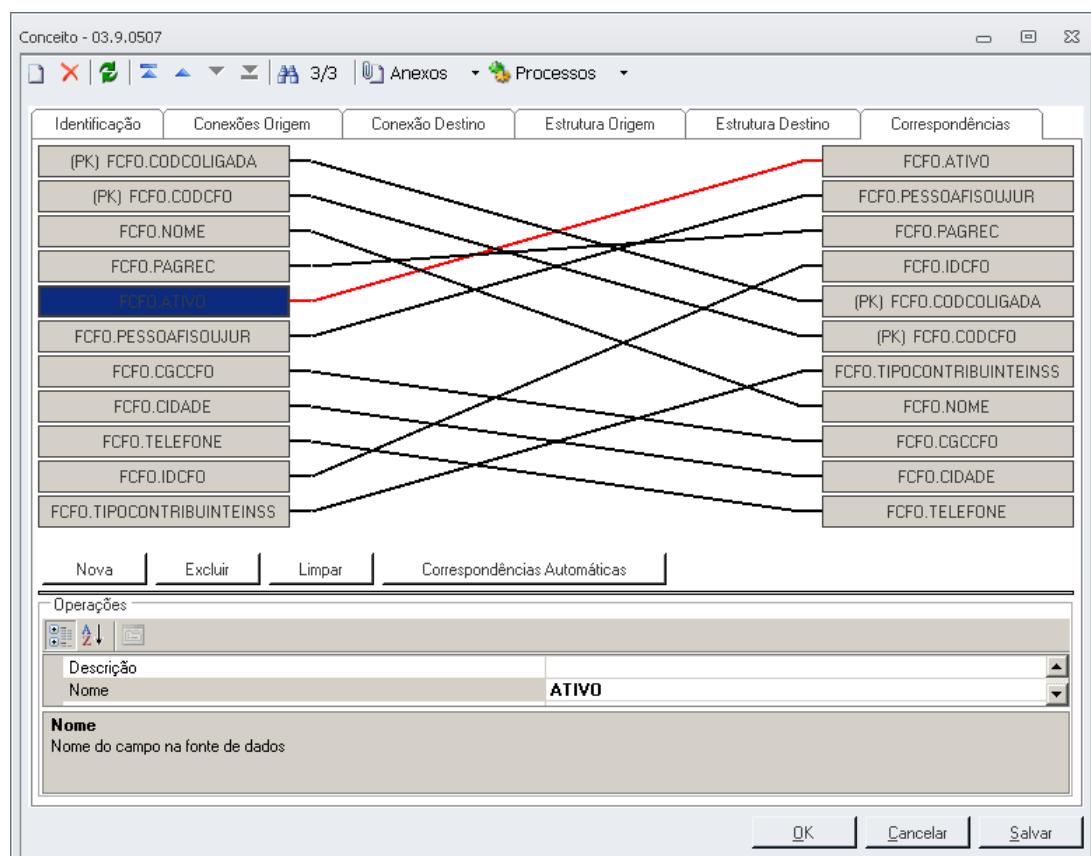


Os passos para montar a Estrutura Origem e Estrutura Destino são semelhantes.

Na Estrutura Origem selecione em Tabelas disponíveis o nome da estrutura do XSD/XML que em nosso exemplo é FCFO, adicione os campos e informe os que serão considerados como chave primária (CODCOLIGADA e CODCFO).

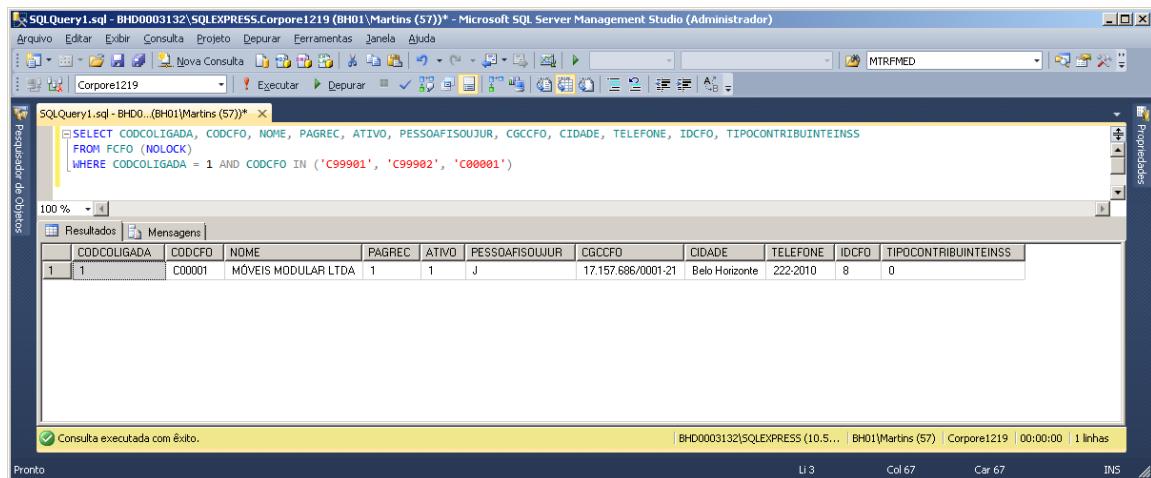
Na Estrutura Destino selecione em Tabelas disponíveis (estarão listadas todas as tabelas da base de dados configurada como destino) a tabela para qual deseja incluir/atualizar os registros, que em nosso exemplo é a tabela de cliente/fornecedor - FCFO, adicione os campos e informe os que são chave primária (CODCOLIGADA e CODCFO).

Na aba Correspondências e efetue os relacionamentos entre os campos origem x destino (ou utilize o recurso de Correspondências automáticas).



Salve o conceito e execute-o (executar o processo com tal conceito marcado – quando pela visão de conceitos – ou executar o processo direto da interface de edição do conceito que acabou de salvar).

Abaixo temos a tabela de cliente/fornecedor (FCFO) antes da execução do conceito. Observe que não existe os registros novos (que se encontram no arquivo XML de origem) e observe os valores para os campos CIDADE e TELEFONE do registro que existe e deverá ser atualizado.



SQLQuery1.sql - BH0003132\SQLEXPRESS.Corpore1219 (BH01\Martins (57)) - Microsoft SQL Server Management Studio (Administrador)

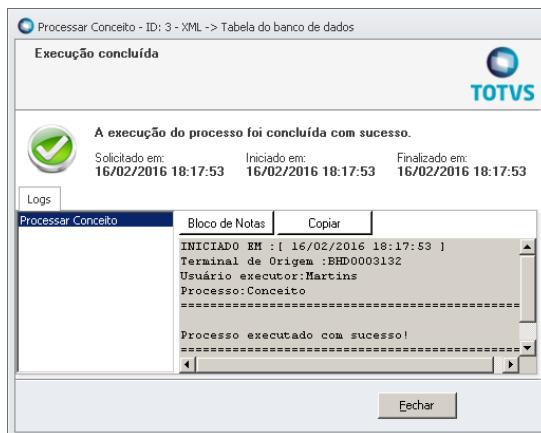
```
SELECT CODCOLIGADA, CODFCO, NOME, PAGREC, ATIVO, PESSOAISOUJUR, CGCCFO, CIDADE, TELEFONE, IDCFO, TIPOCONTRIBUINTEINSS
FROM FCF0 (NOLOCK)
WHERE CODCOLIGADA = 1 AND CODFCO IN ('C99901', 'C99902', 'C00001')
```

Resultados

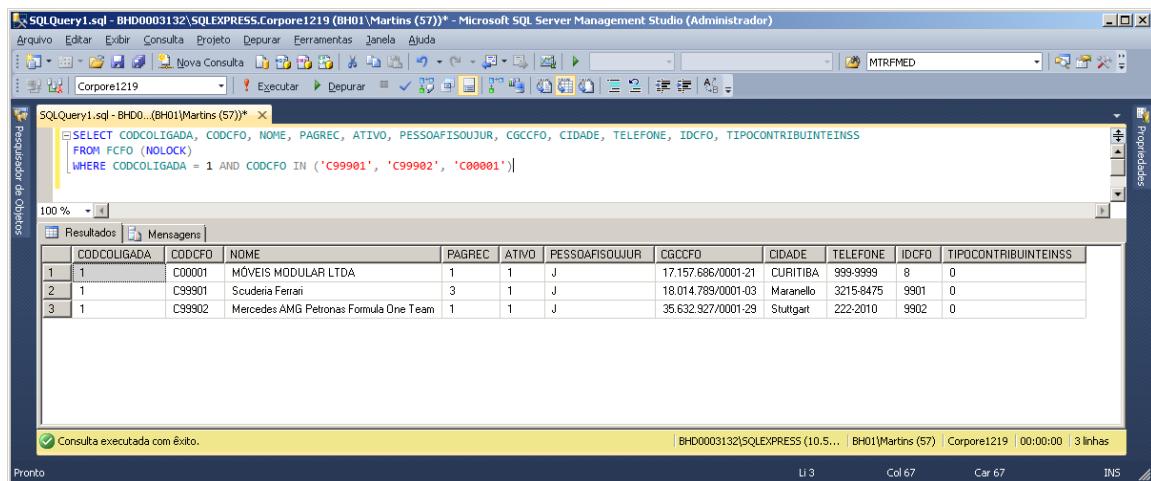
CODCOLIGADA	CODFCO	NOME	PAGREC	ATIVO	PESSOAISOUJUR	CGCCFO	CIDADE	TELEFONE	IDCFO	TIPOCONTRIBUINTEINSS
1	C00001	MÓVEIS MODULAR LTDA	1	1	J	17.157.686/0001-21	Belo Horizonte	222-2010	8	0

Consulta executada com êxito.

## Execução do conceito



Abaixo tempo a tabela de cliente/fornecedor (FCFO) após a execução do conceito (com a mesma consulta executada antes de iniciar o processo). Observe agora que os registros novos (que se encontravam no arquivo XML de origem) foram inseridos e também que os valores dos campos CIDADE e TELEFONE do registro que já existia foram alterados conforme preenchemos no arquivo XML de origem.



SQLQuery1.sql - BH0003132\SQLEXPRESS.Corpore1219 (BH01\Martins (57)) - Microsoft SQL Server Management Studio (Administrador)

```
SELECT CODCOLIGADA, CODFCO, NOME, PAGREC, ATIVO, PESSOAISOUJUR, CGCCFO, CIDADE, TELEFONE, IDCFO, TIPOCONTRIBUINTEINSS
FROM FCF0 (NOLOCK)
WHERE CODCOLIGADA = 1 AND CODFCO IN ('C99901', 'C99902', 'C00001')
```

Resultados

CODCOLIGADA	CODFCO	NOME	PAGREC	ATIVO	PESSOAISOUJUR	CGCCFO	CIDADE	TELEFONE	IDCFO	TIPOCONTRIBUINTEINSS
1	C00001	MÓVEIS MODULAR LTDA	1	1	J	17.157.686/0001-21	CURITIBA	999-9999	8	0
2	C99901	Scuderia Ferrari	3	1	J	18.014.789/0001-03	Maranello	3215-8475	9901	0
3	C99902	Mercedes AMG Petronas Formula One Team	1	1	J	35.632.927/0001-29	Stuttgart	222-2010	9902	0

Consulta executada com êxito.

### Exercício

Acesse o cadastro de Conceito e crie o seguinte registro:

#### 1. Origem Arquivo XML → Destino Banco de Dados

Neste exemplo, vamos importar de um arquivo XML registros de novas Disciplinas para o Banco de Dados.

Campos: CODCOLIGADA (PK), CODDISC (PK), NOME, NOMEREDUZIDO, COMPLEMENTO, ESTAGIO, CURSOLIVRE, CODTIPOCURSO.

**Observação:** Utilize o arquivo XSD “SDISCIPLINA” que se encontra dentro da pasta “Arquivos treinamento TBC\Novo Conceito”.

## 8.2. Conceito com DataServer

Os DataServers são os provedores de dados e serviços da arquitetura do TOTVS RM. São neles que estão todas as regras de negócio para a recuperação e gravação dos dados referentes às entidades (cadastros e processos) dos módulos do TOTVS RM.

O uso de DataServers do TOTVS RM tanto como origem como destino (principalmente neste último) implica que os registros a serem processados serão submetidos as mesmas regras (valores default, validações, consistências, críticas, etc.) observadas pelas funcionalidades equivalentes (cadastros, processos) dos módulos do TOTVS RM. Então temos de nos atentar quanto ao uso de DataServers combinados com inclusão/alteração de registros.

### 8.2.1. Origem DataServer e Destino arquivo texto

Como exemplo do uso de DataServers do TOTVS RM apresentaremos um conceito onde recuperaremos os produtos, através do DataServer de produto, e como saída geraremos uma listagem com os produtos retornados pelo DataServer.

Para a criação do conceito temos as seguintes características (valores dos atributos):

- Identificação
  - Nome: DataServer -> Arquivo texto (ou a gosto do participante);
  - Descrição: Recuperação/origem dos dados sendo um DataServer do TOTVS RM (DataServer de produto EstPrdDataBR) e saída/destino como um arquivo texto com a listagem dos produtos retornados pelo DataServer;
- Conexão Origem
  - Nome: DataServer – Origem (ou a gosto do participante);
  - Tipo: DataServer;
  - Segue abaixo a lista das propriedades disponíveis para a conexão bem como seus significados. Na sequência os valores das propriedades da conexão de nosso exemplo.

- **Conexão Default**

Informe se o Data Server irá utilizar a conexão corrente. Caso contrário, os campos Usuário CORPORE, Senha do CORPORE e Alias CORPORE devem ser preenchidos;

- **Usuário CORPORE**

Informe o código do usuário a ser utilizado ao acessar o DataServer;

- **Senha do CORPORE**

Informe a senha de autenticação correspondente ao código do usuário informado na propriedade Usuário CORPORE;

- **Data Server**

Informe o nome do DataServer a ser utilizado pela conexão

- **Tabela Principal**

Informe o nome da tabela do DataServer que servirá de base para a importação

- **Alias CORPORE**

Esta propriedade, através do alias aqui informado, indica a qual base de dados/instancia o conceito se conectará para acesso aos dados. O alias aqui informado deverá ser um dos alias do arquivo alias.dat

- **Coligada**

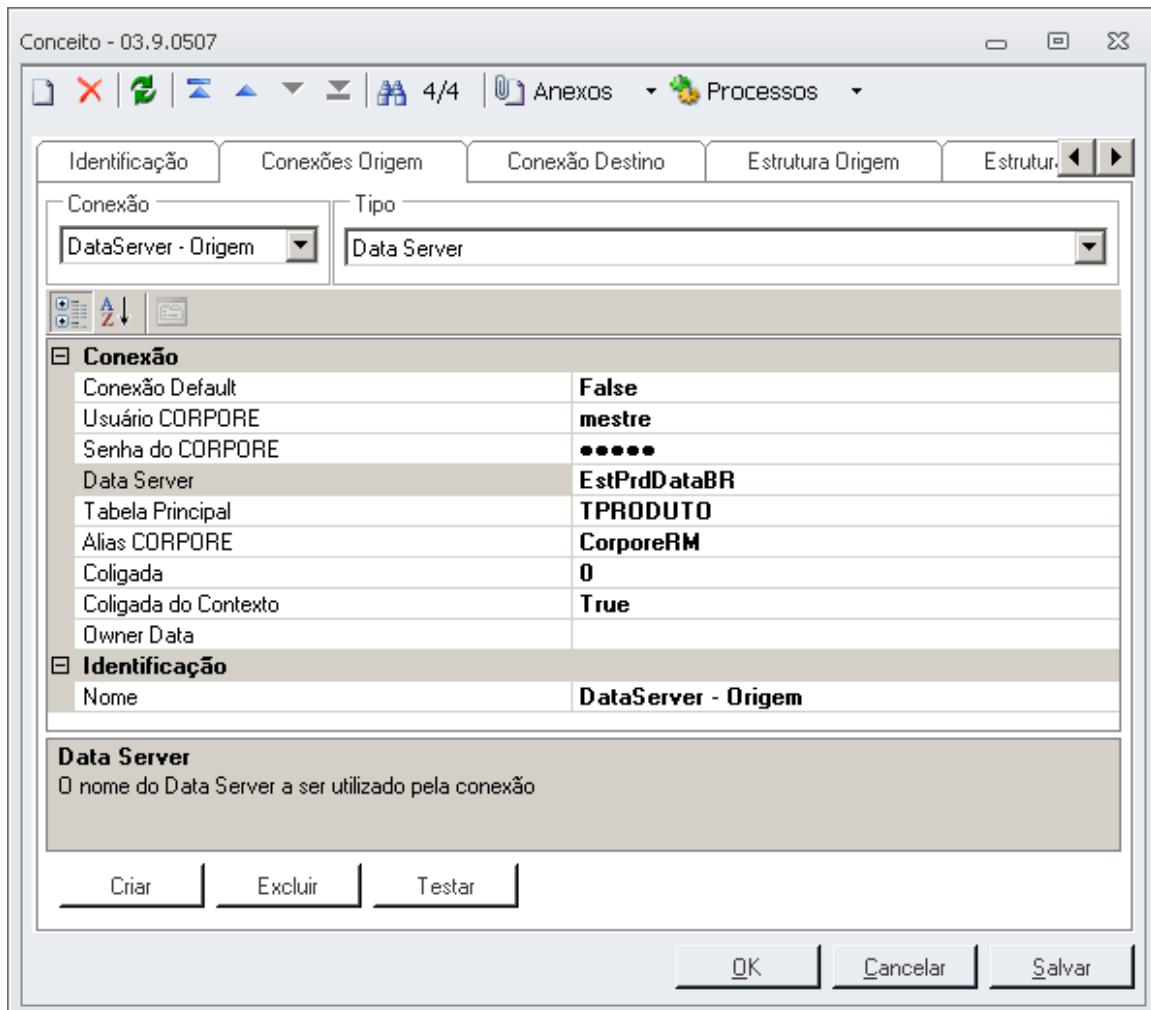
Informe a coligada que será considerada para a conexão e acionamento do DataServer

- **Coligada do contexto**

Indicativo se a coligada utilizada para a conexão e processamento do conceito será a coligada do contexto (o qual se está criando o conceito);

- **Owner Data**

Alguns DataServer necessitam de dados específicos para seus procedimentos de recuperação/gravação de dados bem como controles internos. A estes dados específicos denominamos OwnerData e, em geral, não é obrigatório bem como é específico para cada DataServer.



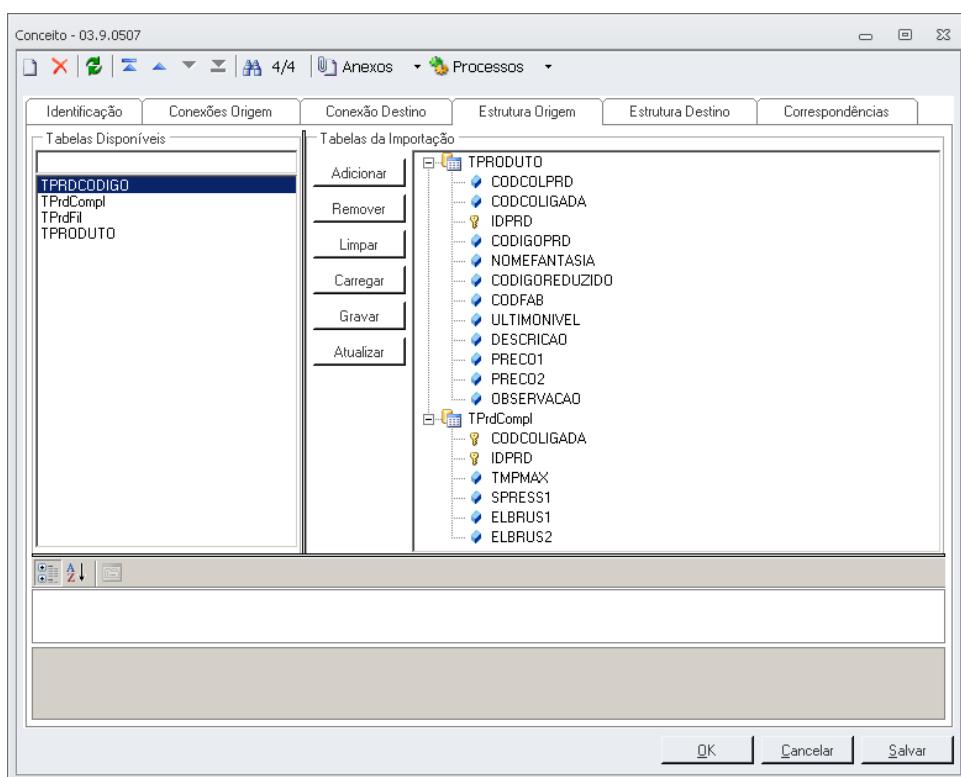
- Conexão Destino
  - Nome Arquivo texto - Destino (ou a gosto do participante);
  - Tipo: Arquivo texto;
  - Propriedades da conexão:
    - Tipo: Delimitado;
    - Delimitador de campos: “|” – pipe (ou a gosto do participante);
    - Tabelas:
      - Identificação e Nome: PRODUTO
      - Campos:
        - CODCOLIGADA
        - IDPRD
        - CODIGOPRD
        - NOMEFANTASIA
        - CODIGOREDUZIDO
        - DESCRICAO
        - PRECO1
        - OBSERVAÇÃO
        - CCTMPMAX
        - CCSPRESS1
        - CCELBRUS2

- Arquivo: C:\Public\TBC\Docs\saidoPRODUTOS.txt (ou a gosto do participante);

- Estrutura Origem

Os dataservers podem trabalhar com várias tabelas/entidades afim de prover todas as informações necessárias para os usuários e/ou seus objetivos (processos, inclusão/alteração/exclusão, mestre-detalhe, etc.). Ao utilizar um DataServer como conexão no conceito o mesmo nos retornará (no campo “Tabelas disponíveis”) uma lista com várias entidades estruturadas (semelhante a tabelas e seus respectivos campos) providas por ele cabendo ao usuário identificar e configurar/parametrizar como Estrutura Origem e/ou Destino aquelas a serem utilizadas no conceito.

Em nosso exemplo utilizaremos as tabelas TPRODUTO (produto) e TPRDCOMPL (campos complementares do produto). Adicione os campos desejados conforme a imagem abaixo e lembre-se de assinalar como “True” (no quadro de propriedades das tabelas/campos apresentada abaixo da árvore de tabela/campos selecionados “Tabelas da importação”) os campos que são a chave primária das tabelas CODCOLIGADA e IDPRD.



- Estrutura Destino

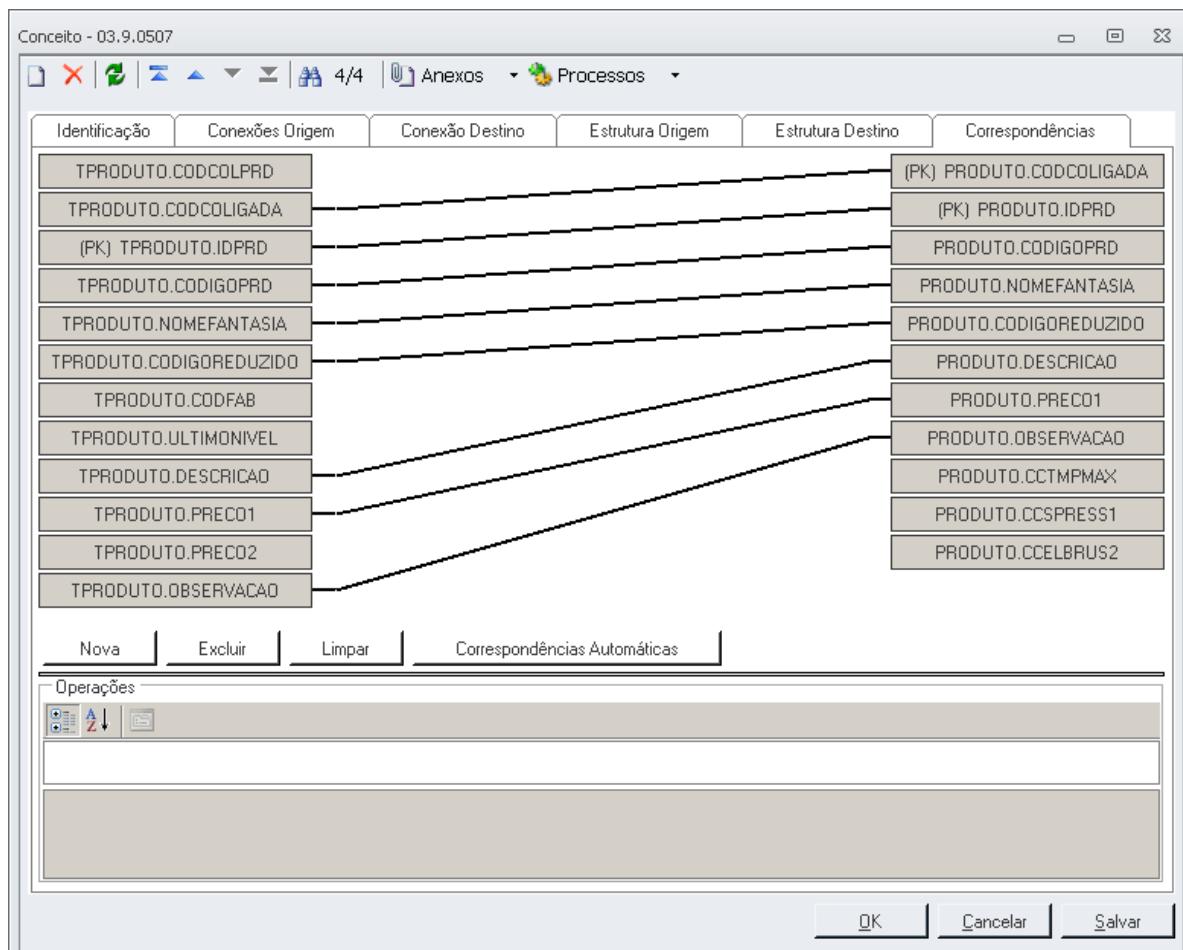
Na estrutura destino informaremos a “tabela” e todos os campos que cadastramos quando realizamos a configuração da conexão destino. Em tabelas disponíveis selecione nosso arquivo texto (PRODUTO ou o nome criado pelo participante), adicione os campos e marque como “True” os que serão chave primária (em nosso exemplo CODCOLIGADA e IDPRD).

