

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.

Gerador de Saídas

RM – Versão 12

Habilitar o participante na criação, configuração e alteração dos layouts. O Gerador de Saídas é uma ferramenta de integração criada para exportar dados de qualquer módulo do RM através de arquivo texto. Seu objetivo é a criação de layouts dos mais diversos tipos, desde uma simples listagem de tabela a Remessa Bancária, EFD-PIS/COFINS.



Sumário

1. Objetivo.....	4
2. Introdução.....	5
3. Principais Conceitos	5
3.1. O que é layout	5
3.2. O que é arquivo	6
3.3. O que é registro	7
3.4. O que é campo	8
3.5. O que é contexto	8
4. Importante.....	9
5. Configuração em ambiente 3 camadas	9
6. Acessando o RM Gerador de Saídas	10
7. Categoria de Layout	12
8. Tipo de Layout.....	15
9. Comparação	17
10. Executar.....	19
11. Layouts	21
11.1. Criando Layout	21
11.2. Construindo e Configurando Layout	23
11.2.1. Formatação Padrão.....	24
11.2.1.1. Texto.....	24
11.2.1.2. Número Inteiro	24
11.2.1.3. Número Decimal	25
11.2.1.3. Data	26
11.2.1.4. Lógico	26
11.3. Adicionando Arquivo ao Layout	27
11.4. Adicionando Registros ao Layout	28
11.5. Adicionando Contexto ao Layout.....	31
11.6. Tipos de Contexto:.....	32
11.6.1. Classe RM.NET:.....	32
11.6.1.1. Consulta SQL:	33
11.6.1.2. Cadastro do Sistema (Tela do Sistema):	35
11.6.1.3. Contexto da Aplicação:	36
11.6.1.4. Conjunto de Dados:	39
11.6.1.5. Contadores:	42
11.6.1.6. Stored Procedure:.....	44
11.6.1.7. Criação de Layout do Gerador de Saídas com Parâmetros	46
11.6.1.8. Operações com Layout.....	50

11.7. Gerar Documentação do Layout.....	51
11.8. Gerar Classe.....	52
12. Informações Complementares:.....	53
12.1. Remoção de Caracteres em Campos do Gerador de Saídas.....	53
12. Exercício.....	55
12.2. Layout de Saldo de Férias Funcionários.....	55
13. Conclusão.....	59
14. Exemplo de Layout.....	59
14.1. Exemplo de Layout ECF.....	59
14.1.1. Layout ECF.....	59

1. Objetivo

O Curso RM Gerador de Saídas visa apresentar as funcionalidades dessa ferramenta, bem como orientar no desenvolvimento de layouts em .txt.

Conteúdos abordados neste Curso:

- Acessando o Gerador de Saídas.

Conceitos Básicos:

- Layout.
- Arquivo.
- Registro.
- Campo.
- Contexto
- Inclusão de um layout
- Configuração do layout
- Criação dos campos do registro

Criação dos contextos do layout:

- Classe RM.NET.
- Consulta SQL.
- Cadastro do Sistema.
- Contexto da Aplicação.
- Parâmetros da Geração.
- Conjunto de Dados.
- Contadores.
- Stored Procedure.
- Associando o contexto aos registros.
- Associando informações do contexto aos campos do registro.

- Filtrando os registros da exportação.
- Gerando o arquivo do layout.

2. Introdução

O que é Gerador de Saídas?

O Gerador de Saídas é uma ferramenta criada para exportar de qualquer módulo do RM através de arquivo texto. Inicialmente a ferramenta foi desenvolvida para ser utilizada pelo Gestão Fiscal onde as rotinas são todas geradas no formato texto.

Disponibilidade

O Gerador de Saídas só está disponível a partir da versão 11.40 podendo ser acessado pela nova MDI. Em versões anteriores a 11.40 é possível executar somente o serviço, não sendo possível a criação e edição dos layouts. O Gerador de Saídas, hoje, se encontra disponível somente no módulo Fiscal e de integração. Estuda-se a possibilidade desta exportação ser feita em outros formatos, como o XML por exemplo.

3. Principais Conceitos

Este material didático foi desenvolvido seguindo uma ordem que facilita o aprendizado. Mostraremos a sequência em que desenvolvemos um layout, iniciando da maneira mais simples e ao decorrer das lições incrementando recursos mais complexos.

Antes de iniciar

Antes de criar um layout temos que nos atentar para um detalhe muito importante: Nunca inicie o desenvolvimento de um layout se você não estiver seguro de que sabe exatamente o que o layout deve conter e de que forma este deve mostrar as informações.

Esclareça todas as dúvidas antes de iniciar o desenvolvimento do layout, desta forma podemos evitar retrabalhos. Após ter conhecimento de todos os detalhes do layout, precisamos formatar a sua estrutura e identificar quais contextos fará parte de seu conteúdo.

3.1. O que é layout