- Correspondências

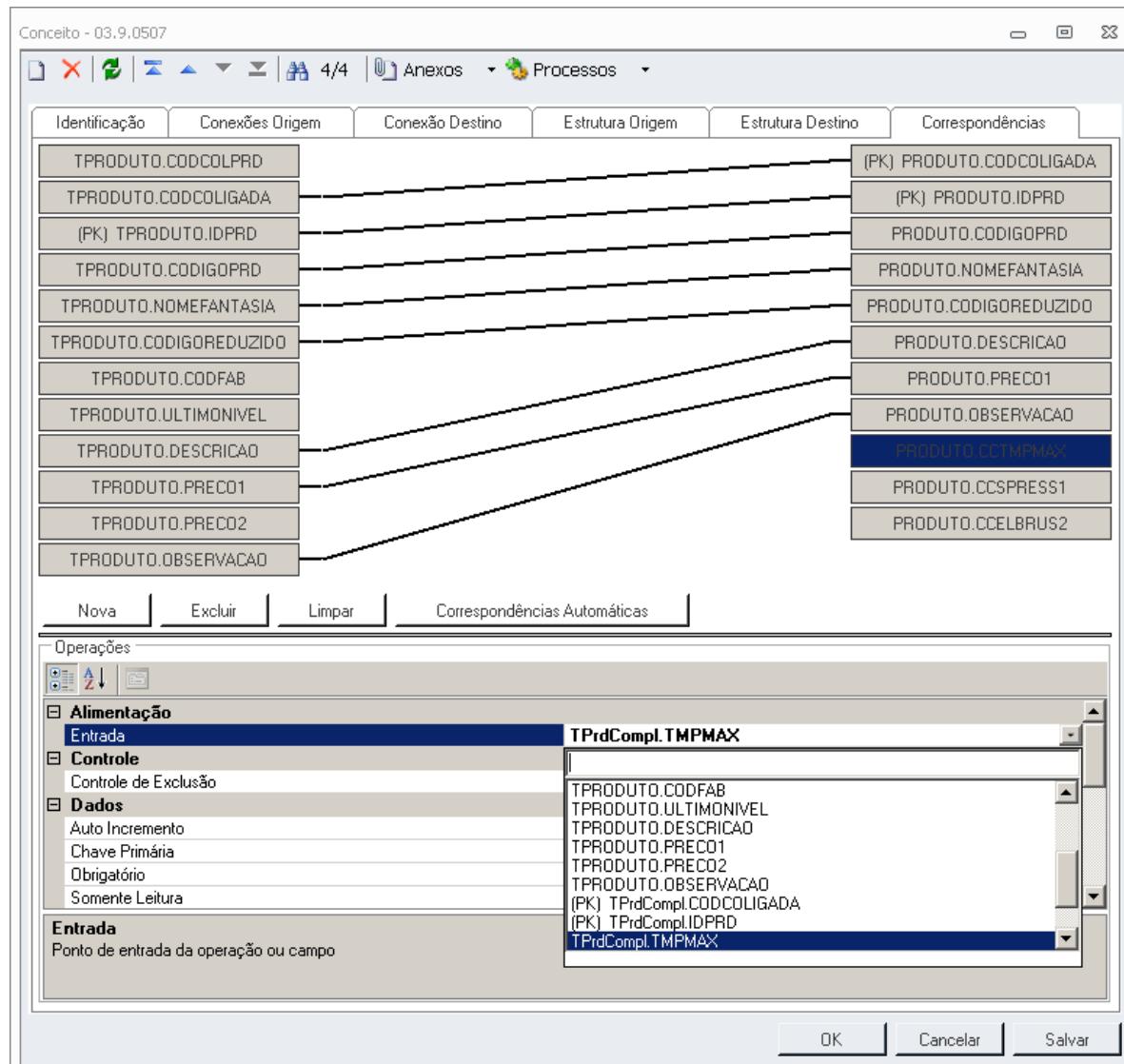
Neste exemplo no depararemos com uma particularidade na funcionalidade de Correspondências: observe que foi selecionado, na Estrutura Origem, a tabela de campos complementares do produto (TPRDCOMPL) e alguns de seus campos, já na lista de tabelas/campos para se efetuar a correspondência tais campos (não são apresentados). Apesar disto não há comprometimento ao utilizar e realizar as correspondências considerando tais campos, isto é, apenas a forma de relacionar um campo destino com um campo origem (que não aparece

visualmente na interface) é outro, que não o relacionamento direto/gráfico (ele passa a ser feito nas propriedades do campo destino).

Observe, na estrutura a direita (destino), que temos três campos criado no arquivo texto para receber os valores provenientes de campos complementares da origem e estes campos não possuem a “ligação” gráfica com campos da origem pois os campos que desejamos são aqueles que não aparecem na estrutura da esquerda (origem).



A correspondência para estes tipos de campos, que devem ser valorados a partir de um campo que não aparece na estrutura origem, é setada nas propriedades do campo. Selecione o campo destino e, na lista de propriedades que é apresentada em Operações (controle logo abaixo da área de efetuar os relacionamentos), procure pela propriedade de nome Entrada (no grupo Alimentação). Observe que você poderá valorar esta propriedade a partir de uma lista e nesta lista estarão todos os campos da origem, mesmo os que não são apresentados na interface de relacionamento da aba de Correspondências.



Após finalizar todas as configurações, correspondências entre campos e informar, no destino, os serão as chaves primárias no destino, basta apenas salvar o conceito e executá-lo.

## TOTVS Business Conector



The screenshot shows the TOTVS Business Conector application window. The menu bar includes 'Sistema', 'Conceito - 03.9.0507 - TOTVS Série T Construção e Pr'. Below the menu bar is a toolbar with icons for 'Integração', 'Ferramentas', 'Customização', 'Gestão', and 'Ambiente'. A secondary toolbar below it has icons for 'Conceito', 'Serviços', 'Monitor da Fila de Integração RM Conector', 'Gerador de Saída', and 'Mensagem Única'. The main area displays a table titled 'Conceito - 03.9.0507' with columns for 'Novo', 'Id. Conceito', 'Nome', and 'Última Exec.'. Five rows are listed, with the fifth row selected. A context menu is open over this row, showing options: 'Processar conceito' (highlighted in blue), 'Processar novo conceito', 'Exportar conceito', 'Importar conceito', 'Assistente de criação de conceito', 'Fórmula Visual', 'Geração de Planilhas em Lote', and 'Cópia da configuração da visão'.

### Execução do Conceito

The screenshot shows a dialog box titled 'Processar Conceito - ID: 5 - DataServer -> Arquivo texto'. The main message is 'Execução concluída' (Execution completed) with a green checkmark icon. Below it, a message states 'A execução do processo foi concluída com sucesso.' (The process execution was completed successfully). It shows the execution details: 'Solicitado em: 22/02/2016 10:04:23', 'Iniciado em: 22/02/2016 10:04:24', and 'Finalizado em: 22/02/2016 10:04:54'. A 'Logs' tab is visible, showing the execution log in a text area:

```
INICIADO EM :[ 22/02/2016 10:04:24 ]
Terminal de Origem :BHD0003132
Usuário executor:Martins
Processo:Conceito
=====
Processo executado com sucesso!
```

A 'Fechar' (Close) button is at the bottom right of the dialog.

E:\Public\TBC\Docs\saidoPRODUTOS.txt - Notepad++ [Administrator]

Arquivo Editar Localizar Visualizar Formatar Linguagens Configurações Macro Executar Plugins Janela ?

new 1 saidoPRODUTOS.txt

```

1 PRODUTO|1|1|01|CD's|0001|CD's|0,0000|||||1
2 PRODUTO|1|10|02.04|Mini-Formulário Salmon|0010|Formulário MiniForm Salmon|6,3700|||||1
3 PRODUTO|1|100|09.68|Porca sext.galv. 1.1/4 G.W.W.|0132|Porca sext.galv. 1.1/4 G.W.W.|0,0500|||||1
4 PRODUTO|1|101|09.69|Tela xadrez com fios galvanizados|0134|Tela xadrez com fios galvanizados|12,0000|||||1
5 PRODUTO|1|102|09.70|Tábua de Madeira 1.40 cm|0135|Chapa dobrada|2,2000|||||1
6 PRODUTO|1|103|09.71|Toco de Madeira 30 x 8 cm|0136|Suporte de borracha para pés da cama|1,2500|||||1
7 PRODUTO|1|104|09.72|Ponteira dourada 1.1/4|0137|Ponteira dourada 1.1/4|1,8000|||||1
8 PRODUTO|1|105|09.73|Póntua para estrado 1.30cm|0138|Madeira para estrado 1.30cm|4,0000|||||1
9 PRODUTO|1|106|09.74|Madeira para estrado 1.80cm|0139|Madeira para estrado 1.30cm|6,0000|||||1
10 PRODUTO|1|107|09.75|Arame para Solda|0140|Arame para solda|3,0000|||||1
11 PRODUTO|1|108|09.76|Tinta Cinza Claro|0141|Tinta Cinza Claro|17,0000|||||1
12 PRODUTO|1|109|09.77|Tinta Azul Del Rey|0142|Tinta Azul Del Rey|17,0000|||||1
13 PRODUTO|1|11|03|Etiquetas|0011|Etiquetas|0,0100|||||1
14 PRODUTO|1|110|09.78|Tinta verde|0143|Tinta vermelha|17,0000|||||1
15 PRODUTO|1|11|08|Cabeceira Completa|0144|Cabeceira completa|80,0000|||||1
16 PRODUTO|1|112|07.16|Lateral esquerda|0152|lateral esquerda|20,0000|||||1
17 PRODUTO|1|113|07.17|tubo para lateral esquerda|0153|tubo para lateral esquerda da cama|15,0000|||||1
18 PRODUTO|1|114|07.18|Pé da cama completo|0155|Pé da cama completo|25,0000|||||1
19 PRODUTO|1|115|07.19|Pé da cama|0156|Pé da cama|16,0000|||||1
20 PRODUTO|1|116|07.20|Pés curvados|0157|Pés curvados|20,0000|||||1
21 PRODUTO|1|117|07.21|Tubo para pé de cama|0160|Tubo para pé de cama|12,0000|||||1
22 PRODUTO|1|118|07.22|Estrado para cama casal padrão completo|0161|Grade completa|35,0000|||||1
23 PRODUTO|1|119|07.23|Cama de solteiro calandrada|0164|Cama de solteiro calandrada|80,0000|||||1
24 PRODUTO|1|12|03.01|Etiqueta Mala Direta 89x23|0012|Etiqueta Mala Direta 89x23 2 Car Pacote de 6000 UnIDPRDades|19,1000|||||1
25 PRODUTO|1|120|07.24|Cama de solteiro Acabada|0165|Cama de solteiro Acabada|120,0000|||||1
26 PRODUTO|1|121|08.19|Cama de Solteiro cor Marfim|0167|Cama de Solteiro 1,90 x 0,80 cor Marfim|180,0000|||||1
27 PRODUTO|1|122|08.20|Cama de solteiro cor Azul Del- Rey|0168|Cama de solteiro 1,90 x 0,80 cor Azul Del- Rey|140,0000|||||1
28 PRODUTO|1|123|08.21|Cama de solteiro cor Vermelha|0169|Cama de solteiro 1,90 x 0,80 cor Vermelha|110,0000|||||1
29 PRODUTO|1|124|09.90|Serviços|0177|Serviços|0,0100|||||1
30 PRODUTO|1|125|07.25|Processador Pentium II - 400 MHZ|0178|Processador Pentium II - 400 MHZ|500,0000|||||1
31 PRODUTO|1|126|07.29|Kit Multimídia Completo|0182|Kit Multimídia Completo|195,0000|||||1
32 PRODUTO|1|127|07.30|Cabos de Força para Computadores|0183|Cabos de Força para Computadores|32,0000|||||1
33 PRODUTO|1|128|07.31|Caixa plástica para Gabinete|0184|Caixa plástica para Gabinete|8,0000|||||1
34 PRODUTO|1|129|07.32|Placa de Fax Modem - 33600|0185|Placa de Fax Modem - 33600|72,0000|||||1
35 PRODUTO|1|13|03.02|Etiqueta Adesiva 102x23|0013|Etiqueta Adesiva 102x23 3 Car 12000U|42,4399|||||1
36 PRODUTO|1|130|07.33|Caixa de Som Multimídia|0186|Caixa de Som Multimídia|32,0000|||||1
37 PRODUTO|1|131|07.34|Microfone|0187|Microfone|18,6999|||||1
38 PRODUTO|1|132|08.03|Tubo de Imagem|0188|Tubo de Imagem|87,2998|||||1
39 PRODUTO|1|133|08.04|Fusível|0189|Fusível|17,7500|||||1
40 PRODUTO|1|134|08.05|Placa de Circuito Integrado|0190|Placa de Circuito Integrado|36,0000|||||1
41 PRODUTO|1|135|08.06|Fita de Gravação|0191|Fita de Gravação|10,0000|||||1

```

Normal text file length: 41852 lines : 500 Ln:1 Col:1 Sel:0|0 DosWindows ANSI INS

### Exercício

Acesse o cadastro de Conceito e crie o seguinte registro:

#### 1. Origem Data Server → Destino Arquivo Texto

Neste exemplo, vamos gerar um arquivo texto a partir do Data Server do cadastro de Departamentos.

Campos: CODCOLIGADA (PK), CODFILIAL (PK), CODDEPARTAMENTO (PK), NOME, CONTATO, TELEFONE, ATIVO.

**Observação:** Salver o arquivo texto em: C:\TBC\Departamentos.txt

## 8.2.2. Origem arquivo XML e Destino DataServer

Os DataServers do TOTVS RM são codificados para prover, principalmente, os cadastros de registros através dos módulos do ERP. Isto significa que acessando-os temos acesso a todos os campos necessários para gerir tais entidades bem como nestes é que estão implementadas as regras de negócio mantendo a consistência dos registros sejam elas a nível de aplicação bem como banco de dados.

O uso de DataServers como destino em um conceito pode ser útil no sentido de automatizar algumas operações cuja massa de dados a se dar entrada no TOTVS RM é expressiva. Importação de registros, carga de dados, processamento de registros bancários, criação de lançamentos, criação de usuários, enfim, manipular sem ser pelas interfaces do TOTVS RM os registros das respectivas entidades e garantindo a integridade das informações.

Neste novo exemplo iremos processar um arquivo XML com uma lista de usuários incluindo-os no TOTVS RM utilizando o DataServer de usuários – GlbUsuarioData – o equivalente ao acessar o TOTVS RM | Serviços Globais | Segurança | Usuários. O arquivo XML origem foi montado com base em um XSD com apenas alguns campos do cadastro de usuários. O DataServer é o mencionado anteriormente e a tabela principal destino é a GUSUARIO.

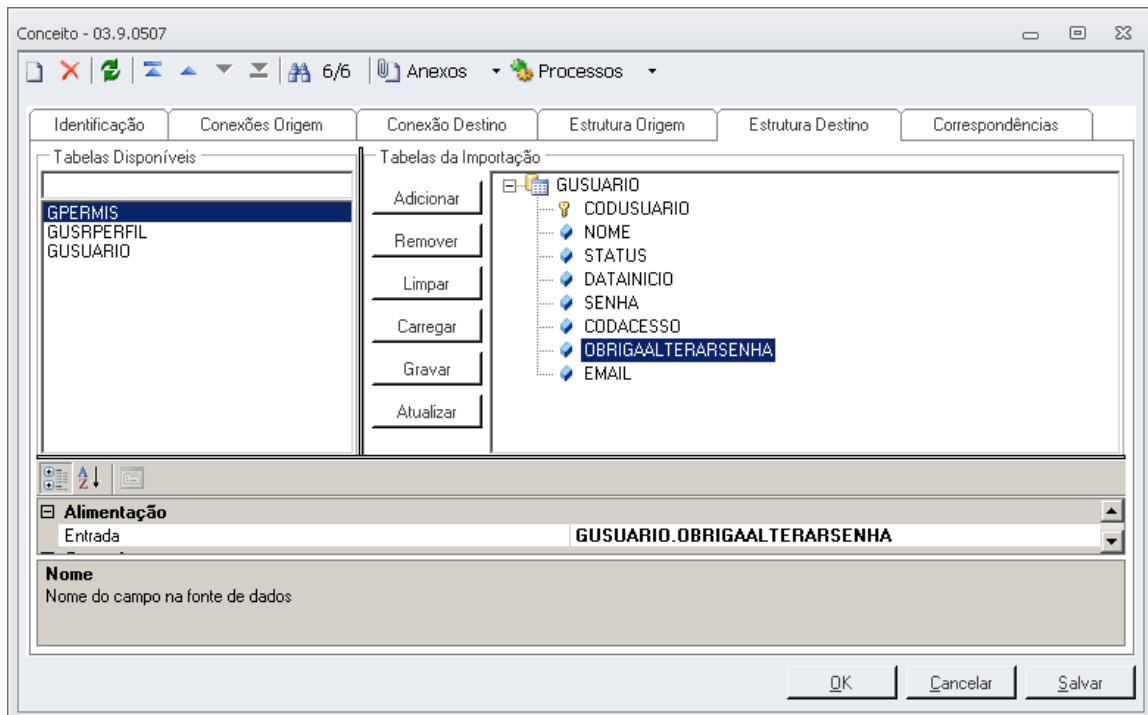
O XSD (Schema) de usuários é o que segue abaixo e você poderá notar que ele possui algumas novidades a começar pelo tipo de campos, determinados campos poderem ser NULL (`nillable="true"`) e também a definição da existência de algumas tags (`minOccurs="0" maxOccurs="1"`).

```
<?xml version="1.0" encoding="utf-8"?>
<xss:schema xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" xmlns:msprop="urn:schemas-microsoft-com:xml-msprop" xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xss:element name="GUSUARIO">
        <xss:complexType>
            <xss:all>
                <xs:element type="xs:string" minOccurs="0" maxOccurs="1" name="CODUSUARIO"></xs:element>
                <xs:element type="xs:string" minOccurs="0" maxOccurs="1" name="NOME"></xs:element>
                <xs:element type="xs:int" minOccurs="0" maxOccurs="1" name="STATUS"></xs:element>
                <xs:element type="xs:dateTime" minOccurs="0" maxOccurs="1" name="DATAINICIO"></xs:element>
                <xs:element type="xs:string" name="SENHA"></xs:element>
                <xs:element type="xs:string" name="CODACESSO"></xs:element>
                <xs:element type="xs:string" nillable="true" name="OBRIGAALTERARSENHA"></xs:element>
                <xs:element type="xs:string" nillable="true" name="EMAIL"></xs:element>
            </xss:all>
        </xss:complexType>
    </xss:element>
</xss:schema>
```

Vamos a criação do conceito e no decorrer das seções iremos apresentar novas características:

- Na aba Identificação informe um nome, descrição para o conceito e não se esqueça, uma vez que trabalharemos com inclusão/alteração de registros, se valorar para True a propriedade Substituir registros;
- Como Conexão Origem crie uma conexão do tipo Arquivo XML, informe o XSD, o caminho e nome de arquivo destino, como Tabela principal informe GUSUARIO (ou o utilizado no XSD) e como Nô raiz informe CODUSUARIO;
- Em Conexão Destino crie uma conexão do tipo DataServer e informe usuário, senha, o DataServer a ser utilizado como GlbUsuarioData, o alias para o banco de dados sendo CorporeRM (ou o a ser utilizado pelo participante) e marque como True a propriedade de Coligada do contexto;
- Na aba de Estrutura origem selecione, em Tabelas disponíveis, a “tabela” definida no arquivo XML (em nosso caso GUSUARIO) e adicione todos os campos. Lembre-se de marcar o campo CODUSUARIO como chave primária;

- Em Estrutura destino, na lista de Tabelas disponíveis (GPERMIS, GUSRPERFIL, GUSUARIO para este DataServer) encontre a tabela GUSUARIO, adicione esta e os campos a serem adicionados são os campos que desejamos incluir dados (CODUSUARIO, NOME, STATUS, DATAINICIO, SENHA, CODACESSO, OBRIGAALTERARSENHA, EMAIL) lembrando de marcar o campo CODUSUARIO como chave primária;



- Nas Correspondências, como optamos por criar os campos de origem (no XSD) tendo os mesmos nomes dos campos no destino, poderemos utilizar o recurso de Correspondências automáticas e deixar que a ferramenta efetue os relacionamentos (ligações, mapeamentos);
- Após efetuar as correspondências vamos editar algumas delas trabalhando com uma nova propriedade – Valor default – nos campos destino. Esta propriedade indica que se tal campo não estiver preenchido então o conceito utilizará o valor nela definido para valorar o campo no destino.  
*OBSERVAÇÃO: Você poderá notar que tais propriedades poderão ser informadas tanto na interface de Correspondências quanto na interface da estrutura Origem/Destino.*

## TOTVS Business Conector



Conceito - 03.9.0507

Identificação	Conexões Origem	Conexão Destino	Estrutura Origem	Estrutura Destino	Correspondências
(PK) GUSUARIO.CODUSUARIO		(PK) GUSUARIO.CODUSUARIO			
GUSUARIO.NOME		GUSUARIO.NOME			
GUSUARIO.STATUS		GUSUARIO.STATUS			
GUSUARIO.DATAINICIO		GUSUARIO.DATAINICIO			
GUSUARIO.SENHA		GUSUARIO.SENHA			
GUSUARIO.CODACESSO		GUSUARIO.CODACESSO			
GUSUARIO.OBRIGAALTERARSENHA		GUSUARIO.OBRIGAALTERARSENHA			
GUSUARIO.EMAIL		GUSUARIO.EMAIL			

Nova | Excluir | Limpar | Correspondências Automáticas

**Operações**

- Alimentação
- Controle
- Dados
- Identificação
- Relacionamento

**Alimentação**

Entrada: GUSUARIO.CODACESSO

**Controle**

Controle de Exclusão: False

**Dados**

Auto Incremento: False  
Chave Primária: False  
Obrigatório: False  
Somente Leitura: False  
Valor Default: Acesso01

**Identificação**

Alias: CODACESSO  
Descrição: CODACESSO  
Nome: CODACESSO  
Tabela: GUSUARIO

**Relacionamento**

Campo PK Referencia: GUSUARIO.CODACESSO  
Campo Referencia: GUSUARIO.CODACESSO  
Campo Tamanhario: False

**Valor Default**

O valor assumido pelo campo quando nenhum outro valor for atribuído

OK | Cancelar | Salvar

Conceito - 03.9.0507

Identificação	Conexões Origem	Conexão Destino	Estrutura Origem	Estrutura Destino	Correspondências
(PK) GUSUARIO.CODUSUARIO		(PK) GUSUARIO.CODUSUARIO			
GUSUARIO.NOME		GUSUARIO.NOME			
GUSUARIO.STATUS		GUSUARIO.STATUS			
GUSUARIO.DATAINICIO		GUSUARIO.DATAINICIO			
GUSUARIO.SENHA		GUSUARIO.SENHA			
GUSUARIO.CODACESSO		GUSUARIO.CODACESSO			
GUSUARIO.OBRIGAALTERARSENHA		GUSUARIO.OBRIGAALTERARSENHA			
GUSUARIO.EMAIL		GUSUARIO.EMAIL			

Nova | Excluir | Limpar | Correspondências Automáticas

**Operações**

- Alimentação
- Controle
- Dados
- Identificação
- Relacionamento

**Alimentação**

Entrada: GUSUARIO.OBRIGAALTERARSENHA

**Controle**

Controle de Exclusão: False

**Dados**

Auto Incremento: False  
Chave Primária: False  
Obrigatório: False  
Somente Leitura: False  
Valor Default: F

**Identificação**

Alias: OBRIGAALTERARSENHA  
Descrição: OBRIGAALTERARSENHA  
Nome: OBRIGAALTERARSENHA  
Tabela: GUSUARIO

**Relacionamento**

Campo PK Referencia: GUSUARIO.OBRIGAALTERARSENHA  
Campo Referencia: GUSUARIO.OBRIGAALTERARSENHA  
Campo Tamanhario: False

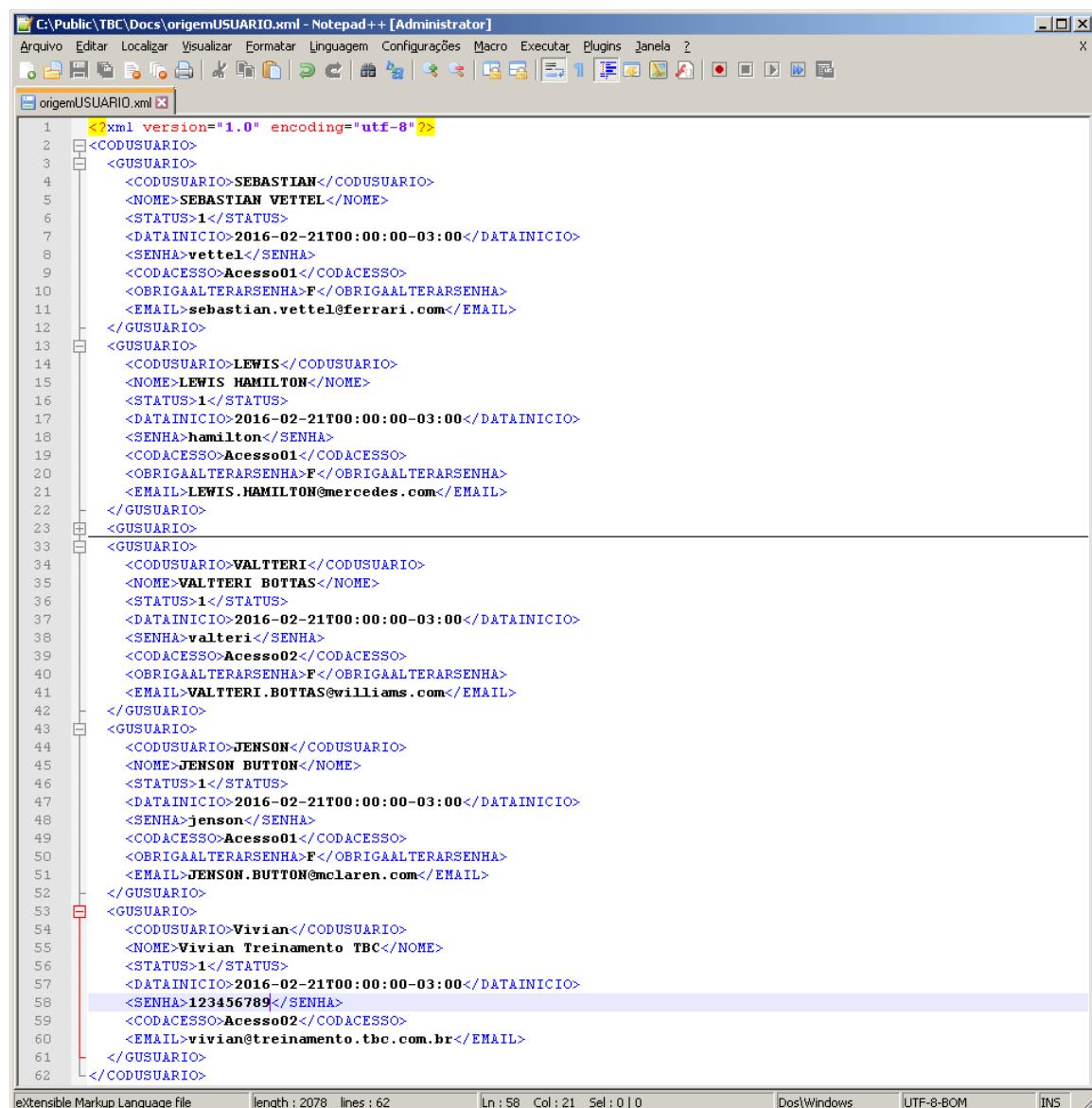
**Valor Default**

O valor assumido pelo campo quando nenhum outro valor for atribuído

OK | Cancelar | Salvar

Finalizada a parametrização do conceito basta gravá-lo e testá-lo.

Usuários a serem criados/alterados no arquivo XML de origem



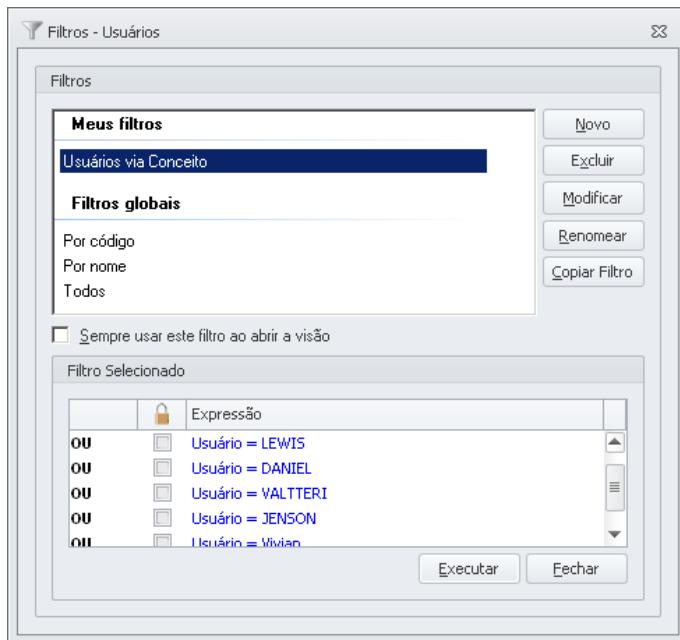
```

<?xml version="1.0" encoding="utf-8"?>
<CODUSUARIO>
    <GUSUARIO>
        <CODUSUARIO>SEBASTIAN</CODUSUARIO>
        <NOME>SEBASTIAN VETTEL</NOME>
        <STATUS>1</STATUS>
        <DATAINICIO>2016-02-21T00:00:00-03:00</DATAINICIO>
        <SENHA>vettel</SENHA>
        <CODACESSO>Acesso01</CODACESSO>
        <OBIGAALTERARSENHA>F</OBIGAALTERARSENHA>
        <EMAIL>sebastian.vettel@ferrari.com</EMAIL>
    </GUSUARIO>
    <GUSUARIO>
        <CODUSUARIO>LEWIS</CODUSUARIO>
        <NOME>LEWIS HAMILTON</NOME>
        <STATUS>1</STATUS>
        <DATAINICIO>2016-02-21T00:00:00-03:00</DATAINICIO>
        <SENHA>hamilton</SENHA>
        <CODACESSO>Acesso01</CODACESSO>
        <OBIGAALTERARSENHA>F</OBIGAALTERARSENHA>
        <EMAIL>LEWIS.HAMILTON@mercedes.com</EMAIL>
    </GUSUARIO>
    <GUSUARIO>
        <CODUSUARIO>VALTTERI</CODUSUARIO>
        <NOME>VALTTERI BOTTAS</NOME>
        <STATUS>1</STATUS>
        <DATAINICIO>2016-02-21T00:00:00-03:00</DATAINICIO>
        <SENHA>valteri</SENHA>
        <CODACESSO>Acesso02</CODACESSO>
        <OBIGAALTERARSENHA>F</OBIGAALTERARSENHA>
        <EMAIL>VALTTERI.BOTTAS@williams.com</EMAIL>
    </GUSUARIO>
    <GUSUARIO>
        <CODUSUARIO>JENSON</CODUSUARIO>
        <NOME>JENSON BUTTON</NOME>
        <STATUS>1</STATUS>
        <DATAINICIO>2016-02-21T00:00:00-03:00</DATAINICIO>
        <SENHA>jenson</SENHA>
        <CODACESSO>Acesso01</CODACESSO>
        <OBIGAALTERARSENHA>F</OBIGAALTERARSENHA>
        <EMAIL>JENSON.BUTTON@mclaren.com</EMAIL>
    </GUSUARIO>
    <GUSUARIO>
        <CODUSUARIO>Vivian</CODUSUARIO>
        <NOME>Vivian Treinamento TBC</NOME>
        <STATUS>1</STATUS>
        <DATAINICIO>2016-02-21T00:00:00-03:00</DATAINICIO>
        <SENHA>123456789</SENHA>
        <CODACESSO>Acesso02</CODACESSO>
        <EMAIL>vivian@treinamento.tbc.com.br</EMAIL>
    </GUSUARIO>
</CODUSUARIO>

```

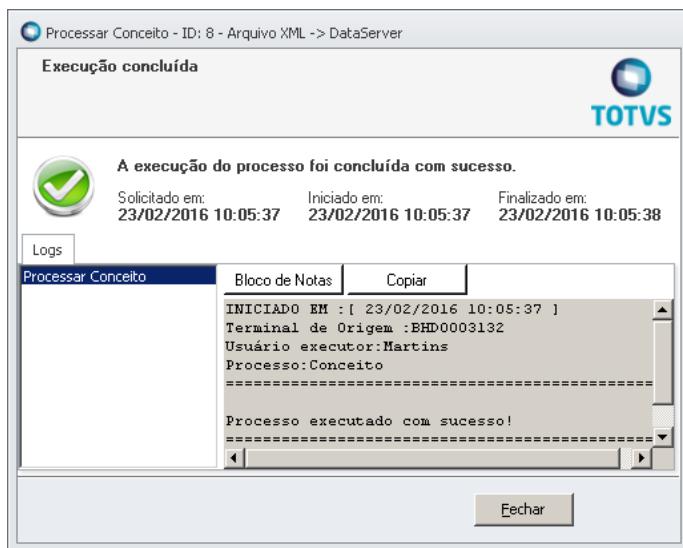
Filtro contendo os usuários a serem trabalhados

# TOTVS Business Conector



## Cadastro de usuário com a aplicação do filtro dos usuários a serem trabalhados

## Execução do conceito



Cadastro de usuário após a execução do conceito com a aplicação do filtro dos trabalhados

[x]	Usuário	Nome	Código de Acesso	Ativo	Início de Validade	Expiração de Validade	Data de Último acesso	E-Mail	Integrado
<input checked="" type="checkbox"/>	VIVIAN	VIVIAN TREINAMENTO TBC	Acesso02	1	21/02/2016		20/10/2011	vivian@treina...	
<input checked="" type="checkbox"/>	VALTTERI	VALTTERI BOTTAS	Acesso02	1	21/02/2016			VALTTERI.BO...	
<input checked="" type="checkbox"/>	SEBASTIAN	SEBASTIAN VETTEL	Acesso01	1	21/02/2016			sebastian.vet...	
<input checked="" type="checkbox"/>	LEWIS	LEWIS HAMILTON	Acesso01	1	21/02/2016			LEWIS.HAMIL...	
<input checked="" type="checkbox"/>	JENSON	JENSON BUTT	Acesso01	1	21/02/2016			JENSON.BUTT...	
<input checked="" type="checkbox"/>	DANIEL	DANIEL RICCIARDO	Acesso01	1	21/02/2016			DANIEL.RICCI...	

## 9. Conceito (novo conceito ou conceito modelo novo)

O novo conceito, ou conceito modelo novo, se trata da ferramenta conceito com uma nova modelagem, arquitetura, interfaces e engines de funcionamento. É a evolução da ferramenta Conceito e abrange uma série de melhorias:

- Design mais simples e intuitivo com funcionamento baseado em drag & drop;
- Leitura de dados de origem em lotes podendo quebrar por intervalos de data e números (quebrando por data ou coligada, por exemplo);
- Permitir a atualização dos dados de origem do processamento;
- Log detalhado de erros de processamento integrado com o Monitor da integração;
- Log de validação estrutural do conceito (evita execução de conceitos montados errados);
- Permitir a utilização de múltiplas conexões de origem ou destino;
- Permitir a utilização de LookUp nas conexões de Data Server e Banco de dados;

- Otimização do uso da memória trabalhando diretamente com os objetos específicos da conexão ao invés de trabalhar somente com datasets;
- Novo Operador (transformação) do tipo "Expressão";
- Conexão WebServices com suporte para leitura e escrita.

## 9.1. Habilitando o novo conceito

Por default o TOTVS RM não vem com o novo conceito habilitado (clientes ainda possuem/trabalham com conceitos antigos) mas após habilitá-lo é bastante simples converter/migrar os conceitos existentes no modelo antigo para o novo modelo.

Os procedimentos a seguir objetivam demonstrar qual o procedimento para parametrizar a base de dados e o TOTVS RM para utilização do novo conceito.

1. Executar o script abaixo na base de dados:

```
DELETE HCPARAMETROS WHERE NOMECOLUNA = 'HabilitaNovoConceito'
GO
INSERT INTO HCPARAMETROS (CODCOLIGADA, NOMECOLUNA, VALORINT) VALUES (0, 'HabilitaNovoConceito', 1)
GO
```

2. Adicionar a linha <add key="EnableConceptConversion" value="true" /> no arquivo de configuração do RM.exe (RM.exe.config), conforme exemplo abaixo:

```
<?xml version="1.0"?>
<configuration>
    <appSettings>
        <add key="JobServer3Camadas" value="false" />
        <add key="Port" value="8050" />
        <add key="Host" value="localhost" />
        <add key="ActionsPath"
value="C:\TOTVS\CorporeRM\RM.Net;C:\TOTVS\CorporeRM\Corpore.Net\Bin" />
        <add key="LibPath" value="C:\TOTVS\CorporeRM\RM.Net" />
        <add key="EnableCompression" value="true" />
        <add key="EnableConceptConversion" value="true" />
    </appSettings>
    <runtime>
        <NetFx40_LegacySecurityPolicy enabled="true" />
    </runtime>
    <startup>
        <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.0" />
    </startup>
</configuration>
```

**OBSERVAÇÃO:** mesmo habilitando o conceito novo é possível trabalhar com os dois modelos no mesmo ambiente/base de dados desde que respeitadas a premissa de que as rotinas (interfaces, execução, etc.) para manter o conceito serão de acordo com o modelo deste. Se for modelo antigo então serão as rotinas do modelo antigo, se for o conceito no modelo novo então os tratamentos serão considerando o novo conceito.

Após tais procedimentos basta agora, sempre que desejar criar um conceito no modelo novo, segurar a tecla CTRL e clicar em INCLUIR.

Sistema Conceito - 03.9.0507 - TOTVS Série T ...

Integração Ferramentas Customização Gestão Ambiente

Conceito Serviços Monitor da Fila de Integração RM Conector Gerador de Saída Mensagem Única

Conceito - 03.9.0507

Layout: Padrão Anexos Processos [Filtro: Todos]

Arraste aqui o cabeçalho de uma coluna para agrupar

[x]	Novo	Id. Conceito	Nome	Última Execução	Nome do Sistema	Action relacionada
▶		1	Conceito Geral	15/02/2016	TODOS	
		2	SQL -> XML	16/02/2016	TODOS	
		3	XML -> Tabela do banco de dados	16/02/2016	TODOS	
		5	DataServer -> Arquivo texto	22/02/2016	TODOS	
		7	SQL -> Arquivo XML	22/02/2016	TODOS	
		8	Arquivo XML -> DataServer	23/02/2016	TODOS	
		9	Arquivo texto -> WebService	24/02/2016	TODOS	
		10	Arquivo XML -> WebService (wsCRMAten...		TODOS	

0 |

Sistema Mensagens

Sistema Conceito (Novo) - TOTVS Série T Construção e Projetos (RM) A...

Integração Ferramentas Customização Gestão Ambiente

Conceito Serviços Monitor da Fila de Integração RM Conector Gerador de Saída Mensagem Única

Conceito - 03.9.0507 Conceito (Novo)

Ligações Propriedades

ID: 0

Nome:

Versão:

Sistema:

Descrição:

Parar execução caso houver algum aviso  
 Usar contexto padrão

Contexto:

Conexões

Nome

- Conexões de Origem
- Conexões de Destino

Conexões

Operações

Parâmetros

Contexto

Valor Fixo

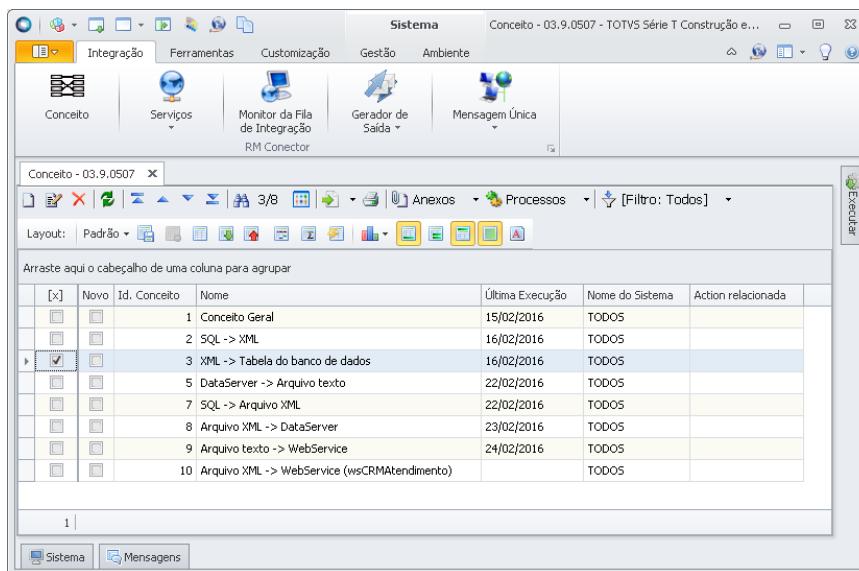
Expressão

Sistema Mensagens

## 9.2. Convertendo conceitos antigos para o novo modelo

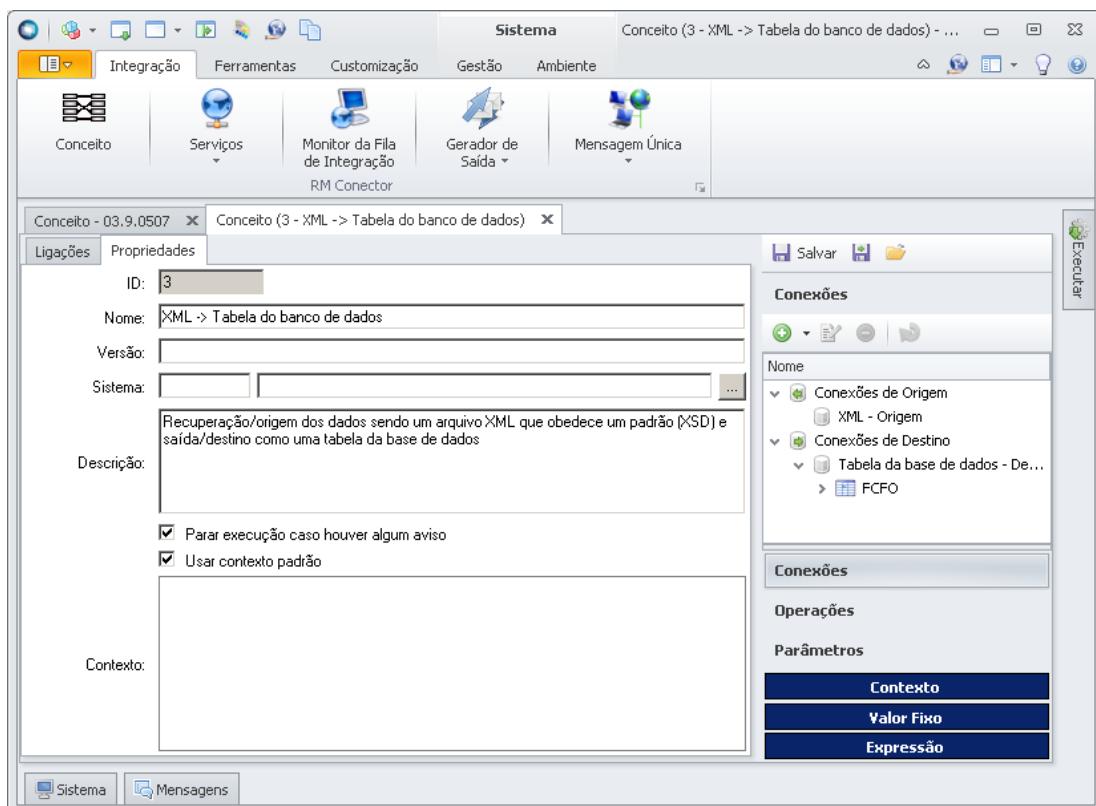
É possível converter um conceito antigo para o novo:

- Selecionar o registro/conceito;
- Segurar a tecla CTRL;
- Clicar no botão EDITAR (ou clicar 2 vezes sobre o registro do conceito antigo);
- O conceito será automaticamente convertido.

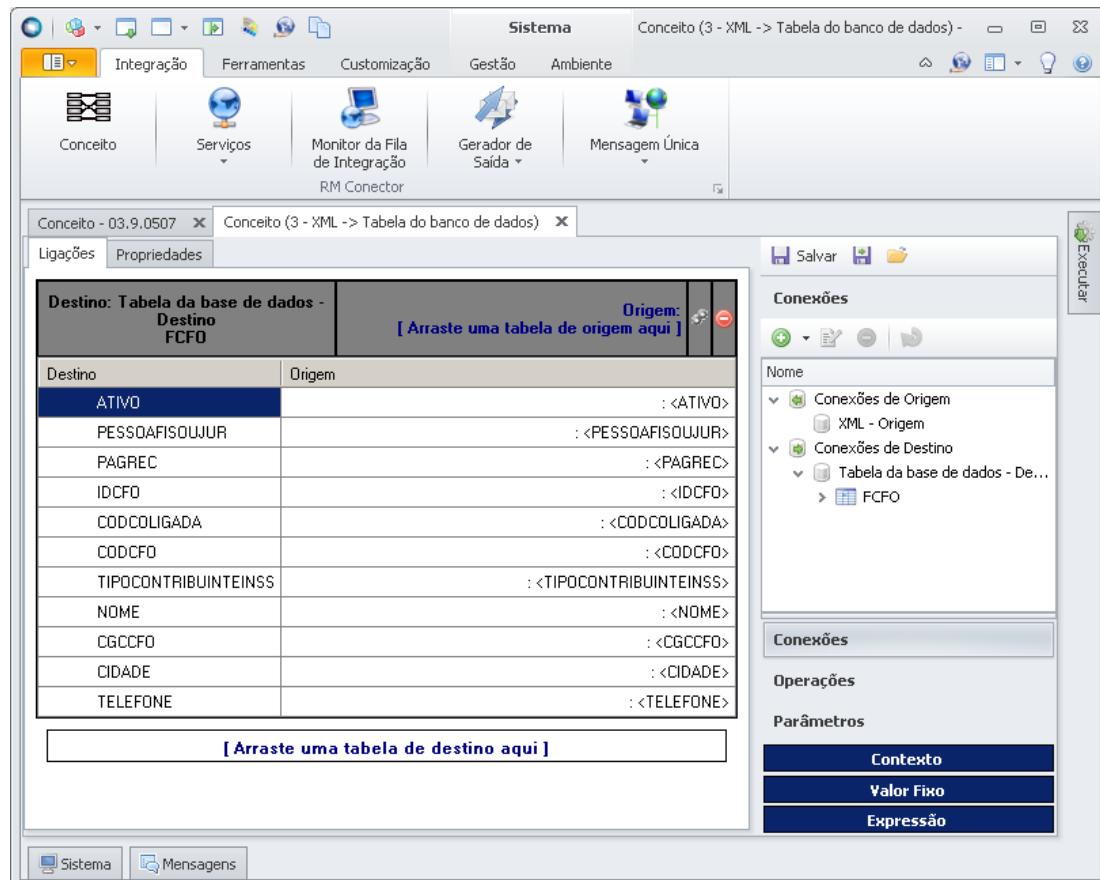


The screenshot shows a list of concepts in a table. The columns are: [x], Novo, Id. Conceito, Nome, Última Execução, Nome do Sistema, and Action relacionada. The selected concept is row 3: "3 XML -> Tabela do banco de dados".

[x]	Novo	Id. Conceito	Nome	Última Execução	Nome do Sistema	Action relacionada
		1	Conceito Geral	15/02/2016	TODOS	
		2	SQL -> XML	16/02/2016	TODOS	
<input checked="" type="checkbox"/>		3	XML -> Tabela do banco de dados	16/02/2016	TODOS	
		5	DataServer -> Arquivo texto	22/02/2016	TODOS	
		7	SQL -> Arquivo XML	22/02/2016	TODOS	
		8	Arquivo XML -> DataServer	23/02/2016	TODOS	
		9	Arquivo texto -> WebService	24/02/2016	TODOS	
		10	Arquivo XML -> WebService (wsCRMAtendimento)		TODOS	



The screenshot shows the details of the converted concept. The 'Ligações' tab is selected. The concept has ID 3, Name "XML -> Tabela do banco de dados", and Version 1.0. The 'Descrição:' field contains: "Recuperação/origem dos dados sendo um arquivo XML que obedece um padrão (XSD) e saída/destino como uma tabela da base de dados". The 'Operações' section includes checkboxes for "Parar execução caso houver algum aviso" and "Usar contexto padrão". The 'Conexões' section shows the connection details for both source and destination systems. The 'Operações' section also includes tabs for 'Contexto', 'Valor Fixo', and 'Expressão'.



### Exercício

1. Selecione o primeiro registro criado no cadastro de Conceito e realize a conversão para o novo modelo.

Após a conversão tente realizar a execução do mesmo e verifique se o processo foi executado com sucesso.

## TOTVS Business Conector



The screenshot shows the TOTVS Business Conector application window. In the center, there is a dialog box titled "ATENÇÃO" (Attention) with the message: "Este conceito foi convertido apartir de um formato antigo. Após salvar ele somente poderá ser alterado no novo editor." (This concept was converted from an old format. After saving, it can only be changed in the new editor.) Below the message are "Sim" (Yes) and "Não" (No) buttons. To the right of the dialog, there is a "Conexões" (Connections) panel showing a tree structure of connections. On the left, there is a "Mensagens de Validação do Objeto" (Object Validation Messages) panel displaying validation errors for an XML-to-table connection. The main workspace shows a list of destination tables.

The screenshot shows the TOTVS Business Conector application window with a list of concepts in the main workspace. A red arrow points to the third item in the list, which is "3 XML -> Tabela do banco de dados". The list includes the following items:

	Novo	Id. Conceito	Nome	Última Execução	Nome do Sistema	Action relacionada
<input type="checkbox"/>	<input type="checkbox"/>	1	Conceito Geral	15/02/2016	TODOS	
<input type="checkbox"/>	<input type="checkbox"/>	2	SQL -> XML	16/02/2016	TODOS	
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	3	XML -> Tabela do banco de dados	16/02/2016	TODOS	
<input type="checkbox"/>	<input type="checkbox"/>	5	DataServer -> Arquivo texto	22/02/2016	TODOS	
<input type="checkbox"/>	<input type="checkbox"/>	7	SQL -> Arquivo XML	22/02/2016	TODOS	
<input type="checkbox"/>	<input type="checkbox"/>	8	Arquivo XML -> DataServer	23/02/2016	TODOS	
<input type="checkbox"/>	<input type="checkbox"/>	9	Arquivo texto -> WebService	24/02/2016	TODOS	
<input type="checkbox"/>	<input type="checkbox"/>	10	Arquivo XML -> WebService (wsCRMAtendimento)		TODOS	

### 9.3. Criando um conceito no novo modelo

Para ilustrar o uso do novo conceito, e alguns de seus novos recursos, vamos elaborar um conceito para recuperar dados de um SQL e listá-los como arquivo texto.

A origem de dados será um SQL que recupera da base do TOTVS RM os alunos matriculados em cursos e a situação das matrículas. Note que no SQL vamos dispor de um parâmetro que é a coligada para que, ao executá-lo, o usuário informe/selecione para qual coligada deseja filtrar os alunos/matrículas.

```

SELECT DISTINCT
    GCOLIGADA.NOME NOMEEMPRESA, SMATRICPL.RA, PESSOA.NOME NOMEALUNO, PESSOA.CPF,
    SHABILITACAOFILIAL.CODCURSO, SCURSO.NOME NOMECURSO,
    YEAR(SPLETIVO.DTINICIO) ANOLETIVO, SMATRICPL.PERIODO, SMATRICPL.CODTURMA,
    SMATRICPL.CODSTATUS, SSTATUS.DESCRICAO
FROM SMATRICPL (NOLOCK)
JOIN GCOLIGADA (NOLOCK) ON (GCOLIGADA.CODCOLIGADA = SMATRICPL.CODCOLIGADA)
JOIN SALUNO (NOLOCK) ON (SALUNO.CODCOLIGADA = SMATRICPL.CODCOLIGADA AND SALUNO.RA = SMATRICPL.RA)
JOIN PESSOA (NOLOCK) ON (SALUNO.CODPESSOA = PESSOA.CODIGO)
LEFT JOIN SHABILITACAOALUNO (NOLOCK) ON (
    SHABILITACAOALUNO.CODCOLIGADA = SMATRICPL.CODCOLIGADA
    AND SHABILITACAOALUNO.IDHABILITACAOFILIAL =
    SMATRICPL.IDHABILITACAOFILIAL
    AND SHABILITACAOALUNO.RA = SMATRICPL.RA)
LEFT JOIN SHABILITACAOFILIAL (NOLOCK) ON (
    SHABILITACAOFILIAL.CODCOLIGADA =
    SHABILITACAOALUNO.CODCOLIGADA
    AND SHABILITACAOFILIAL.IDHABILITACAOFILIAL =
    SHABILITACAOALUNO.IDHABILITACAOFILIAL)
LEFT JOIN SPLETIVO (NOLOCK) ON (
    SPLETIVO.CODCOLIGADA = SMATRICPL.CODCOLIGADA
    AND SPLETIVO.IDPERLET = SMATRICPL.IDPERLET)
JOIN SCURSO (NOLOCK) ON (
    SCURSO.CODCOLIGADA = SHABILITACAOFILIAL.CODCOLIGADA
    AND SCURSO.CODCURSO = SHABILITACAOFILIAL.CODCURSO)
JOIN SSTATUS (NOLOCK) ON (
    SSTATUS.CODCOLIGADA = SMATRICPL.CODCOLIGADA
    AND SSTATUS.CODSTATUS = SMATRICPL.CODSTATUS)
WHERE SMATRICPL.CODCOLIGADA = :CODCOLIGADA
/*and*/
  
```

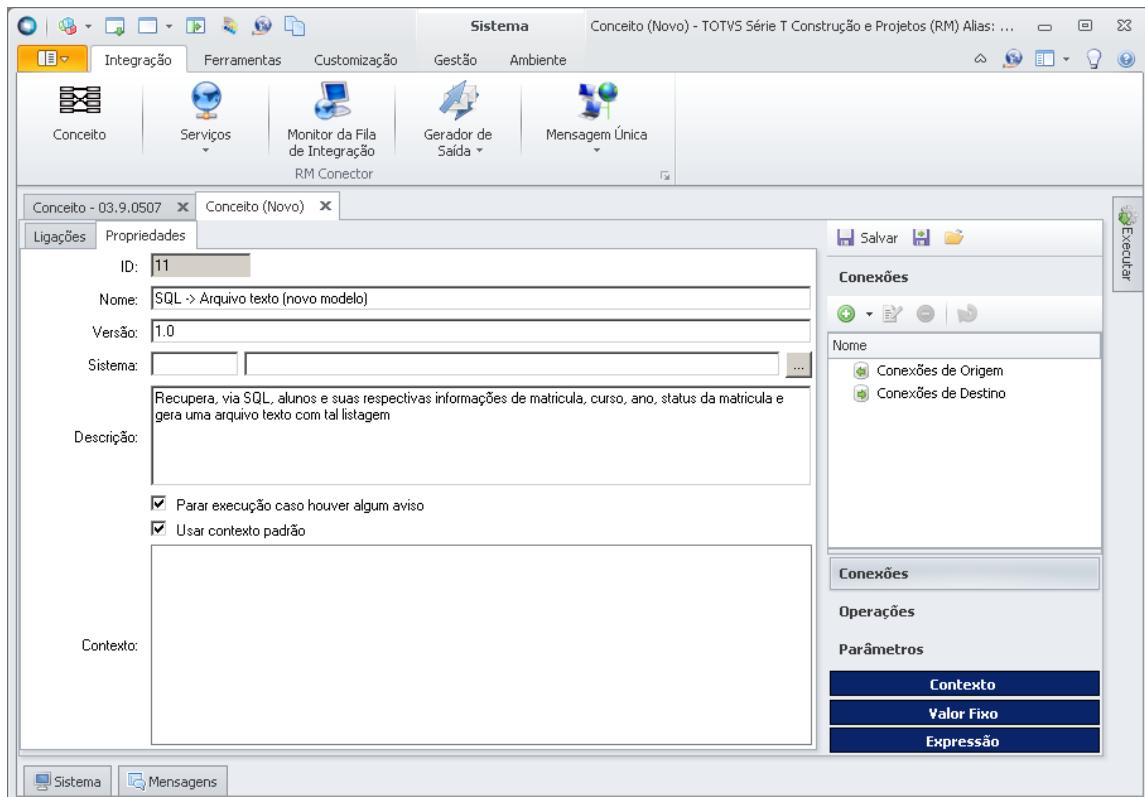
Então vamos a criação do conceito (lembre-se de clicar em <Incluir> mantendo pressionada a tecla CRTL).

### 9.4. Identificação (Propriedades e Ligações)

A edição se abre com duas abas: Ligações e Propriedades.

- Ligações é a aba equivalente a Correspondências do conceito antigo.
- Propriedades é a aba principal para as configurações básicas do conceito. Nesta aba teremos acesso a conexões origem/destino, operações/transformações (que posteriormente utilizaremos nas Ligações [correspondências]) e parâmetros de execução do conceito.  
As estruturas de origem/destino encontram-se “dentro” das conexões de origem/destino.

**OBSERVAÇÃO:** Vale mais uma vez lembrar que o novo conceito faz uso do recurso “drag & drop” (arrastar e soltar) para a montagem de várias estruturas.



## 9.5. Parâmetros para execução do conceito

Neste conceito, como exemplo, iremos utilizar uma nova funcionalidade do novo conceito, a possibilidade de definir parâmetros a serem informados pelo usuário na execução do conceito.

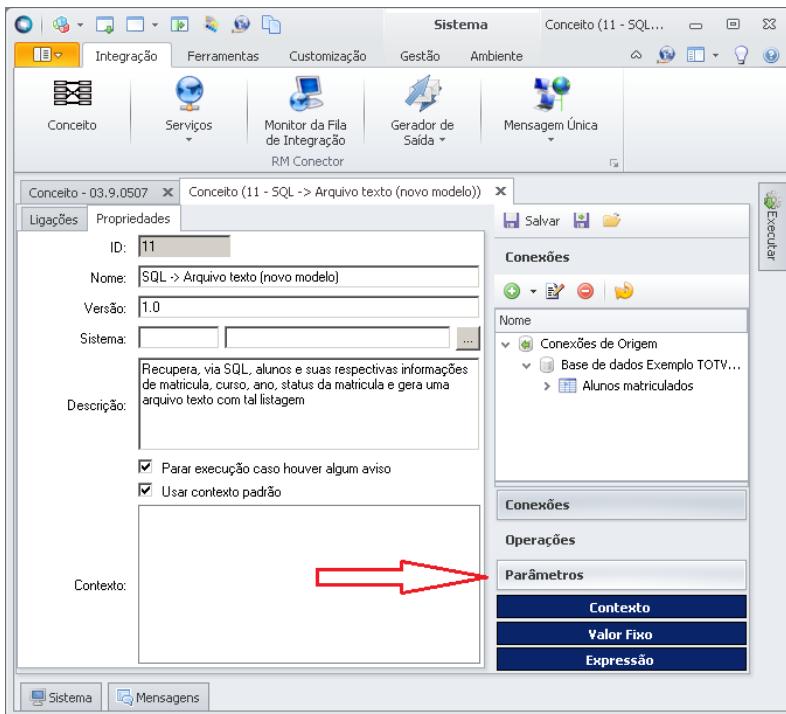
Ao executar este conceito o usuário deverá informar/selecionar para qual coligada os alunos/matriculados deverão ser recuperados. O valor informado deve ser repassado ao SQL para que ele proceda com este filtro ao recuperar os registros.

A SQL de origem dos dados já está preparada para receber este valor:

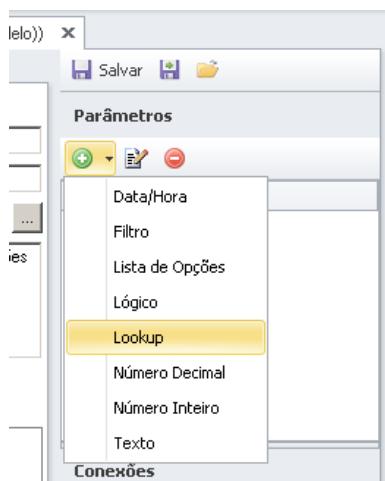
```
WHERE SMATRICPL.CODCOLIGADA = :CODCOLIGADA
```

Então criaremos e configuraremos um parâmetro do conceito de maneira a ser apresentado ao usuário, permita o seu preenchimento e tal valor possa ser utilizado no SQL que recupera os dados.

Na aba Propriedades do conceito e clique na opção “Parâmetros” e o conceito apresentará, no lado direto da interface de edição, o acesso a tais recursos.



Como queremos apresentar ao usuário uma lista de coligadas para seleção então crie um novo parâmetro – acessando “Adicionar parâmetro” (botão com a imagem “+” em verde) – já selecionando o tipo “Lookup”.

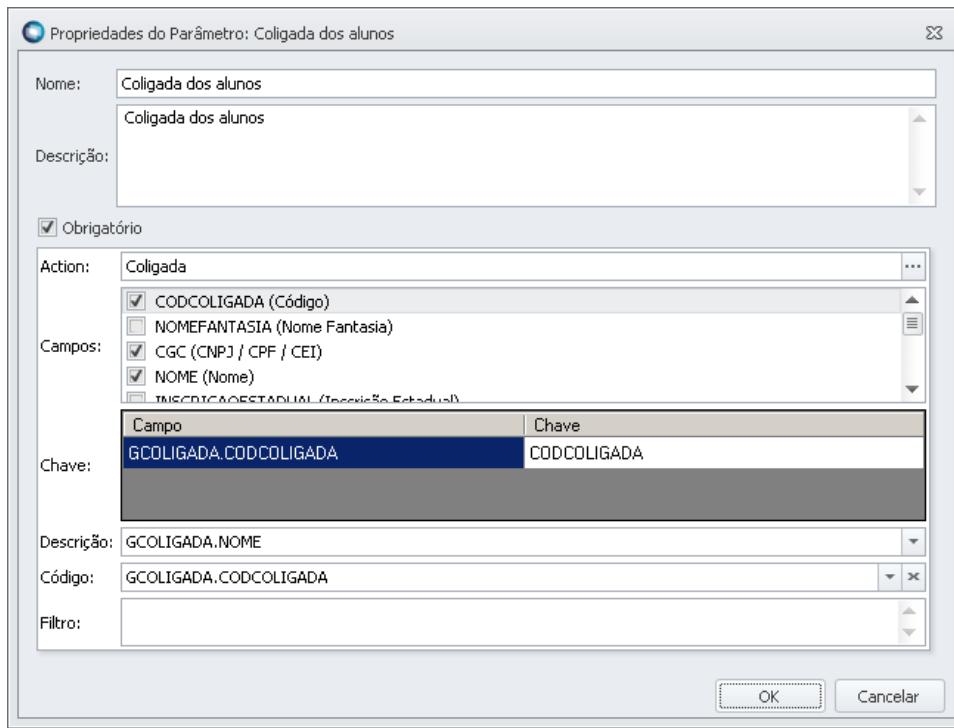


Na interface para definição dos parâmetros, informe um nome a este será o nome a ser apresentado ao usuário no wizard de execução do conceito.

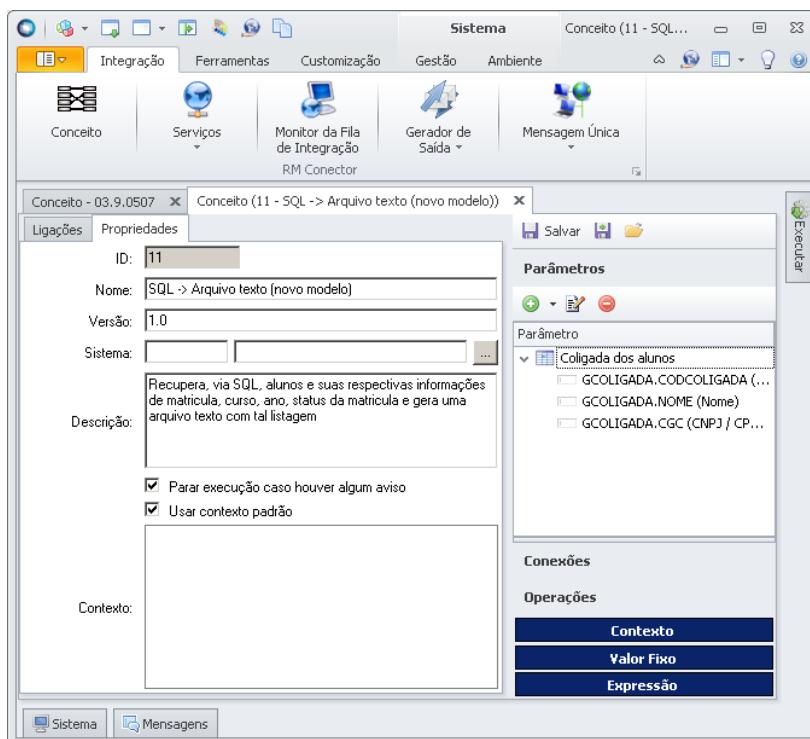
Como se trata de um lookup, na propriedade “Action” informe qual será o provedor de dados para este lookup (em nosso exemplo será Coligada).

Na propriedade “Campos” selecione os campos que serão apresentados pelo lookup e o campo que será o valor, dentro do conceito, deste parâmetro.

Em “Descrição” informe o campo que será apresentado no campo texto do lookup e no campo “Código” o campo que representará o valor deste parâmetro quando utilizado pelo conceito.

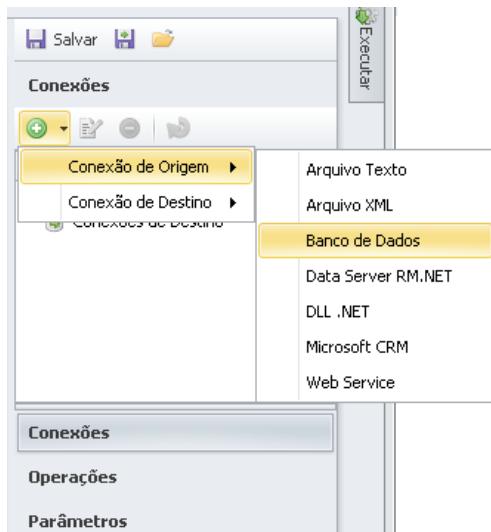


Ao clicar em "OK", e retornar a tela anterior, ele constará da lista de parâmetros e, ao executar o conceito, um lookup será apresentado ao usuário para uma coligada seja selecionada.



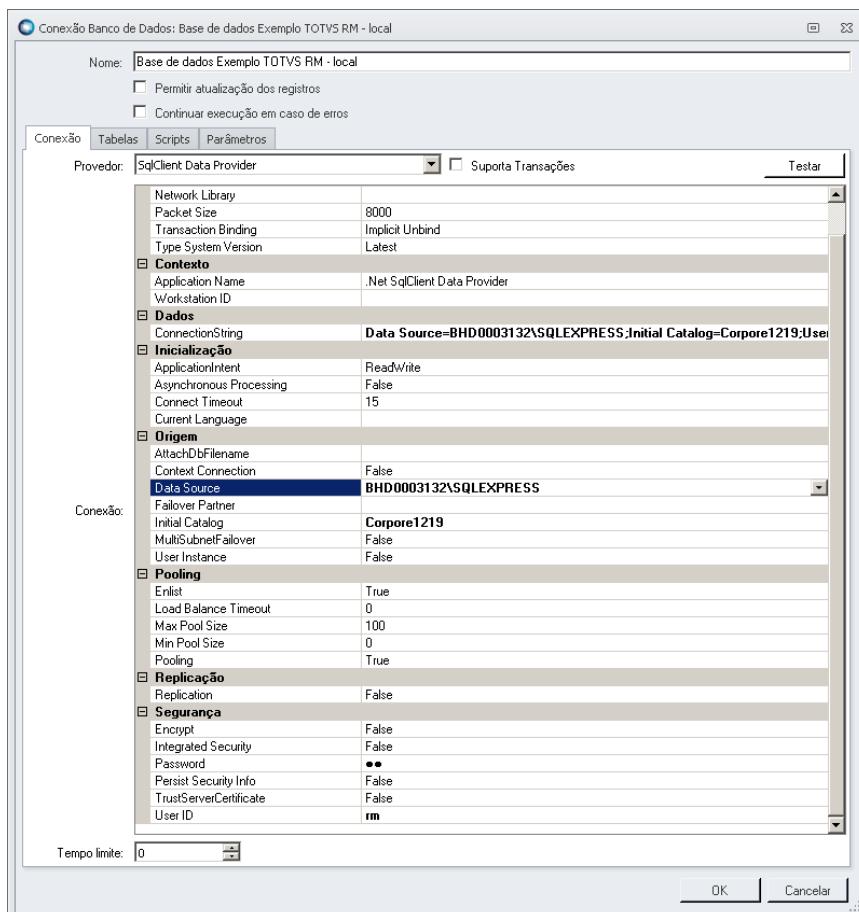
## 9.6. Conexão Origem e Conexão Destino

As conexões de origem/destino são criadas acessando o item Conexões, do lado direito da interface, e então o botão "Adicionar conexão" (botão com a imagem "+" em verde) onde nesta funcionalidade já se seleciona o tipo de conexão.



### 9.6.1. Conexão/Estrutura de dados origem

Crie a conexão de origem sendo do tipo “Banco de dados”, informe um nome, selecione um provedor como sendo “SqlClient data Provider” e configure a string de conexão.

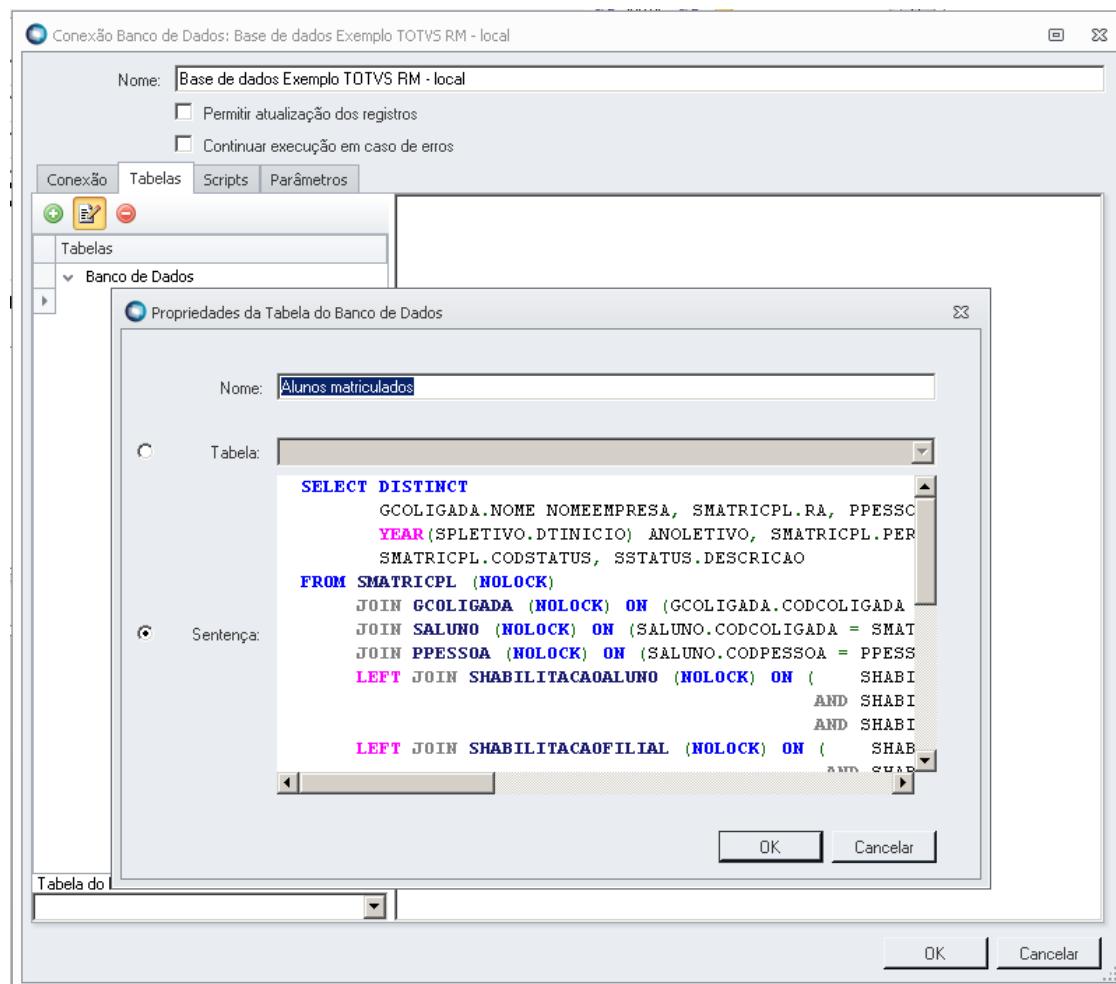


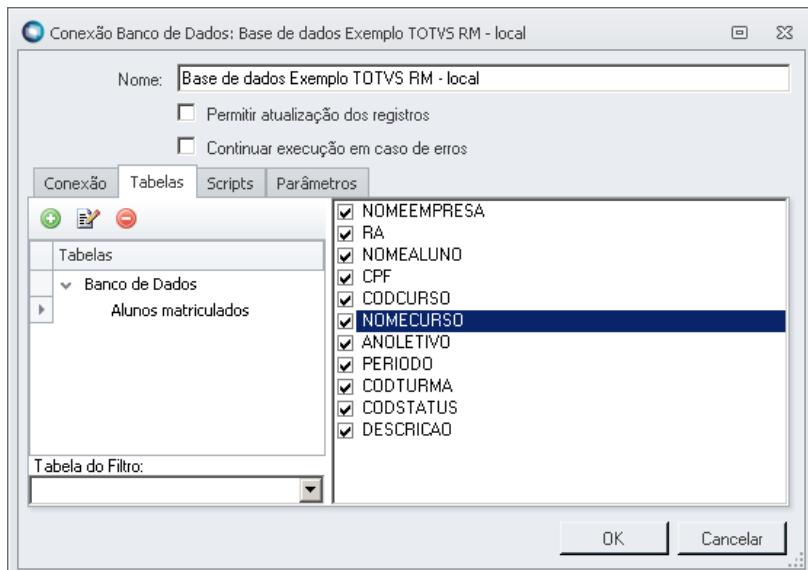
Após a conexão, nesta mesma interface na aba Tabelas, vamos definir a origem dos dados (em nosso exemplo um SQL) e a estrutura de dados de origem em um único passo.

Na aba Tabelas acesse o botão “Adicionar tabela” (botão com a imagem “+” em verde). Na interface que será apresentada informe um nome à tabela, selecione a opção “Sentença” e em seguida informa o SQL (que citamos mais acima) e clique “OK”.

**OBSERVAÇÃO:** No uso de SQLs para a recuperação de dados é obrigatório que a sentença tenha na cláusula WHERE o texto /\*and\*/ - quando já existirem condições na cláusula where - ou o texto /\*where\*/ - quando a sentença não possuir cláusula where.

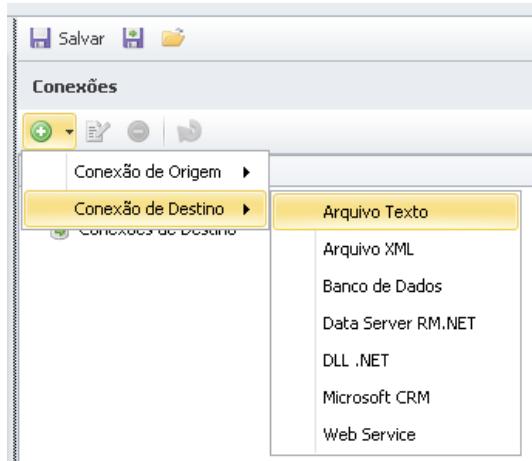
Você vai notar que assim que retornar para a tela anterior (após o clique no “OK”) a estrutura origem já estará montada bastando apenas que você selecione os campos a serem utilizados.



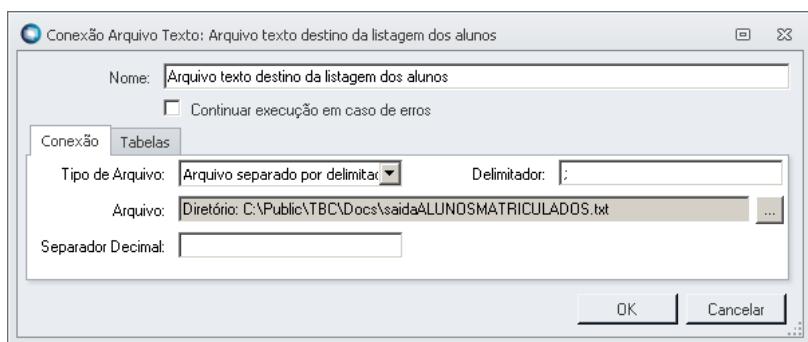


### 9.6.2. Conexão/Estrutura de dados destino

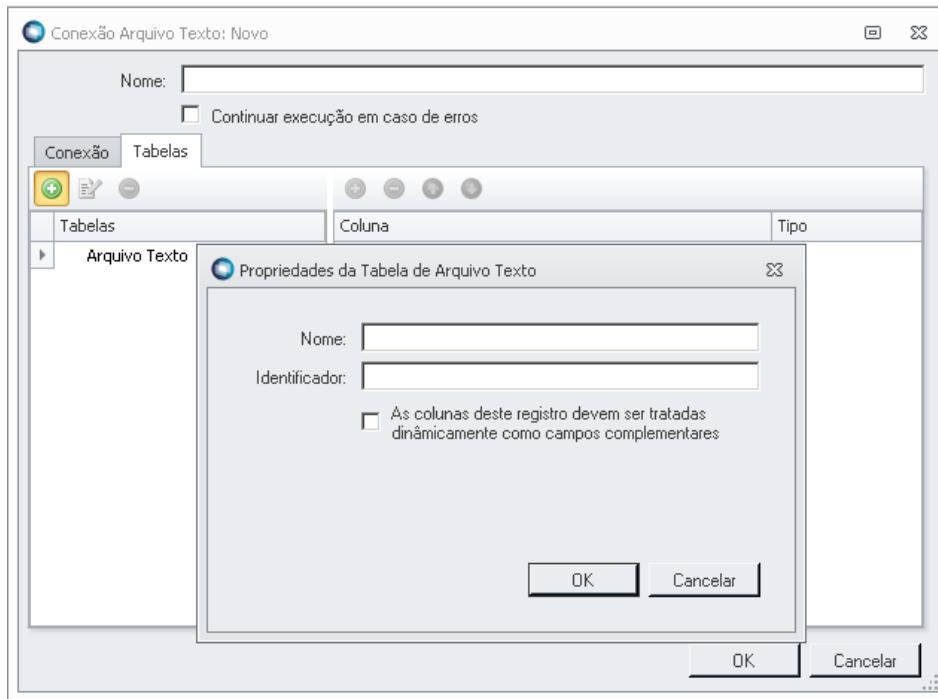
A conexão/estrutura de destino é criada e configurada de maneira semelhante à origem mas considerando que se trata de um arquivo texto. Então no momento de “Adicionar conexão” temos de selecionar “Destino” e o tipo como “Arquivo texto”.



Informe um nome, se teremos delimitador de campos ou não, o caractere delimitador (se for o caso) e o diretório/nome de arquivo de destino.

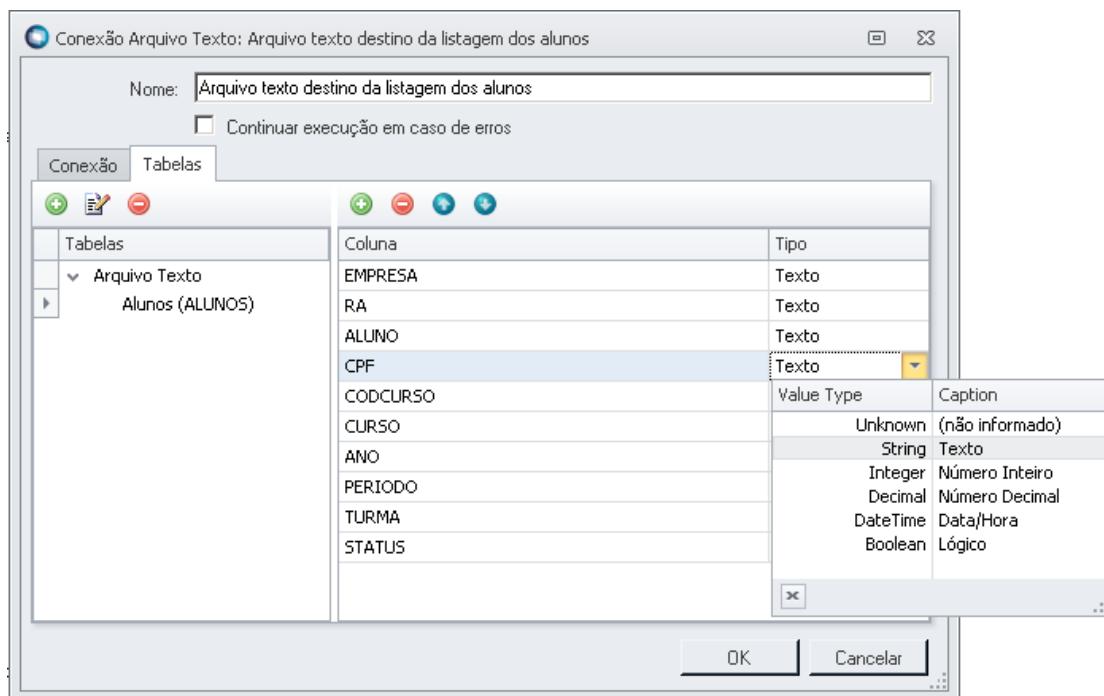


Para a estrutura destino, ainda na interface de edição da conexão destino, acesse a aba “Tabelas” e então adicione uma tabela informando o nome e o identificador.



Na sequência, após clicar em “OK” na definição do nome/identificador, selecione a tabela criada e adicione seus campos/colunas clicando no botão “Adicionar coluna” (botão com a imagem “+” em verde) no controle do lado direito da tela.

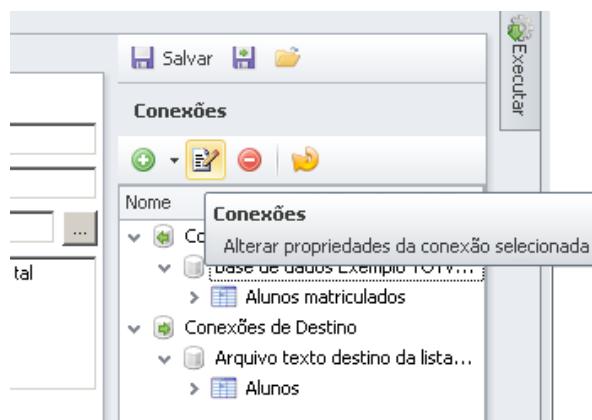
**ATENÇÃO:** No novo conceito é necessário definir o tipo de cada campo.



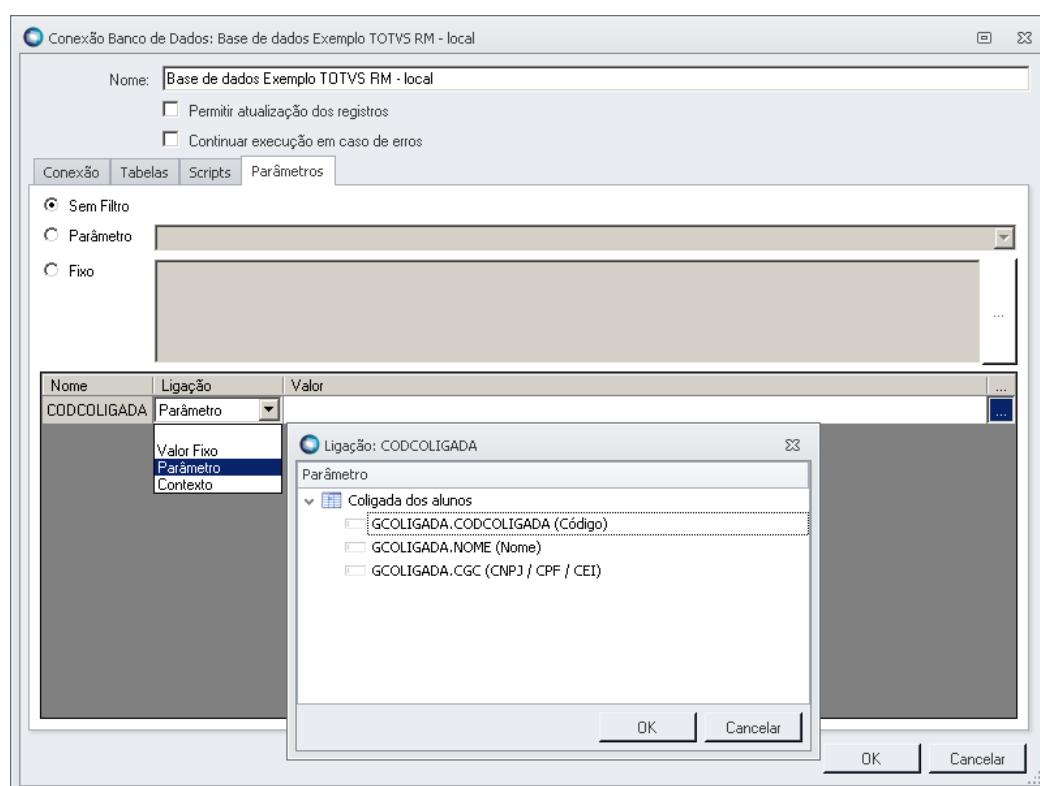
### 9.6.3. Vínculo de parâmetros às fontes de dados (conexões)

Uma vez que temos o parâmetro coligada configurado vamos fazer com que o parâmetro :CODCOLIGADA do SQL na conexão de origem reconheça o valor selecionado pelo usuário.

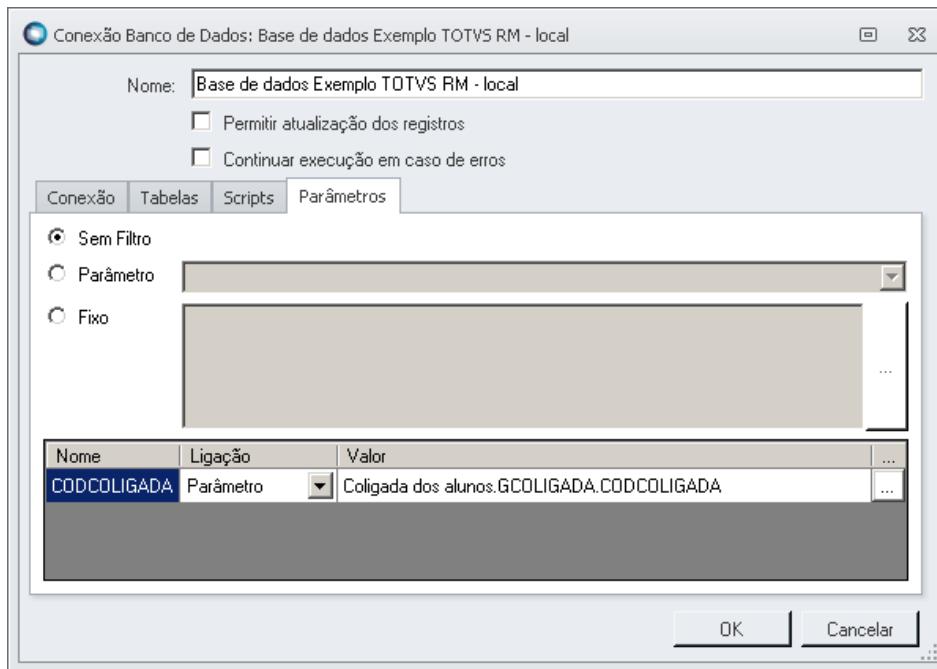
Na aba de Propriedades do conceito, no lado direito acessando a opção Conexões, selecione a conexão de origem e acesse sua edição.



Na interface de edição da conexão de origem acesse a aba “Parâmetros”. Observe que o conceito já reconheceu o parâmetro do SQL, o listou na visão e precisamos, através desta visão, informar a origem do valor a ser utilizado. Selecione o registro que representa o parâmetro do SQL (em nosso caso CODCOLIGADA), na coluna “Ligaçao” selecione o tipo de procedência do valor (em nosso caso será “Parâmetro” pois é um parâmetro do conceito). Ainda na visão, na coluna “Valor”, selecione o campo que conterá o valor a ser utilizado (em nosso caso GCOLIGADA.CODCOLIGADA) e clique em “OK”.



## TOTVS Business Conector



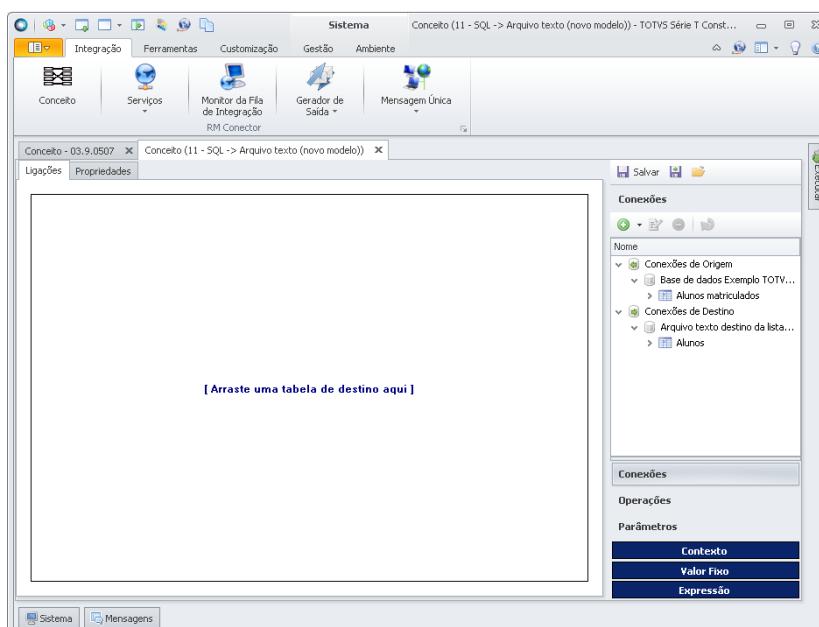
Ao clicar em “OK” então finalizamos a amarração do parâmetro do conceito (coligada a ser informada/selecionada pelo usuário no momento de execução do conceito) com o parâmetro do SQL de recuperar os registros. Isso significa que o SQL será filtrado pela coligada selecionada pelo usuário.

### 9.7. Ligações (Correspondências)

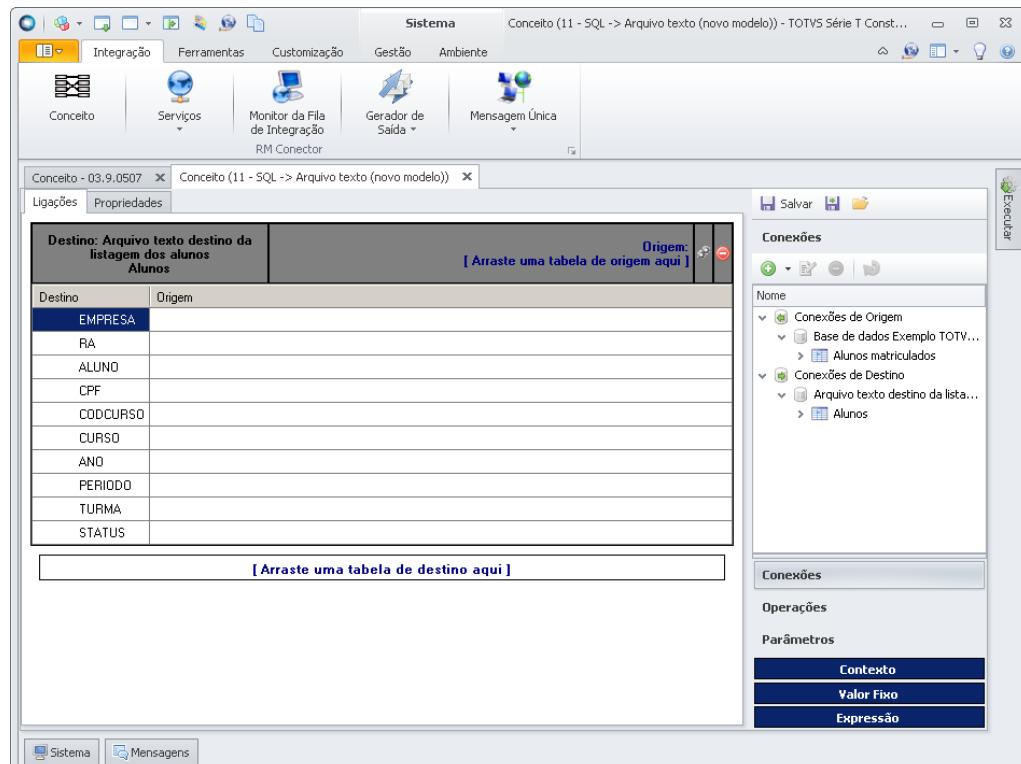
Uma vez que configuramos as conexões e estruturas de origem/destino o próximo passo é efetuar a correspondências dos campos (relacionamento, mapeamento) que no novo conceito é realizado através da aba “Ligações”.

Na interface de edição do conceito acesse a aba “Ligações”.

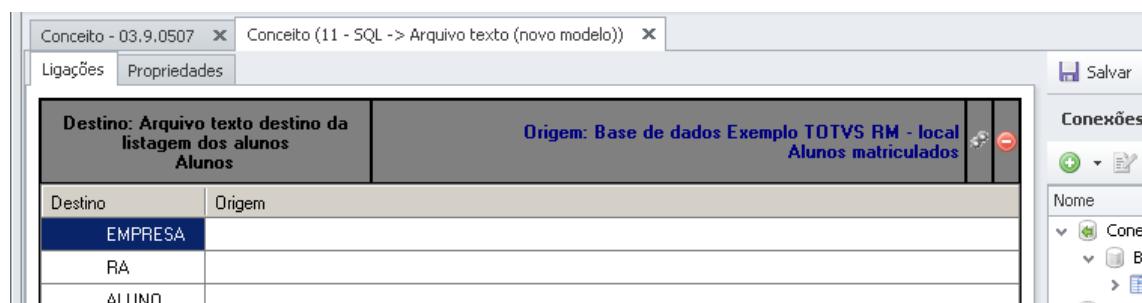
Observe que temos a área das ligações propriamente ditas e, ao lado direito, estão as conexões e estruturas de origem e destino.



Simplesmente selecione a estrutura de destino e “drag & drop” (arraste e solte) na área de ligações.



Para efetuar o mapeamento com a estrutura origem, efetue o mesmo passo de “arrastar e soltar” agora selecionando a estrutura origem e “soltando-a” na barra de título das ligações (a barra em cinza) na área onde está escrito “Origem: [Arraste uma tabela de origem aqui]”.



A correspondência entre os campos também é realizada seguindo a mesma ideia: arrastando e soltando. Na área de conexões expanda a tabela/estrutura de origem de maneira a apresentar seus campos. Com os campos em vista basta selecionar o campo origem desejado e arrastá-lo e soltá-lo no campo destino desejado.

The screenshot shows the TOTVS Business Conector interface with the 'Conceito' (Concept) screen open. The 'Destino' (Destination) section is set to 'Arquivo texto destino da listagem dos alunos' (Text file destination for student list). The 'Origem' (Origin) section is set to 'Base de dados Exemplo TOTVS RM - local' (Example TOTVS RM - local database), specifically 'Alunos matriculados'. The 'Conexões' (Connections) panel on the right shows a tree structure of connections, with a red arrow pointing from the 'PERIODO' field in the 'Destino' table to the 'CODCURSO' node under 'Alunos matriculados'.

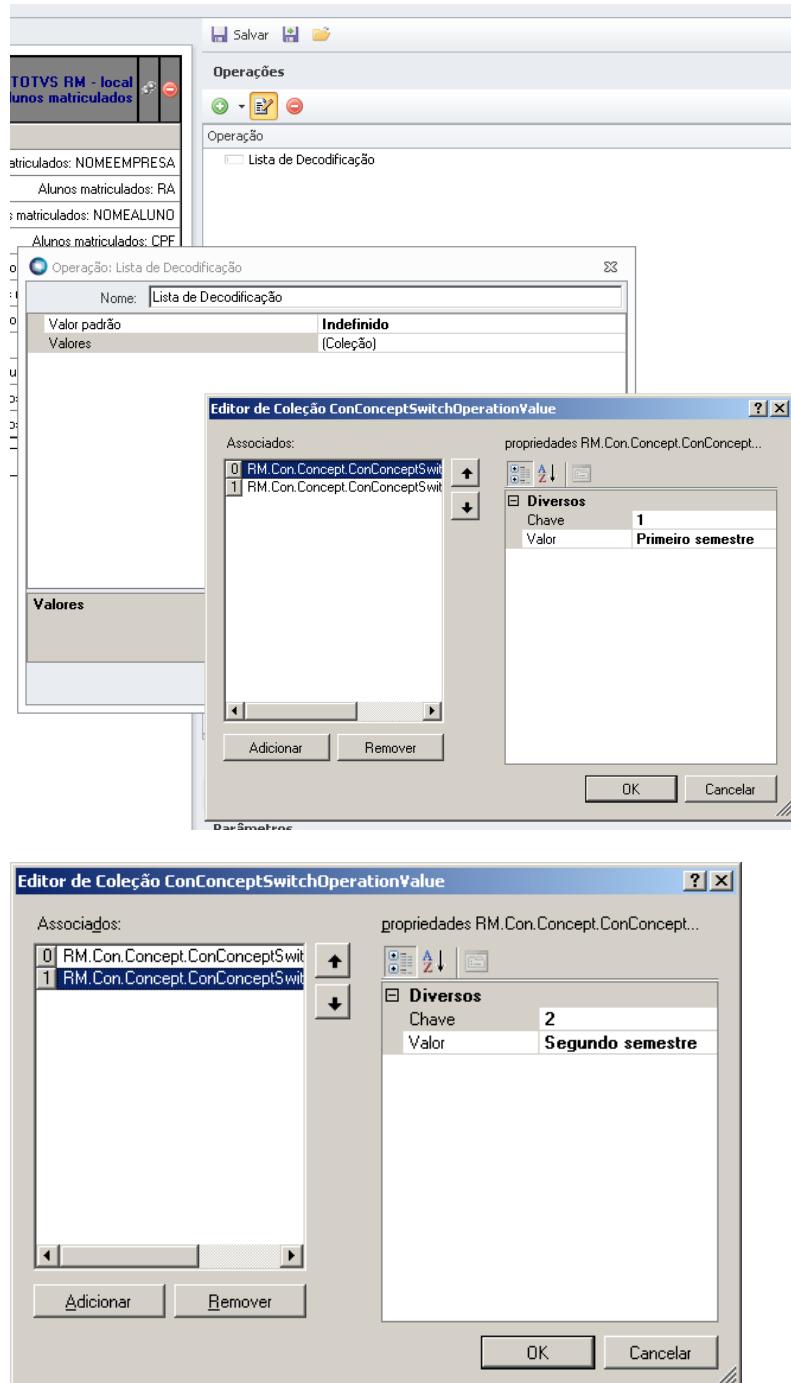
O campo destino PERÍODO foi propositalmente deixado sem relacionamento para apresentarmos, no novo conceito, o uso de operações (ou transformações). A transformação a ser utilizada é a “Lista de decodificação” mas antes precisamos criá-la/cadastrá-la.

## 9.8. Operações (Transformações)

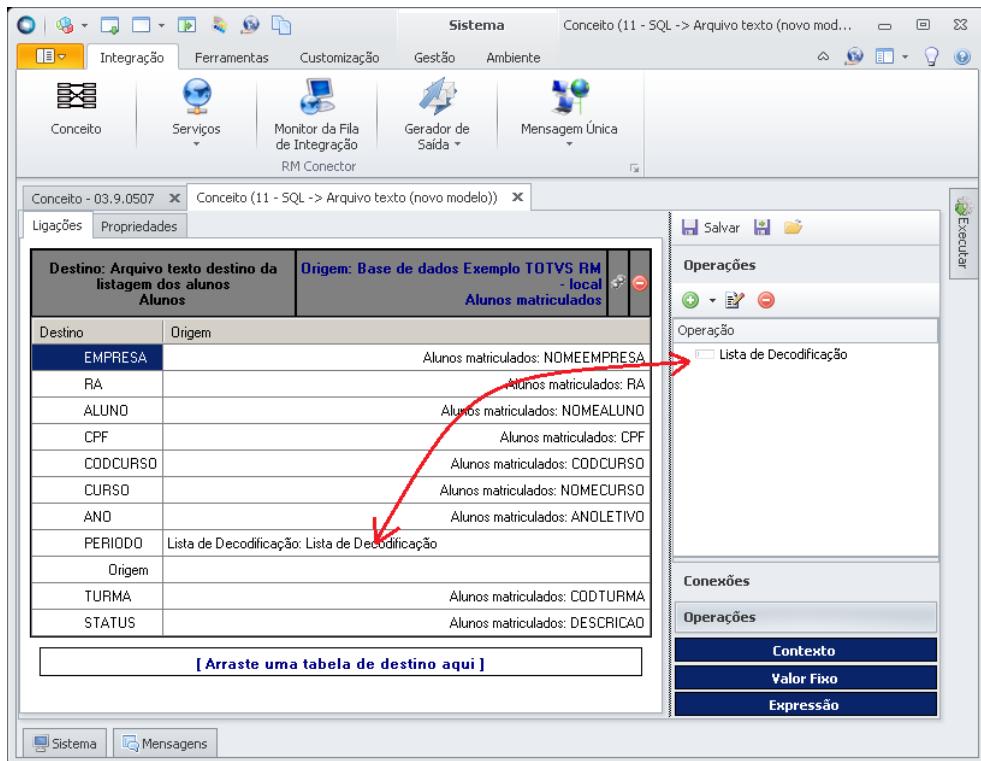
Acesse, no lado direito da tela abaixo da opção de “Conexões”, a opção de “Operações”. Adicione uma operação (botão verde com o símbolo do “+”) e selecione o tipo “Lista de decodificação”.

The screenshot shows the 'Operações' (Operations) panel on the right side of the interface. It lists various operations: Arredondamento, Auto-incremento (RM), Concatenação, Conversão de Valor, Data Atual, Geração de Hash, Guid, and Lista de Decodificação. The 'Lista de Decodificação' option is highlighted with a yellow box.

Informe um valor padrão (default) e, através da propriedade “Coleções”, os pares de valores: valor X na origem (onde X deve ser informado na propriedade “Chave”) será o valor Y no destino (onde Y deve ser informado na propriedade “Valor”).

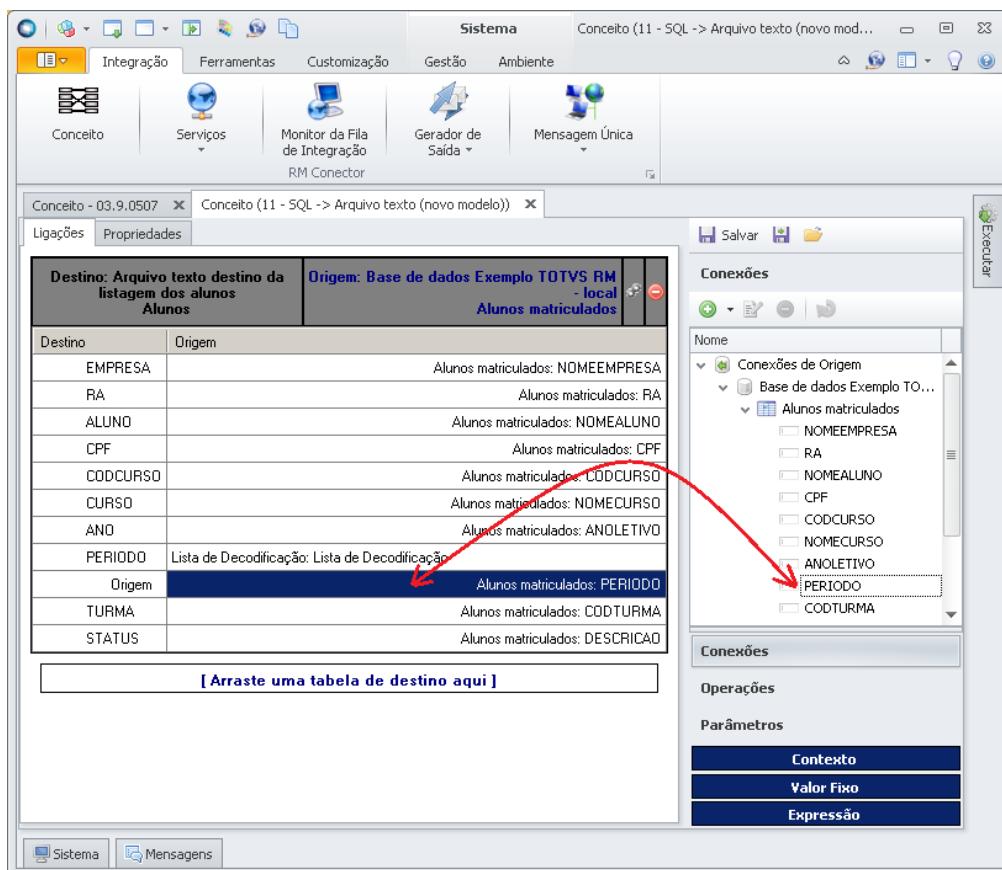


Após a criação da operação ela constará da lista de operações e podemos utilizar o método de “drag & drop” de tal operação para o campo desejado, em nosso caso o PERÍODO.



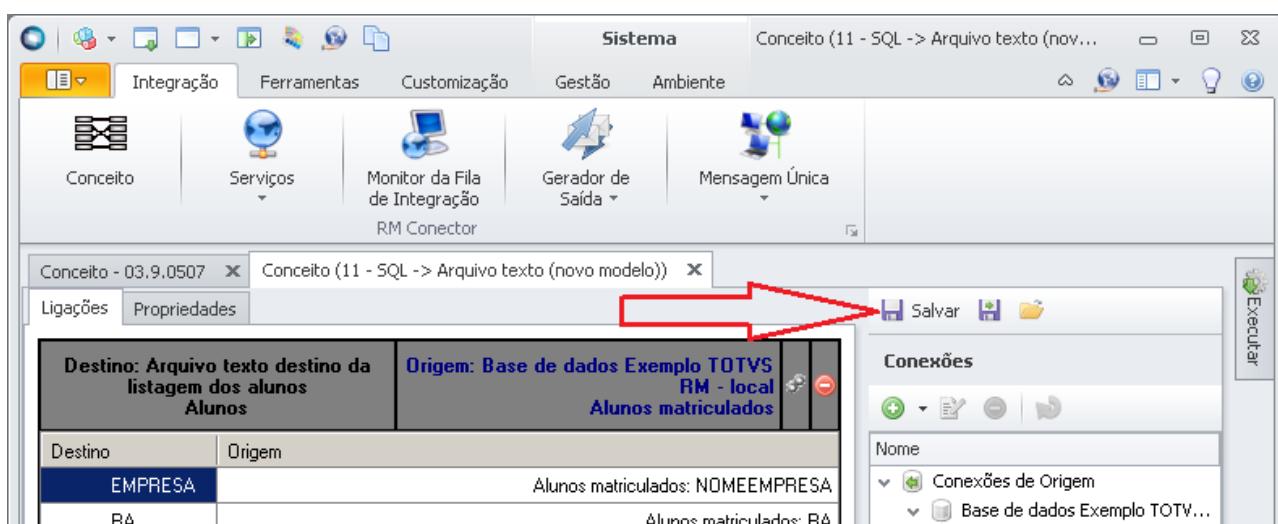
Definimos o preenchimento do campo PERÍODO, através da operação de lista de decodificação, e agora precisamos informar qual a origem dos valores para a lista de decodificação que resultará no campo PERÍODO.

Ao relacionar a operação com o campo destino (passo anterior) perceba que foi apresentado a propriedade “Origem” abaixo do campo destino. Esta propriedade é onde informaremos o campo origem a ser utilizado pela operação para gerar o valor do campo destino. A maneira de informar qual o campo origem é igual ao feito antes, drag & drop. Acesse a opção de “Conexões”, liste os campos da estrutura/tabela de origem, selecione o desejado para tal “Origem” e efetue o relacionamento.



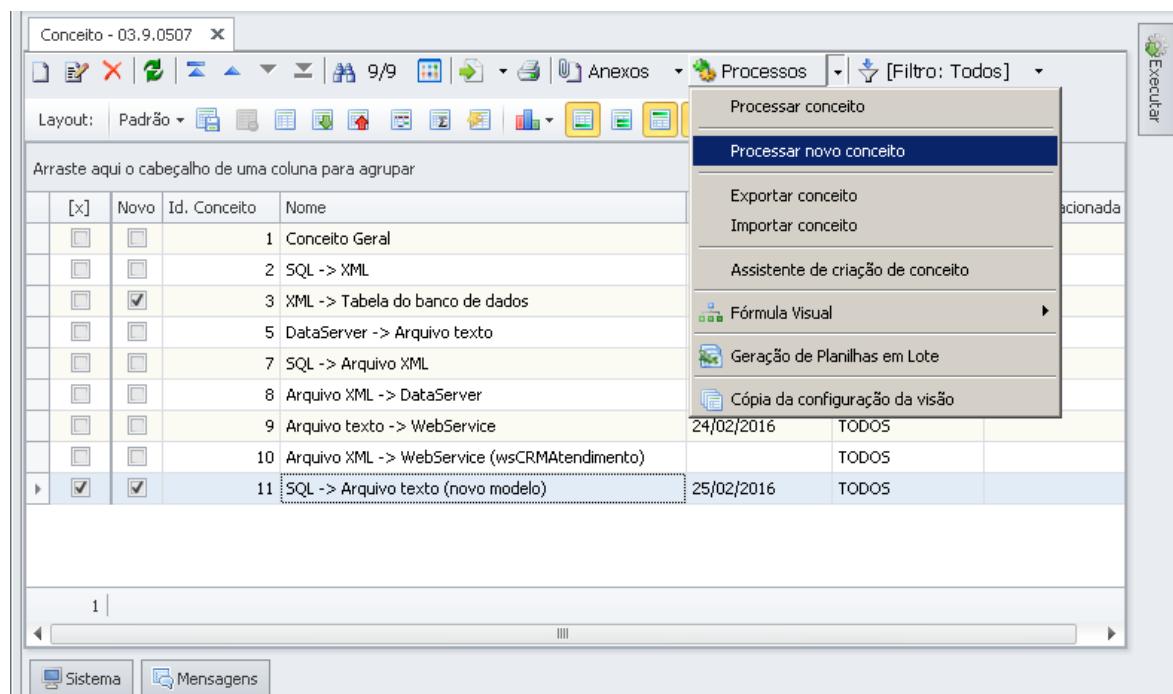
Depois de finalizar a configuração das ligações acrescentando uma operação/transformação podemos salvar o conceito.

Atente-se pois o “Salvar” agora é realizado por um botão que fica posicionado acima da área de opções.  
Como uma das novidades do novo conceito temos algumas validações do conceito ao salvá-lo e caso o mesmo tenha algum problema relacionado às conexões e/ou estruturas então uma mensagem explicativa será apresentada ao usuário mas permitirá o salvamento do conceito.



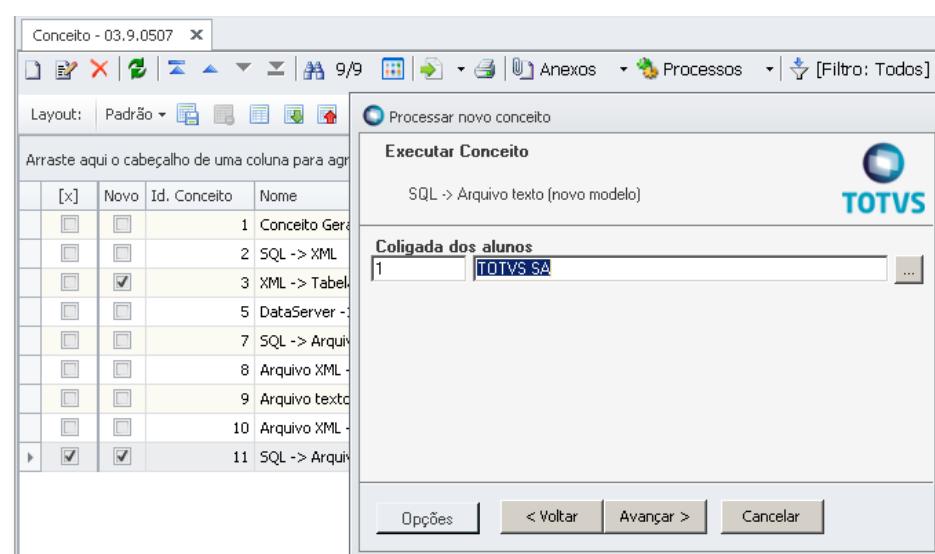
## 9.9. Execução

A execução do novo conceito é realizada através da visão de conceitos acessando o menu de processos. Neste menu temo um processo de nome “Processar novo conceito” e é ele que você deve acionar para executar os conceitos elaborados considerando o novo modelo.

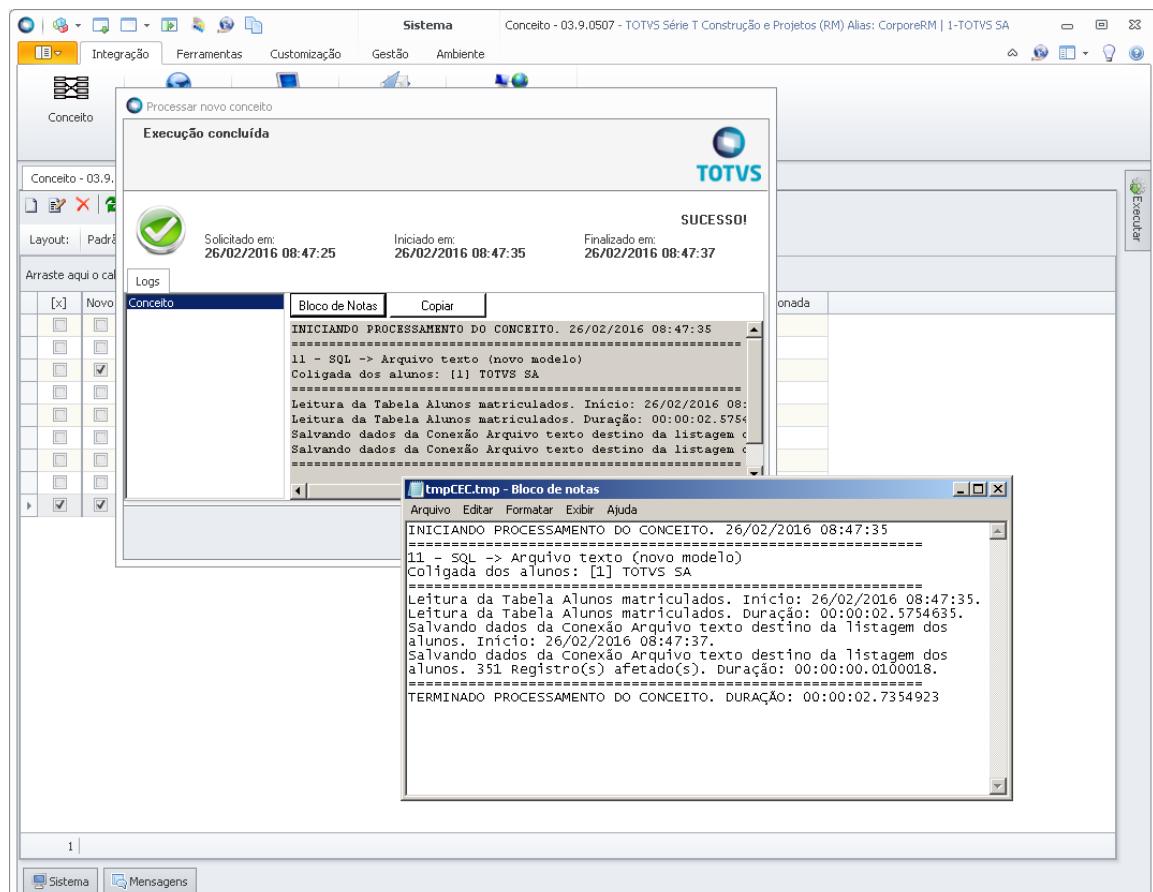
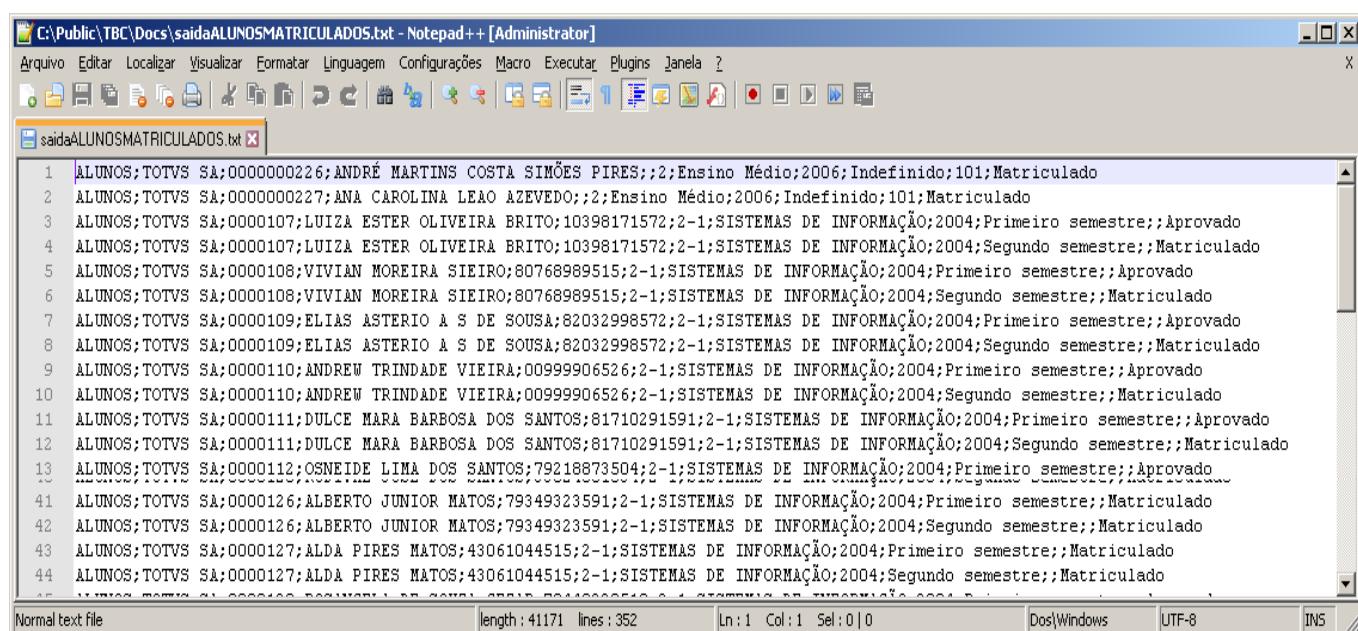


Ao selecionar o conceito a ser executado e acionar o processo será apresentado o wizard de processos onde ao navegar por este serão apresentados os parâmetros de execução do conceito caso este tenha parâmetros definidos. Lembre-se que em nosso exemplo criamos um parâmetro para o usuário selecionar a coligada a ser utilizada para filtrar os alunos.

Basta selecionar uma coligada e seguir no wizard até a execução do conceito.



Ao final da execução será apresentado o log com informações mais detalhadas.

The screenshot shows a Notepad++ window with the file "saídaALUNOSMATRICULADOS.txt" open. The file contains a large amount of text, mostly student records, with some lines numbered from 1 to 44. The text is in Portuguese and includes fields like student ID, name, and course information. The Notepad++ status bar at the bottom shows "Normal text file", "length: 41171 lines: 352", "In: 1 Col: 1 Sel: 0 | 0", "Dos\Windows", "UTF-8", and "INS".

```

1 ALUNOS;TOTVS SA;0000000226;ANDRÉ MARTINS COSTA SIMÕES PIRES;;2;Ensino Médio;2006;Indefinido;101;Matriculado
2 ALUNOS;TOTVS SA;0000000227;ANA CAROLINA LEAO AZEVEDO;;2;Ensino Médio;2006;Indefinido;101;Matriculado
3 ALUNOS;TOTVS SA;0000107;LUIZA ESTER OLIVEIRA BRITO;10398171572;2-1;SISTEMAS DE INFORMAÇÃO;2004;Primeiro semestre;;Aprovado
4 ALUNOS;TOTVS SA;0000107;LUIZA ESTER OLIVEIRA BRITO;10398171572;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
5 ALUNOS;TOTVS SA;0000108;VIVIAN MOREIRA SIEIRO;80768989515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Primeiro semestre;;Aprovado
6 ALUNOS;TOTVS SA;0000108;VIVIAN MOREIRA SIEIRO;80768989515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
7 ALUNOS;TOTVS SA;0000109;ELIAS ASTERIO A S DE SOUSA;82032998572;2-1;SISTEMAS DE INFORMAÇÃO;2004;Primeiro semestre;;Aprovado
8 ALUNOS;TOTVS SA;0000109;ELIAS ASTERIO A S DE SOUSA;82032998572;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
9 ALUNOS;TOTVS SA;0000110;ANDREW TRINDADE VIEIRA;00999906526;2-1;SISTEMAS DE INFORMAÇÃO;2004;Primeiro semestre;;Aprovado
10 ALUNOS;TOTVS SA;0000110;ANDREW TRINDADE VIEIRA;00999906526;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
11 ALUNOS;TOTVS SA;0000111;DULCE MARA BARBOSA DOS SANTOS;81710291591;2-1;SISTEMAS DE INFORMAÇÃO;2004;Primeiro semestre;;Aprovado
12 ALUNOS;TOTVS SA;0000111;DULCE MARA BARBOSA DOS SANTOS;81710291591;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
13 ALUNOS;TOTVS SA;0000112;OSNEIDE LIMA DOS SANTOS;79218873504;2-1;SISTEMAS DE INFORMAÇÃO;2004;Primeiro semestre;;Aprovado
14 ALUNOS;TOTVS SA;0000112;OSNEIDE LIMA DOS SANTOS;79218873504;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
15 ALUNOS;TOTVS SA;0000126;ALBERTO JUNIOR MATOS;79349323591;2-1;SISTEMAS DE INFORMAÇÃO;2004;Primeiro semestre;;Matriculado
16 ALUNOS;TOTVS SA;0000126;ALBERTO JUNIOR MATOS;79349323591;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
17 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Primeiro semestre;;Matriculado
18 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
19 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
20 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
21 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
22 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
23 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
24 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
25 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
26 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
27 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
28 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
29 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
30 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
31 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
32 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
33 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
34 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
35 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
36 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
37 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
38 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
39 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
40 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
41 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
42 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
43 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado
44 ALUNOS;TOTVS SA;0000127;ALDA PIRES MATOS;43061044515;2-1;SISTEMAS DE INFORMAÇÃO;2004;Segundo semestre;;Matriculado

```

### Exercício

Utilizando o Novo Conceito, acesse o cadastro de Conceito e crie o seguinte registro:

#### 2. Origem Banco de Dados (SQL) → Destino Arquivo XML

Neste exemplo, vamos exportar dados da tabela de Extrato de Caixa (FXCX) para um arquivo XML.

Sentença:

```
SELECT CODCOLIGADA,  
       CODFILIAL,  
       CODCXA,  
       NUMERODOCUMENTO,  
       VALOR,  
       HISTORICO  
  FROM   FXCX  
 WHERE  YEAR(DATA) > 2013  
 /*and*/
```

**Observação:** Utilize o arquivo XSD “FXCX” salvo na pasta “Arquivos treinamento TBC\Novo Conceito” .

## 9.10. Conceito com WebServices

A utilização de WebServices por meio da ferramenta Conceito tem aplicações diferentes, dependendo interface utilizada pelo usuário (Conceito “Antigo” ou Conceito “Novo”).

Utilizando o Conceito “Antigo” a aplicação se dá apenas quando o WebService for o destino dos dados, isto é, podemos ter como origem um conjunto de dados proveniente de uma tabela/SQL, arquivos texto/XML, DataServer e precisamos trabalhar estas informações acionando um serviço que está disponibilizado através de um webservice.

Trabalhando na interface mais nova temos uma aplicação mais abrangente. O WebService pode ser utilizado como destino e/ou origem dos dados, extraíndo e incluindo dados do RM por meio desta conexão.

A premissa para o consumo de webservices pelo conceito é que estes webservices devem prover o wsdl (uma espécie de contrato do serviço, inerente para o SOAP) e as requisições possam ser feitas via SOAP (1.1 e 1.2), GET ou POST.

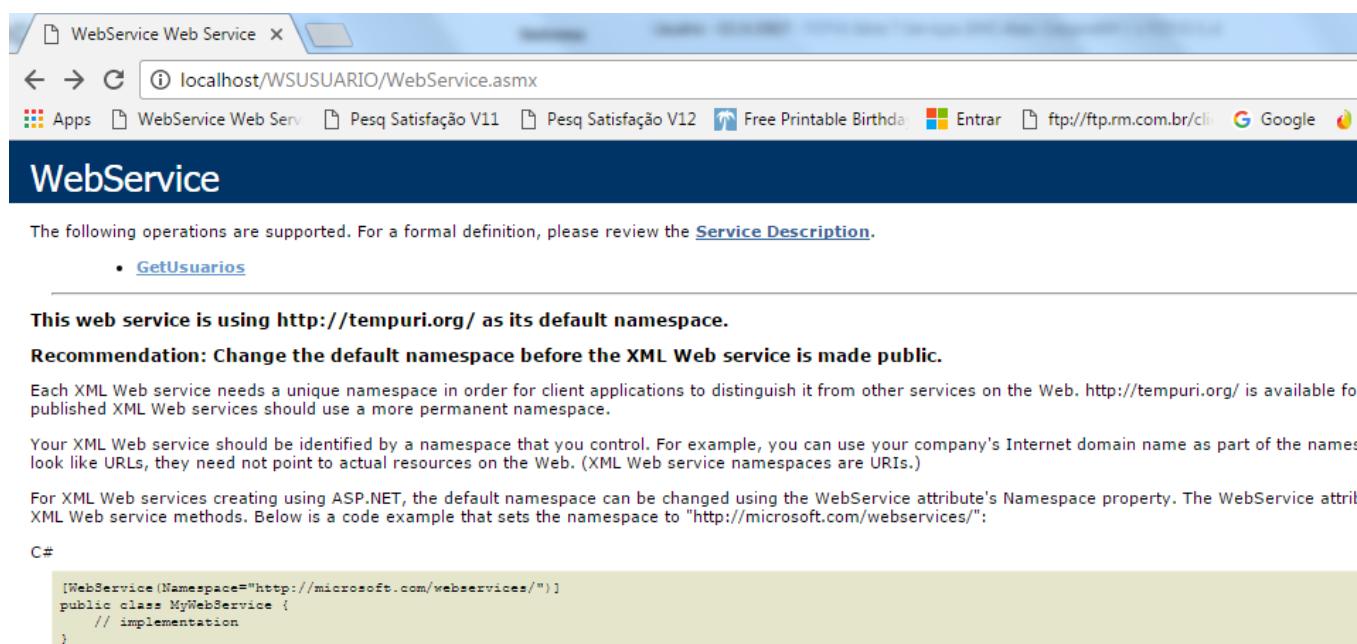
Através desta possibilidade, é possível que os clientes tenham serviços fornecidos por sistema de terceiros, customizados, criados por eles mesmos ou até mesmo através da internet (web) onde estes serviços já estejam disponíveis como webservices e já sejam domínio/consumo pela empresa e desejar expandir seu uso através de conceitos (integrar usuários, gerar lançamentos, movimentar estoque/vendas, abrir chamados, etc.).

### 9.10.1. Origem WebService de terceiros e Data Server

Para ilustrar a utilização de webservices pelo conceito, consumiremos um webservice criado para fornecer dados de novos usuários e incluiremos essas informações no RM por meio do DataServer de Usuários.

O webservice que disponibiliza tal serviço foi criado no padrão SOAP, contém apenas o método que retorna as informações de novos usuários contendo todos os campos necessários para o cadastro do RM.

Como origem vamos buscar as informações do webservice WSUSUARIO, conforme imagem abaixo.



The following operations are supported. For a formal definition, please review the [Service Description](#).

- [GetUsuarios](#)

This web service is using <http://tempuri.org/> as its default namespace.

**Recommendation:** Change the default namespace before the XML Web service is made public.

Each XML Web service needs a unique namespace in order for client applications to distinguish it from other services on the Web. <http://tempuri.org/> is available for published XML Web services should use a more permanent namespace.

Your XML Web service should be identified by a namespace that you control. For example, you can use your company's Internet domain name as part of the names look like URLs, they need not point to actual resources on the Web. (XML Web service namespaces are URIs.)

For XML Web services creating using ASP.NET, the default namespace can be changed using the WebService attribute's Namespace property. The WebService attribute XML Web service methods. Below is a code example that sets the namespace to "<http://microsoft.com/webservices/>:

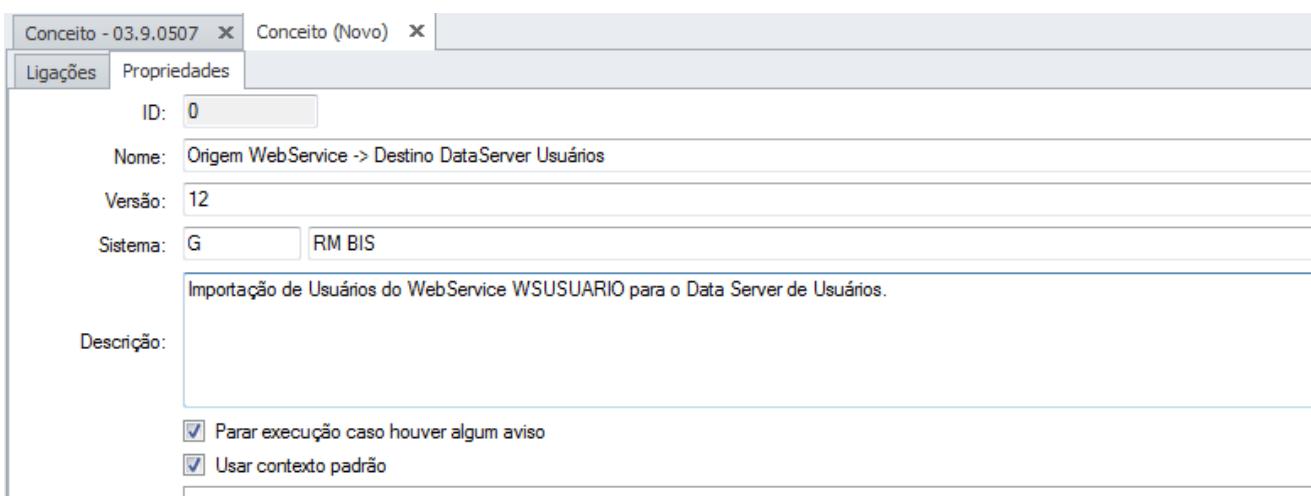
```
C#
[WebService(Namespace="http://microsoft.com/webservices/")]
public class MyWebService {
    // implementation
}
```

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
<?xml version="1.0" encoding="utf-8"?>
<ArrayOfUSUARIO xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://tempur/treinamento/WSUSUARIO" xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
  <USUARIO>
    <CODUSUARIO>treinamento1</CODUSUARIO>
    <NOME>treinamento1</NOME>
    <STATUS>1</STATUS>
    <DATAINICIO>2016-10-06T09:26:39.9526304-03:00</DATAINICIO>
    <CONFIRMBTNOK>1</CONFIRMBTNOK>
    <SENHA>123456</SENHA>
    <ULTIMACOLIGADA>1</ULTIMACOLIGADA>
    <CODACESSO>Acesso01</CODACESSO>
    <OBRIGAALTERARSENHA>F</OBRIGAALTERARSENHA>
    <NUMLOGININVALIDO>0</NUMLOGININVALIDO>
    <EMAIL>treinamento1@totvs.com.br</EMAIL>
    <ACESSONET>T</ACESSONET>
  </USUARIO>
  <USUARIO>
    <CODUSUARIO>treinamento2</CODUSUARIO>
    <NOME>treinamento2</NOME>
    <STATUS>1</STATUS>
    <DATAINICIO>2016-10-06T09:26:39.9526304-03:00</DATAINICIO>
    <CONFIRMBTNOK>1</CONFIRMBTNOK>
    <SENHA>123456</SENHA>
    <ULTIMACOLIGADA>1</ULTIMACOLIGADA>
    <CODACESSO>Acesso01</CODACESSO>
    <OBRIGAALTERARSENHA>F</OBRIGAALTERARSENHA>
    <NUMLOGININVALIDO>0</NUMLOGININVALIDO>
    <EMAIL>treinamento2@totvs.com.br</EMAIL>
    <ACESSONET>T</ACESSONET>
  </USUARIO>
</ArrayOfUSUARIO>
```

Vamos criar o conceito e parametrizar seus campos/propriedades de acordo com as instruções e imagens a seguir:

- Selecionando o botão CTRL, clique no botão Incluir para criar um novo registro utilizando o Novo Conceito.



- Neste exemplo preenchemos apenas os campos Nome, Versão, Sistema e Descrição conforme imagem acima.
- Após salvar o conceito vamos configurar as conexões origem e destino.

### Conexão Origem: WebService

- Neste exemplo utilizaremos as seguintes informações para configurar a conexão:

Endereço WSDL: <http://localhost/WSUSUARIO/WebService.asmx?Wsdl>

Protocolo: SOAP 1.2

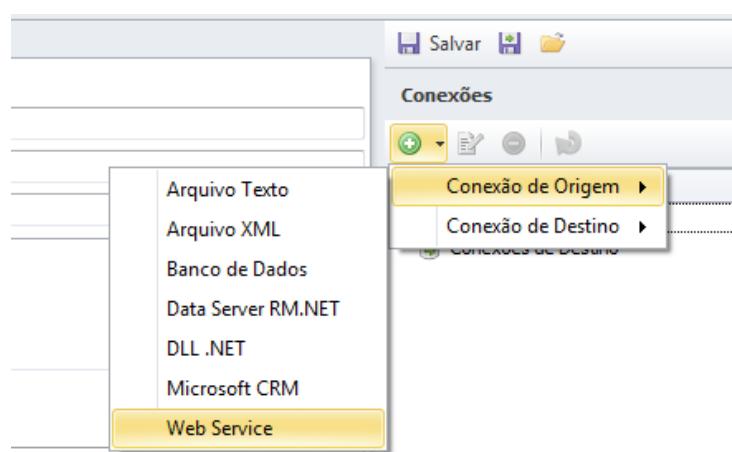
Usar autenticação padrão: Marcado

Usar proxy padrão: Marcado

Usar autenticação do proxy padrão

Método: RM.Con.Concept.Proxy.WebService.GetUsuarios

Para configurar a conexão origem adicione uma **Nova conexão | Conexão Origem | Web Service.**

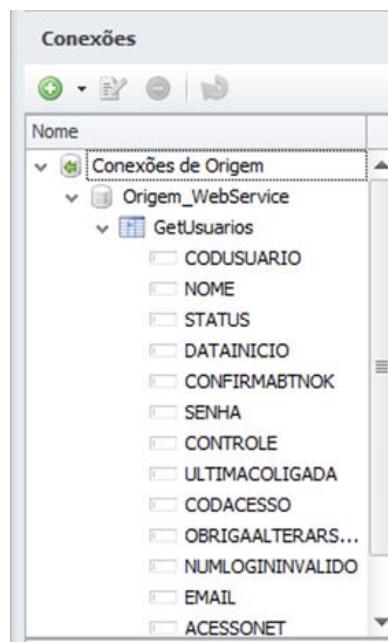


Neste exemplo utilizaremos as seguintes informações para configurar a conexão:

**Coneção Web Service: Novo**

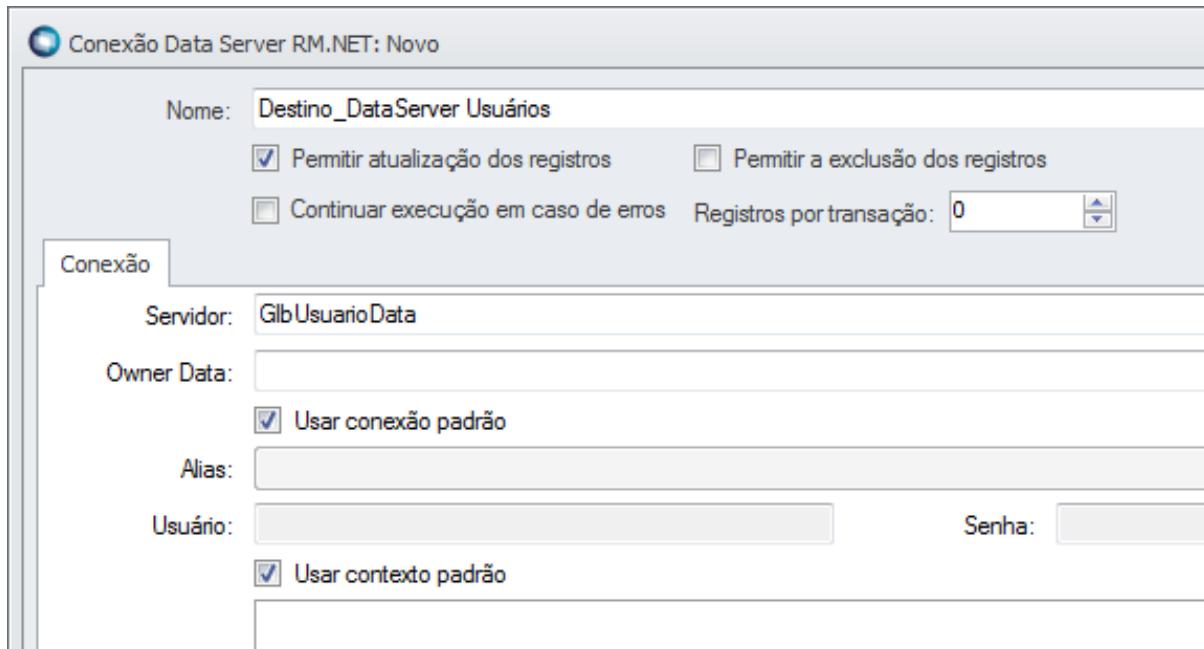
Nome:	Origem_WebService
<input type="checkbox"/> Continuar execução em caso de erros	
Conexão	Parâmetros
Endereço WSDL:	<input type="text" value="http://localhost/WSUSUARIO/WebService.asmx?Wsdl"/>
Protocolo:	<input type="text" value="SOAP 1.2"/>
<input checked="" type="checkbox"/> Usar autenticação padrão	
Usuário:	<input type="text"/>
<input checked="" type="checkbox"/> Usar proxy padrão	
Proxy:	<input type="text"/>
<input checked="" type="checkbox"/> Usar autenticação do proxy padrão	
Usuário:	<input type="text"/>
Método:	<input type="text" value="RM.Con.Concept.Proxy.WebService.GetUsuarios"/>

- Após finalizar a configuração acima, clique em "OK". Abaixo da conexão criada já aparecerá os campos disponíveis para realizar as ligações.



## Conexão Destino: DataServer Usuários

Para configurar a conexão destino adicione uma **Nova conexão | Conexão Destino | Data Server RM.NET**. Neste exemplo utilizaremos as seguintes informações para configurar a conexão:

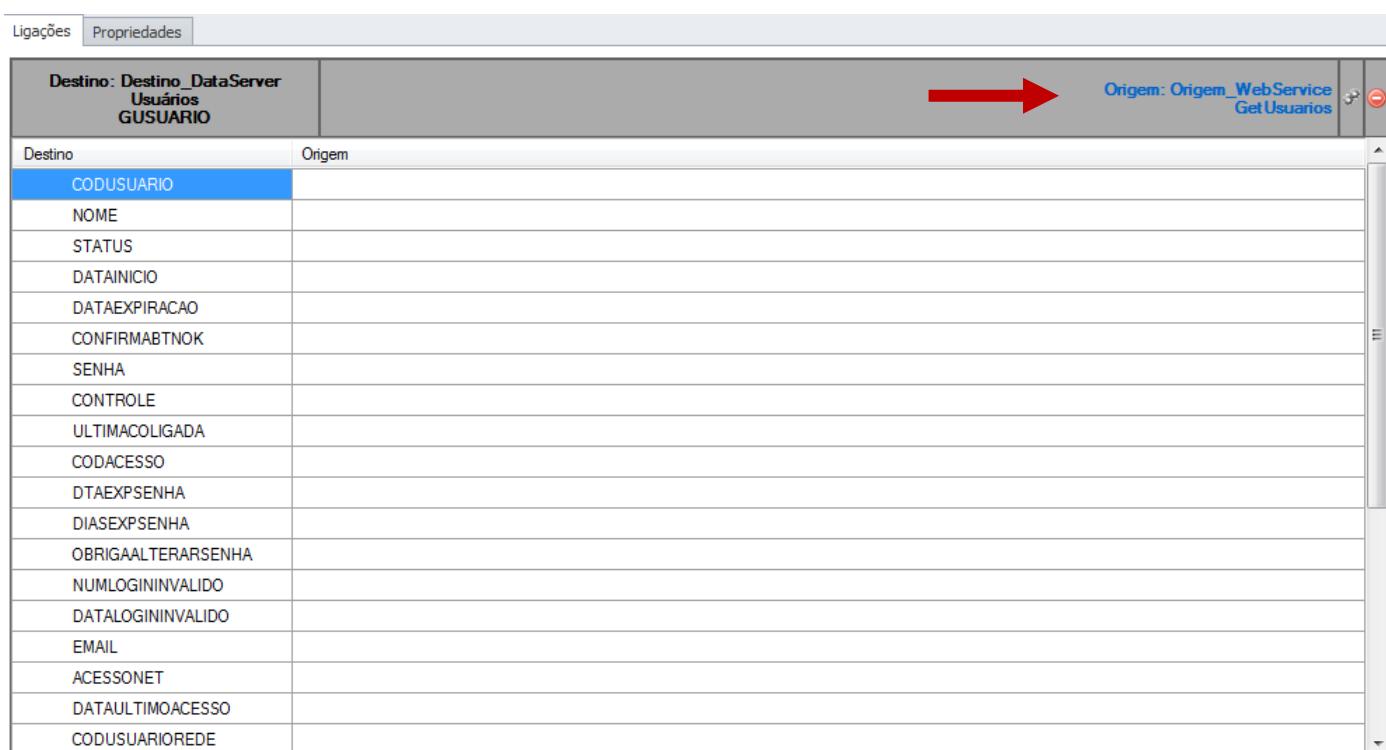


Após clicar em “OK” esta conexão já estará configurada.

## Ligações

Para finalizar a construção deste conceito, selecione a aba “Ligações”. Arraste a tabela destino (GUSUARIO) para esta aba e na sequência a tabela origem.

A tabela origem é arrastada para o canto direito superior da aba Ligações, conforme imagem abaixo.



Destino: Destino_DataServer Usuários GUSUARIO		Origem: Origem_WebService Get Usuarios
Destino	Origem	
CODUSUARIO		
NOME		
STATUS		
DATAINICIO		
DATAEXPIRACAO		
CONFIRMABTNOK		
SENHA		
CONTROLE		
ULTIMACOLIGADA		
CODACESSO		
DTAEXPSENHA		
DIASEXPSENHA		
OBRIGAALTERARSENHA		
NUMLOGININVALIDO		
DATALOGININVALIDO		
EMAIL		
ACESSONET		
DATAULTIMOACESSO		
CODUSUARIODE		

Faça a ligação de todos os campos disponíveis na tabela origem arrastando-os para os respectivos campos destino.

Após finalizar as ligações, salve o conceito e execute o processo “Processar novo conceito”.

 **SUCESO!**

Solicitado em: **29/09/2016 15:23:47**      Iniciado em: **29/09/2016 15:23:50**      Finalizado em: **29/09/2016 15:23:55**

**Logs**

Conceito	Bloco de Notas	Copiar
<b>INICIANDO PROCESSAMENTO DO CONCEITO.</b> 29/09/2016 15:23:51 <hr/> 39 - Origem WebService -> Destino DataServer Usuários <hr/> Leitura da Tabela GetUsuarios. Início: 29/09/2016 15:23:51. Leitura da Tabela GetUsuarios. Duração: 00:00:02.6310551. Salvando dados da Conexão Destino_DataServer Usuários. Início: Salvando dados da Conexão Destino_DataServer Usuários. 2 Reg: <hr/> <b>TERMINADO PROCESSAMENTO DO CONCEITO. DURAÇÃO:</b> 00:00:03.873081		

Acesse o cadastro de Usuários e verique os registros. Os usuários treinamento1 e treinamento2 foram criados.

## 10. WebServices do TOTVS Business Connect

Conforme você pode observar no diagrama apresentado na introdução, os WebServices do TBC são uma das formas de integrar o TOTVS RM com os demais sistemas seja ele como fonte dos dados ou então o destino, resultante de entrada direta dos dados (similar a cadastros) ou então a partir da execução de processos e estes gerarem as informações nas entidades.

A adoção de WebServices na integração de sistemas permite uma integração e comunicação mais ágil, dinâmica e segura entre as aplicações além de se utilizar de protocolos padrões e amplamente difundido nos tráfego de informações. Os WebServices do TBC podem ser hospedados no Internet Information Services (IIS) ou no RM. Host.

Atualmente o Totvs Business Conector disponibiliza os WebServices abaixo e todos os seus métodos (WebMethods) utilizam autenticação SOAP.

- DataServers do TOTVS RM – wsDataServer;
- Processos do TOTVS RM – wsProcess;
- TOTVS Gestão de Relacionamento com o Cliente – wsCRMAtendimento;
- Conceitos do TOTVS Business Connect – wsConConceito;
- TOTVS Message (ByYouESB) – wsTOTVSMessages;
- TOTVS Educacional – wsEdu;
- TOTVS Obras e Projetos – wsPrj;
- Lançamento Financeiro – wsFin;
- Movimentos – wsMov.

Mais detalhes podem ser encontrados em: <http://tdn.TOTVS.com/pages/releaseview.action?pageId=185748469>

## 10.1. Data Server (wsDataServer)

Os DataServer são os que mantém as regras de negócio do TOTVS RM a nível de entidades (cadastros) e o webservice wsDataServer disponibiliza o acesso direto a tais objetos de negócio.

Os métodos disponibilizados por este webservice são os seguintes:

- AutenticaAcesso: Autentica o usuário no TOTVS RM. O usuário e senha devem ser informados via SOAP, criando um Token para isto;
- AutenticaAcessoAuth: Autentica o usuário no ambiente TOTVS RM;
- GetSchema ou GetSchemaAuth: Faz a leitura das definições de um DataServer (regra de negócio TOTVS RM) informado. Retorna um XSD com a estrutura das tabelas, campos, etc. do DataServer. A partir deste XSD é que deverão ser montados os XMLs para a inserção de registros (SaveRecord);
- ReadRecord ou ReadRecordAuth: Faz a leitura do DataServer (regra de negócio TOTVS RM) e retorna o registro específico (em uma estrutura XML), conforme chave primária;
- ReadView ou ReadViewAuth: Faz a leitura do DataServer (regra de negócio TOTVS RM) e retorna conjunto de registros (em uma estrutura XML), conforme definido no filtro;
- SaveRecord ou SaveRecordAuth: Insere/Altera um registro no DataServer (regra de negócio TOTVS RM). Este método somente irá incluir o registro se este for válido de acordo com as regras de negócio definidas pelo DataServer. Os valores são passados ao DataServer através de uma estrutura XML que pode ser obtida a partir do XSD retornado pelo GetSchema.

**OBSERVAÇÃO:** Os *WebMethods* que possuem no final do nome a expressão “Auth”, não utilizam Token e podem ser consumidos através do Browser.

Lista de DataServers disponíveis para os métodos acima pode ser obtida através do link <http://tdn.TOTVS.com/download/attachments/143655277/DataServer+para+WebServices.rar?version=1&modificationDate=1398882522704>, lembrando que tais DataServer estão em constante evolução a cada versão do TOTVS RM.

## 10.2. Processos (wsProcess)

Assim como os DataServer mantém os objetos de negócios do TOTVS RM, os processos são serviços responsáveis por, a partir da entrada e/ou recuperação de informações, trabalhar os dados de acordo com regras de negócio e produzir uma saída seja ela como dados em sistemas, relatórios, informações de BI, etc.

O TOTVS RM é “recheado” de processos em todos os módulos e é possível executar alguns através do uso deste webservice. Para consumi-los é necessário enviar uma requisição um envelope SOAP passando os parâmetros de cada serviço/processo.

O método disponível para a execução do processo é:

- ExecuteWithXmlParams: Serviço responsável por executar processos com parâmetros XML.  
Os parâmetros solicitados pelo ExecuteWithXmlParams são:
  - ProcessServerName: Nome do server do processo;
  - strXmlParams: Parâmetros, em uma estrutura XML, para execução o processo

**OBSERVAÇÃO:** Os outros métodos expostos foram descontinuados, e encontram-se disponíveis somente por motivo de compatibilidade.

### 10.3. TOTVS Message - ByYou ESB (wsTOTVSMessages)

Se trata de um serviço utilizado para comunicação com o ByYou ESB e é utilizado em algumas integrações internas das aplicações da família TOTVS. É disponibilizado como integrante do TBC pois este é o gerenciador de integrações do TOTVS RM.

Seus métodos são os listados a seguir:

- AutenticaAcesso: Autentica o usuário no ambiente TOTVS RM. O usuário e senha devem ser informados via SOAP criando um Token para isto;
- ReceberArquivo: recebe uma mensagem XML do ESB e grava na fila do TOTVS Business Conector para posterior processamento. No header do XML estarão contidas informações que identificam a mensagem, inclusive qual o DataServer (regra de negócio TOTVS RM) irá processar a inclusão no TOTVS RM.

### 10.4. TOTVS Educacional (wsEdu)

Este serviço disponibiliza métodos para integração com o TOTVS Educacional.

Métodos:

- AutenticaAcesso: Autentica o usuário no ambiente TOTVS RM. O usuário e senha devem ser informados via SOAP criando um Token para isto.
- ImprimeBoleto: Retorna o Stream do boleto para impressão.  
É necessário informar os parâmetros:
  - ListarBoletos: Lista boletos em aberto.

### 10.5. TOTVS Obras e Projetos (wsPrj)

Este webservice disponibiliza serviços para integração com o TOTVS Obras e Projetos.

Os serviços disponibilizados são representados pelos métodos a seguir:

- AutenticaAcesso: Autentica o usuário no ambiente RM. O usuário e senha devem ser informados via SOAP criando um Token para isto.
- ApropriarMovimentoTituloAPagar: Apropriação de custos de um título a pagar.

### 10.6. TOTVS Gestão de Relacionamento com Cliente (wsCRMAtendimento)

WebService voltado a prover funcionalidades do módulo de Gestão de relacionamento com o cliente do TOTVS RM. Este serviço disponibiliza métodos para integração com o TOTVS Gestão de Relacionamento com Cliente.

Sua utilização pode ser através de URAs eletrônicas (centrais telefônicas, sites de relacionamento, fóruns) ou então sistemas que desejam integrar com a abertura de atendimentos/chamados (solicitações) e gerir estes sejam eles através de status dos atendimentos e/ou através de fluxos de tarefas.

Os métodos disponíveis são:

- AutenticaAcesso: Autentica o usuário no ambiente TOTVS RM. O usuário e senha devem ser informados via SOAP criando um Token para isto;

- AvancaEtapaAtendimento: Executa o processo de avançar etapa do atendimento com todos os seus procedimentos, regras e fórmulas, encaminhando o atendimento a princípio para a etapa informada (pode ser alterado por alguma fórmula);
- AvancaEtapaAtendimentoSetResponsavel: Executa o processo de avançar etapa do atendimento com todos os seus procedimentos, regras e fórmulas, encaminhando o atendimento a princípio para a etapa e atendente informados (pode ser alterado por alguma fórmula);
- BuscaAtendenteByUsuario: Atendente que está associado ao usuário informado;
- BuscaAtendenteContatoCliente: Atendente que é o responsável, neste atendimento, por manter contato com o cliente para trocar informações sobre a solicitação constante no atendimento;
- BuscaAtendentesByRamal: Lista de atendentes que possuem, no cadastro de atendentes, o campo ramal informado;
- BuscaAtendenteResponsavelPeloAtendimento: Atendente atual que se encontra com o atendimento;
- BuscaRamalAtendente: Ramal do atendente;
- BuscaDadosClienteByCPFCNPJ: Lista de clientes e suas informações cujo CPF e/ou CNPJ combinem com o informado;
- BuscaDadosClienteByTelefone: Lista de clientes e suas informações cujo telefone combina com o informado;
- BuscaDadosContatoClienteByTelefone: Lista de contatos e suas informações (algumas informações do cliente do contato) cujo telefone combina com o informado;
- BuscaEmailAtendente: e-mail do atendente;
- BuscaRamalUsuario: Ramal do usuário (que deve estar associado a um atendente);
- CriarAtendimento: Cria um novo atendimento no TOTVS Gestão do Relacionamento com o Cliente;
- CriarAtendimentoCliente: Cria um novo atendimento no TOTVS Gestão do Relacionamento com o Cliente;
- CriarAtendimentoSimples: Cria um novo atendimento no TOTVS Gestão do Relacionamento com o Cliente;
- CriarAtendimentoWorkflow: Cria um novo atendimento no TOTVS Gestão do Relacionamento com o Cliente;
- EncaminharAtendimentoParaAtendente: Repassa/Encaminha um determinado atendimento para um atendente ou grupo de atendente possibilitando prover algumas informações adicionais;
- EncaminharAtendimentoParaUsuario: Repassa/Encaminha um determinado atendimento para um atendente ou grupo de atendente possibilitando prover algumas informações adicionais;
- SendMail: Envia e-mail;

### 10.7. Conceitos do TOTVS Business Connect (wsConConceito)

Neste treinamento nos aprofundamentos na ferramenta de conceito apresentando o seu potencial.

Trabalhando em conjunto com tal ferramenta, este webservice disponibiliza um serviço que permite a execução de um conceito previamente criado dando mais flexibilidade às integrações.

A ferramenta conceito e os WebServices do TBC são independentes, mas tanto podem usar um webservice como destino de um conceito como pode usar o “wsConConceito” para executar um conceito existente.

Serviços:

- AutenticaAcesso: Autentica o usuário no ambiente RM. O usuário e senha devem ser informados via SOAP criando um Token para isto;
- ExecutarConceito: Executa um conceito pre-cadastrado, conforme o identificador informado.

### 10.8. Lançamento Financeiro (wsFin)

Este WebService do TBC proporciona serviços para realizar “pré-lançamentos” financeiros no TOTVS Gestão Financeira.

O serviço não comunicará diretamente com o módulo financeiro e sim será preenchido um DataServer do TBC onde os dados serão pre-validados e entregue ao financeiro no formato esperado.

Como temos na TOTVS a diretriz de padronizar as integrações então todos os produtos TOTVS devem trabalhar com estruturas XMLs únicas evitando, desta forma, o processo de transformação de mensagens. Baseando nisto o serviço aqui disponibilizado recebe como parâmetro uma estrutura XML e este por sua vez será validado, transformado e por fim enviado a base de dados do TOTVS RM.

Os serviços disponibilizados através deste webservice são: Inclusão, Baixa e Exclusão de Lançamentos Financeiros

**OBSERVAÇÃO:** A alteração de lançamentos não está contemplada. Em alternativa pode ser feita uma exclusão seguida de uma inclusão com os dados já alterados.

**ATENÇÃO:** A base de dados deve obrigatoriamente possuir o Novo Modelo de Baixa (Para o cliente possuir este recurso o mesmo deve entrar em contato com o Suporte do TOTVS RM, produto Gestão Financeira), caso o cliente tente utilizar este WebService no modelo antigo de baixa, o recurso não irá funcionar.

### 10.9. Movimentos no Gestão de Estoque, Compras, Faturamento (wsMovn)

Este serviço disponibiliza métodos para integração com Movimentos do TOTVS Gestão de Estoque, Compras e Faturamento.

Os serviços disponibilizados através deste webservice são: Inclusão, Alteração e Exclusão de movimentos do módulo de estoque, compras e faturamento do TOTVS RM.

O método responsável por estas funcionalidades é o “SaveMovimento” que recebe um XML com as informações do movimento juntamente com o tipo de operação (inclusão, alteração, exclusão).

## 11. Conclusão

Ao finalizar o treinamento verificamos que através das ferramentas do TOTVS Business Conector, sejam elas individualmente ou complementares, é possível a integração entre entidades (Cadastros/Formulários/Tabelas/arquivos) distintas para integração do ERP com bases e sistemas legados.

O TOTVS RM, através do TBC, oferece aos clientes e parceiros serviços em WebService para busca e atualização de registros. Estes WebServices seguem a mesma regra negócio do registro alterado e recuperado via interface padrão RM. Com isto, nossos parceiros e clientes podem integrar sistemas de terceiros de forma simples e rápida.

Com inúmeros recursos poderosos e uma interface visual intuitiva, o TBC consegue tornar o trabalho de integrar outras soluções às soluções TOTVS (e/ou terceiros) em uma tarefa muito simples.

## 12. Anexos

### 12.1. Como expor os WebServices do TOTVS Business Connect através do Host (recomendado)

Este tópico tem por objetivo demonstrar o procedimento para configurar o Host para disponibilizar os WebServices responsáveis pela integração de aplicativos de terceiros com o TOTVS RM. Os webservices disponíveis são:

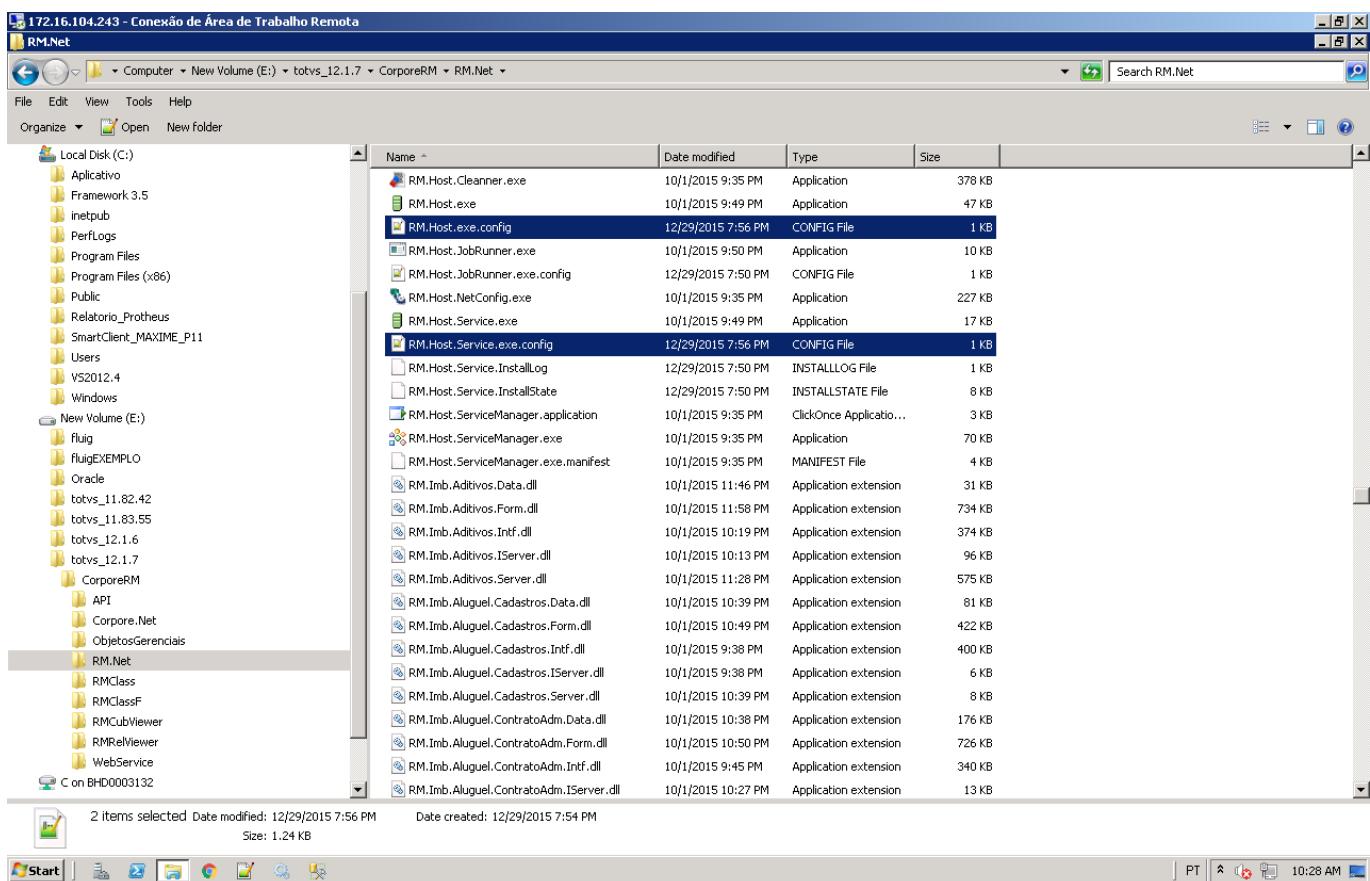
- wsDataServer;
- wsConceito;
- wsProcess;
- wsTotvsMessage.

Cada webservice possui os seus respectivos métodos.

O webservice indicado para consulta de cadastro e inclusão/alteração/exclusão de um cadastro é o wsDataServer. Já o indicado para executar um processo (faturar movimento, cancelar movimento, cancelar lançamento, etc.) é o wsProcess.

Procedimentos:

1. Edite o arquivo RM.Host.Service.exe.config e RM.Host.exe.config, que está na pasta <diretório de instalação do RM>\RM.Net:



2. Adicione as seguintes TAGs no bloco de instruções <appSettings> ... </appSettings>

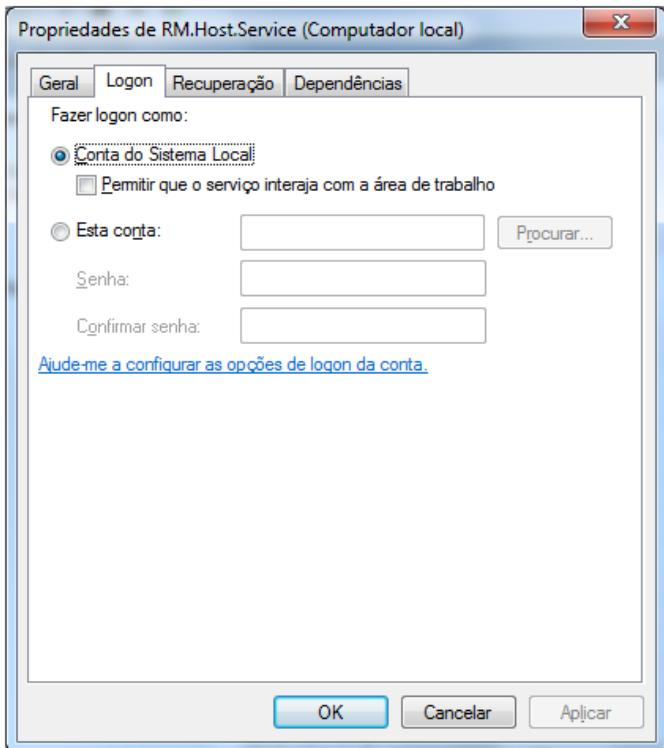
```
<add key="DEFAULTDB" value="CorporeRM" />
<add key="HTTPPORT" value="8051" />
```

**OBSERVAÇÃO:** No valor da tag DefaultDB deve ser informado qual será o Alias, do arquivo Alias.dat que contém a informação da base de dados que será utilizada. No valor da tag HTTPPORT deve ser informada qual será a porta a ser utilizada pelo Host para os serviços via WS (geralmente é informado o padrão que é 8051).

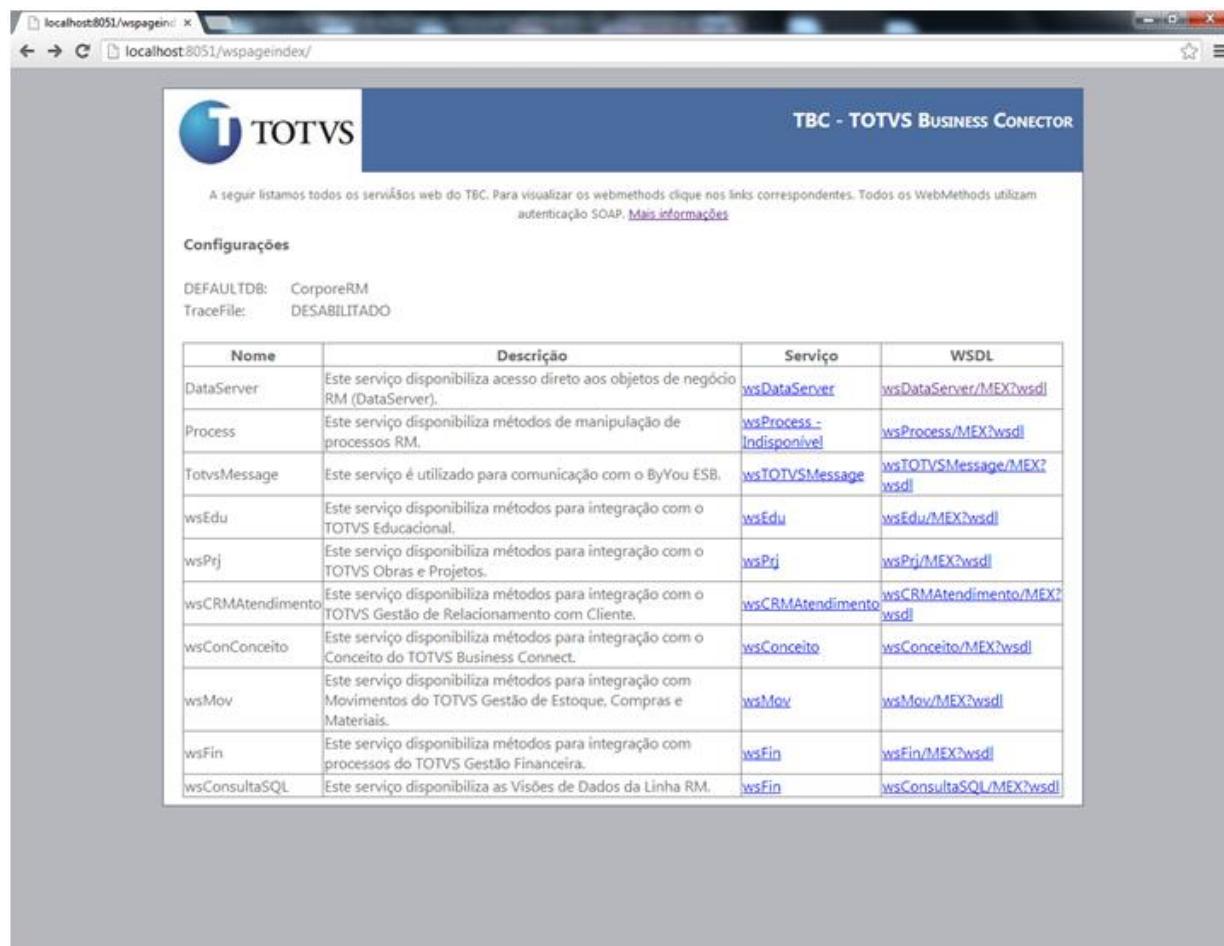
3. Abaixo segue um arquivo de exemplo de configuração do Host:

```
<?xml version="1.0"?>
<configuration>
    <appSettings>
        <add key="JobServer3Camadas" value="false" />
        <add key="Port" value="8050" />
        <add key="ActionsPath" value="C:\TOTVS1152\CorporeRM\RM.Net;C:\TOTVS1152\CorporeRM\Corpore.Net\Bin" />
        <add key="EnableCompression" value="true" />
        <add key="DEFAULTDB" value="CorporeRM" />
        <add key="HTTPPORT" value="8051" />
    </appSettings>
    <runtime>
        <NetFx40_LegacySecurityPolicy enabled="true" />
    </runtime>
    <system.runtime.remoting>
        <customErrors mode="Off" />
    </system.runtime.remoting>
    <startup>
        <supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.0" />
    </startup>
</configuration>
```

4. Alterar a conta de autenticação do serviço do RM.Host, no gerenciador de serviços do Windows, para utilizar a conta LOCALSYSTEM ou de um usuário válido na rede, conforme print abaixo:



5. Para verificar se as configurações estão corretas informe a URL abaixo no browser (Internet Explorer, Mozilla Firefox, etc.):  
`http://<ip ou hostname da máquina em que o Host está configurado>:<porta que foi informada no arquivo de configuração do host, na tag HTTPPORT>/wsPageIndex`



A seguir listamos todos os serviços web do TBC. Para visualizar os webmethods clique nos links correspondentes. Todos os WebMethods utilizam autenticação SOAP. [Mais informações](#)

**Configurações**

Nome	Descrição	Serviço	WSDL
DataServer	Este serviço disponibiliza acesso direto aos objetos de negócio RM (DataServer).	wsDataServer	wsDataServer/MEX?wsdl
Process	Este serviço disponibiliza métodos de manipulação de processos RM.	wsProcess - Indisponível	wsProcess/MEX?wsdl
TotvsMessage	Este serviço é utilizado para comunicação com o ByYou ESB.	wsTOTVSMassage	wsTOTVSMassage/MEX?wsdl
wsEdu	Este serviço disponibiliza métodos para integração com o TOTVS Educacional.	wsEdu	wsEdu/MEX?wsdl
wsPrj	Este serviço disponibiliza métodos para integração com o TOTVS Obras e Projetos.	wsPrj	wsPrj/MEX?wsdl
wsCRMAtendimento	Este serviço disponibiliza métodos para integração com o TOTVS Gestão de Relacionamento com Cliente.	wsCRMAtendimento	wsCRMAtendimento/MEX?wsdl
wsConConceito	Este serviço disponibiliza métodos para integração com o Conceito do TOTVS Business Connect.	wsConceito	wsConceito/MEX?wsdl
wsMov	Este serviço disponibiliza métodos para integração com Movimentos do TOTVS Gestão de Estoque, Compras e Materiais.	wsMov	wsMov/MEX?wsdl
wsFin	Este serviço disponibiliza métodos para integração com processos do TOTVS Gestão Financeira.	wsFin	wsFin/MEX?wsdl
wsConsultaSQL	Este serviço disponibiliza as Visões de Dados da Linha RM.	wsFin	wsConsultaSQL/MEX?wsdl

**OBSERVAÇÃO:** Para consumir os webservices deve utilizar um aplicativo que utilize protocolo Soap nas chamadas. Indicamos o SoapUI. Esse software pode ser baixado pelo link:  
<http://sourceforge.net/projects/soapui/files/>

## 12.2. Como expor os WebServices do TotvsBusinessConnect através do IIS

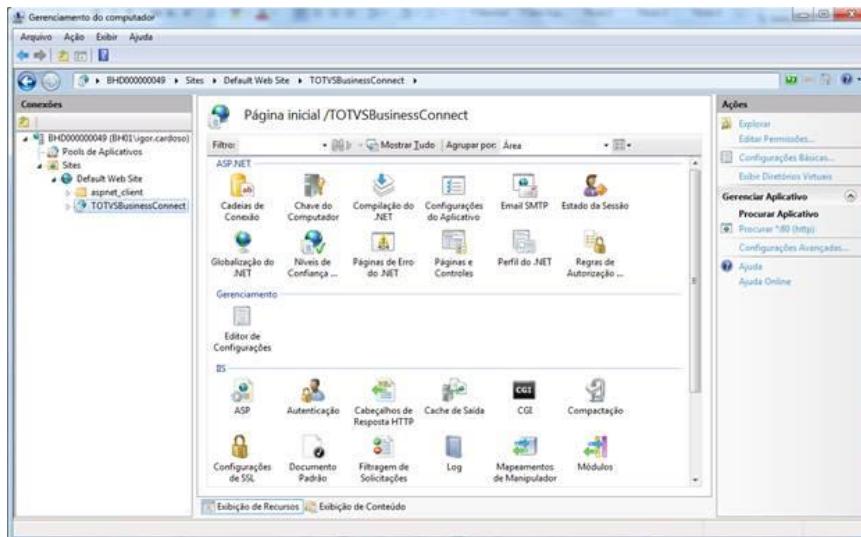
Caso queira configurar o WebServices para uma integração com aplicativos de terceiros, ou seja, um sistema legado, deve-se utilizar os serviços do TBC hospedados ou no Host do TOTVS RM ou no IIS do Windows (Internet Information Services).

Então para o uso destes serviços através de seus webservice alguns requisitos são necessários:

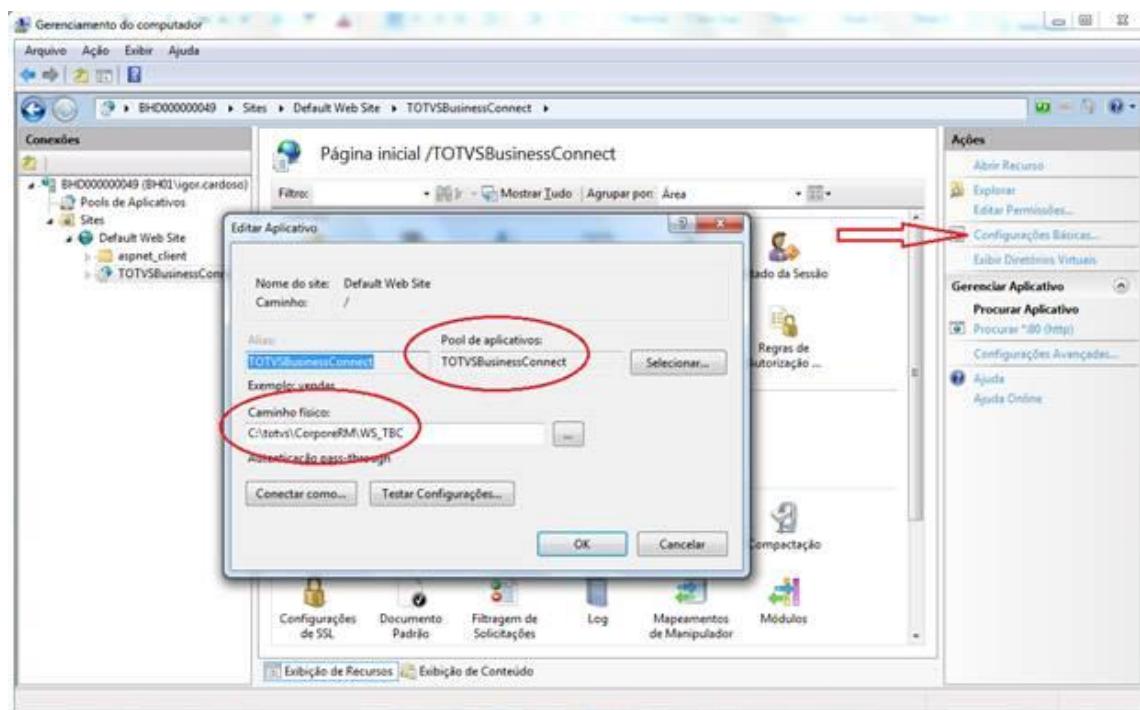
- Instalação do TOTVS Business Connect;
- Internet Information Service (IIS) instalado e devidamente configurado (as restrições CGI e ISAPI devem estar liberadas para o Asp Net 4.0).

A instalação padrão do TBC incluirá um aplicativo no IIS

## TOTVS Business Conector

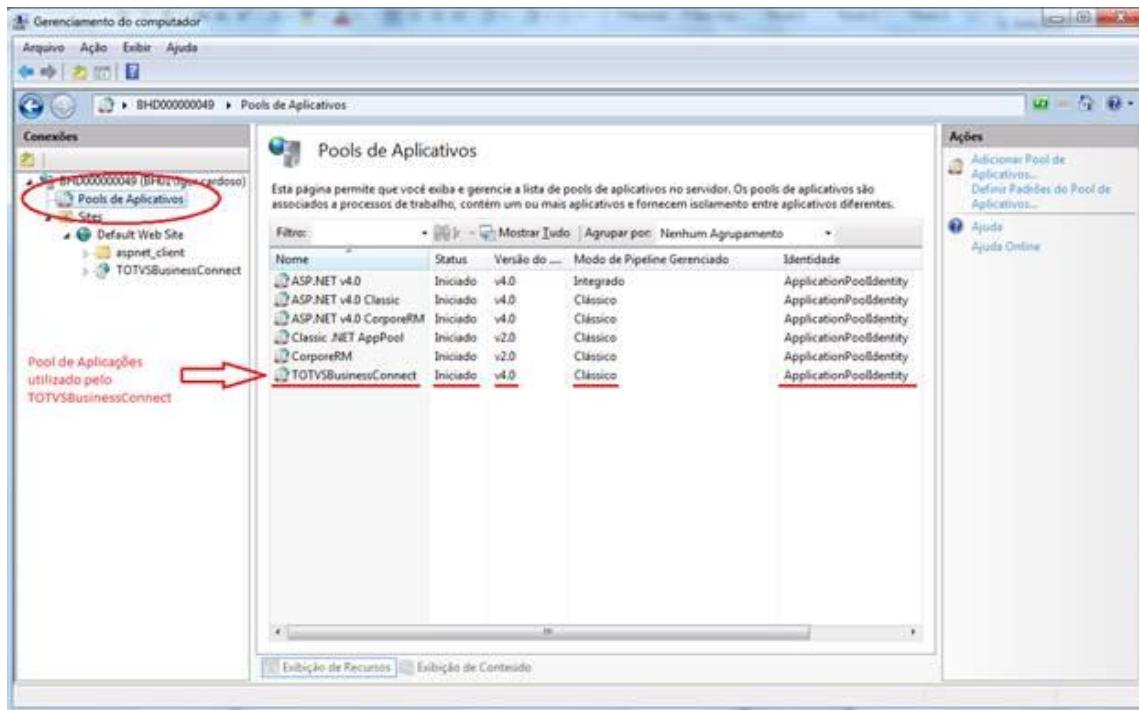


1. Verifique nas configurações básicas do aplicativo se o "Caminho Físico" está apontando para o caminho onde o TBC foi instalado, e qual o Pool de Aplicativos está selecionado:

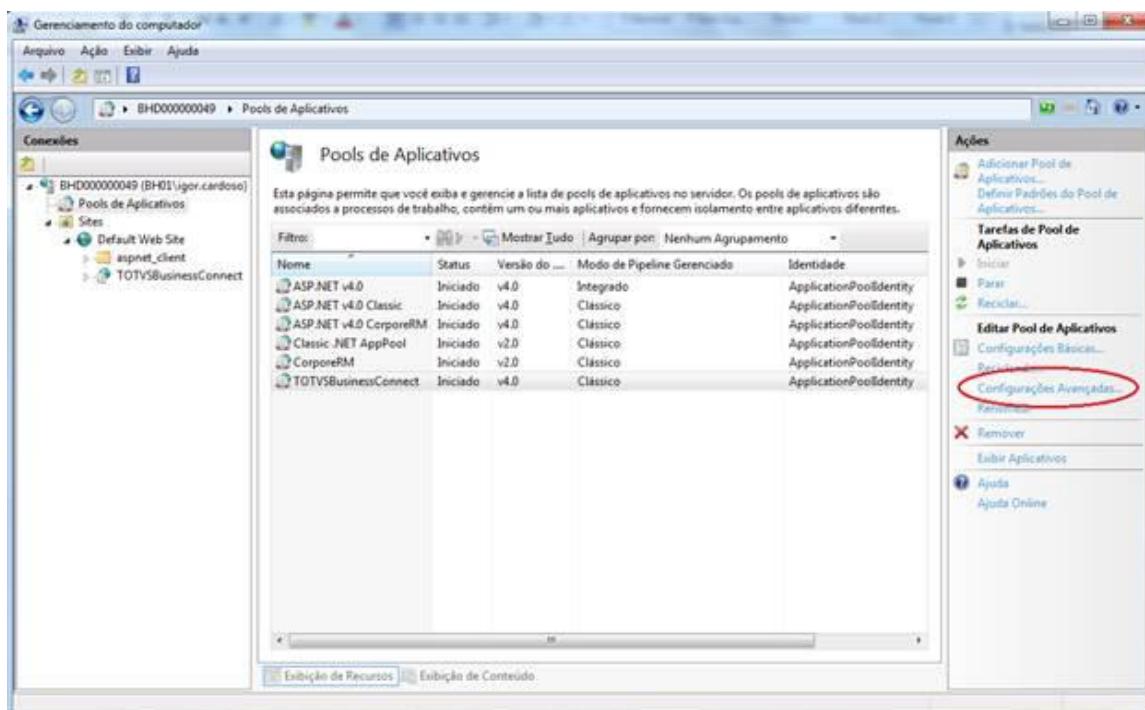


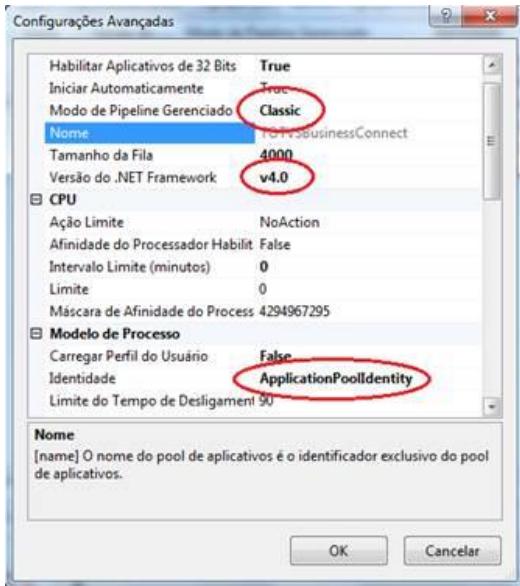
2. Nas configurações dos Pools, verifique se o Pool de Aplicativos utilizado está com as seguintes propriedade e seus respectivos valores:

- Status = Iniciado;
- Versão FrameWork = v4.0;
- Modo de Pipeline = Clássico;
- Identidade = "ApplicationPoolIdentity";



Caso não esteja as altere acessando a opção de “Configurações avançadas” do pool;





3. No arquivo Web.config existente no <diretório de instalação>\WS\_TBC deve ser alterada as seguintes TAGs:

- <add key="Host" value="localhost"/>  
Informar o hostname ou ip da máquina que contém o serviço do Host do RM;
- <add key="Port" value="8050"/>  
Informar a porta em que o serviço do Host do RM está sendo executado;
- <add key="DBAlias" value="CORPORERM"/>  
Informar o Alias, do arquivo Alias.dat, que contém a informação de qual base será utilizada na integração;

Alterar a linha abaixo:

```
<authorization>
  <deny users= " ? " />
</authorization>
```

Para:

```
<authorization>
  <allow users= " * " />
</authorization>
```

**OBSERVAÇÃO:** Os arquivos “<diretório de instalação do RM>\WS\_TBC\Web.Config” e “<diretório de instalação do RM>\RM.Net\RM.Host.Service.exe.config” devem estar idênticos para as seguintes informações abaixo:

```
<add key="JobServer3Camadas" value="false" />
<add key="EnableCompression" value="false" />
```

4. Os WebServices serão disponibilizados na porta 80 (como é a porta padrão pode ser omitida na url). Mas o administrador do sistema poderá alterar esta e demais configurações, se desejado, editando o arquivo Web.config que se encontra no diretório de instalação RM na pasta WS\_TBC: (verificar as configurações JOSERVER3CAMADAS, DABALIAS, HOST e PORTA de acordo com as configurações do RM.Host.Service.exe.config ou RM.Host.exe.config);

Web.config - Bloco de notas

```

Arquivo Editar Formatar Exibir Ajuda

<add type="Microsoft.web.Services3.
    </soapExtensionImporterTypes>
    <soapServerProtocolFactory type="Microsoft.
        </webServices>
        <pages/>
    <!--Inicio - Tags utilizadas para autenticação do tipo "HTTP Ba
<httpModules>
    <add name="BasicAuthenticationModule" type="RM.Con.Conektor.w
</httpModules>
<authentication mode="None"/>
<authorization>
    <allow users="*"/>
</authorization>
<!--Fim-->
</system.web>
<appSettings>
    <add key="RM.Con.Conektor.wsSecurity.BasicAuthenticationModule_
    <add key="RemoteLoad" value="false"/>
    <add key="SelfHosted" value="false"/>
    <add key="Host" value="localhost"/>
    <add key="Port" value="8050"/>
    <add key="DBAlias" value="CorpoRM"/>
    <add key="JobServer3Camadas" value="false"/>
    <add key="EnableCompression" value="false" />
</appSettings>
</Microsoft.web.services3>

```

5. A partir de então, após tais configurações/parametrizações, os webservices do TBC já podem ser acessados, por exemplo através do navegador Internet Explorer:



Nome	Descrição	Serviço	WSDL
wsDataServer	Este serviço disponibiliza acesso direto aos objetos de negócio RM (DataServer).	<a href="#">wsDataServer.asmx</a>	<a href="#">wsDataServer.asmx?wsdl</a>
wsProcess	Este serviço disponibiliza métodos de manipulação de processos RM.	<a href="#">wsProcess.asmx - indisponível</a>	<a href="#">wsProcess.asmx?wsdl</a>
wsTOTSMessage	Este serviço é utilizado para comunicação com o ByYou ESB.	<a href="#">wsTOTSMessage.asmx</a>	<a href="#">wsTOTSMessage.asmx?wsdl</a>
wsEdu	Este serviço disponibiliza métodos para integração com o TOTVS Educacional.	<a href="#">wsEdu.asmx</a>	<a href="#">wsEdu.asmx?wsdl</a>
wsPrj	Este serviço disponibiliza métodos para integração com o TOTVS Obras e Projetos.	<a href="#">wsPrj.asmx</a>	<a href="#">wsPrj.asmx?wsdl</a>
wsCRMAtendimento	Este serviço disponibiliza métodos para integração com o TOTVS Gestão de Relacionamento com Cliente.	<a href="#">wsCRMAtendimento.asmx</a>	<a href="#">wsCRMAtendimento.asmx?wsdl</a>
wsConConceito	Este serviço disponibiliza métodos para integração com o Conceito do TOTVS Business Conector.	<a href="#">wsConConceito.asmx</a>	<a href="#">wsConConceito.asmx?wsdl</a>
wsMov	Este serviço disponibiliza métodos para integração com Movimentos do TOTVS Gestão de Estoque, Compras e Materiais.	<a href="#">wsMov.asmx</a>	<a href="#">wsMov.asmx?wsdl</a>
wsFin	Este serviço disponibiliza métodos para integração com processos do TOTVS Gestão Financeira.	<a href="#">wsFin.asmx</a>	<a href="#">wsFin.asmx?wsdl</a>
wsConsultaSQL	Este serviço disponibiliza métodos para integração com Consulta SQL do RM.	<a href="#">wsConsultaSQL.asmx</a>	<a href="#">wsConsultaSQL.asmx?wsdl</a>

## 12.3. DataServers do TOTVS RM

Os DataServers podem ser utilizados em conceitos cuja conexão de origem e/ou destino sejam do tipo DataServer bem como pode ser utilizado pelo webservice wsDataServer em uma integração direta com aplicativos de terceiros, ou seja, um sistema legado.

Conforme mencionado, os serviços via DataServers disponibilizam acesso direto aos objetos/classes de negócio do TOTVS RM.

Para o consumo do webservice wsDataServer temos os seguintes métodos disponíveis:

- AutenticaAcesso: Autentica o usuário no ambiente TOTVS RM. O usuário e senha devem ser informados via SOAP, criando um Token para isto;
- AutenticaAcessoAuth: Autentica o usuário no ambiente TOTVS RM;
- GestSchema ou GestSchemaAuth : Faz a leitura do DataServer (regra de negócio RM) informado;
- ReadRecord ou ReadRecordAuth : Faz a leitura do DataServer (regra de negócio RM) e retorna o registro específico (XML), conforme chave primária;
- ReadView ou ReadViewAuth : Faz a leitura do DataServer (regra de negócio RM) e retorna conjunto de registros (XML), conforme definido no filtro;
- SaveRecord ou SaveRecordAuth: Insere um registro no DataServer (regra de negócio RM). Este método somente irá incluir o registro se este for válido de acordo com as regras de negócio definidas pelo DataServer.

*OBSERVAÇÃO: Os WebMethods que possuem no final do nome a expressão “Auth” não utilizam Token e podem ser consumidos através do Browser.*

A lista DataServers disponíveis para os serviços citados se encontra em “<http://tdn.TOTVS.com/pages/releaseview.action?pagId=143655277>” mas cabe lembrar que tais DataServer estão em constante evolução de acordo com as versões do TOTVS RM.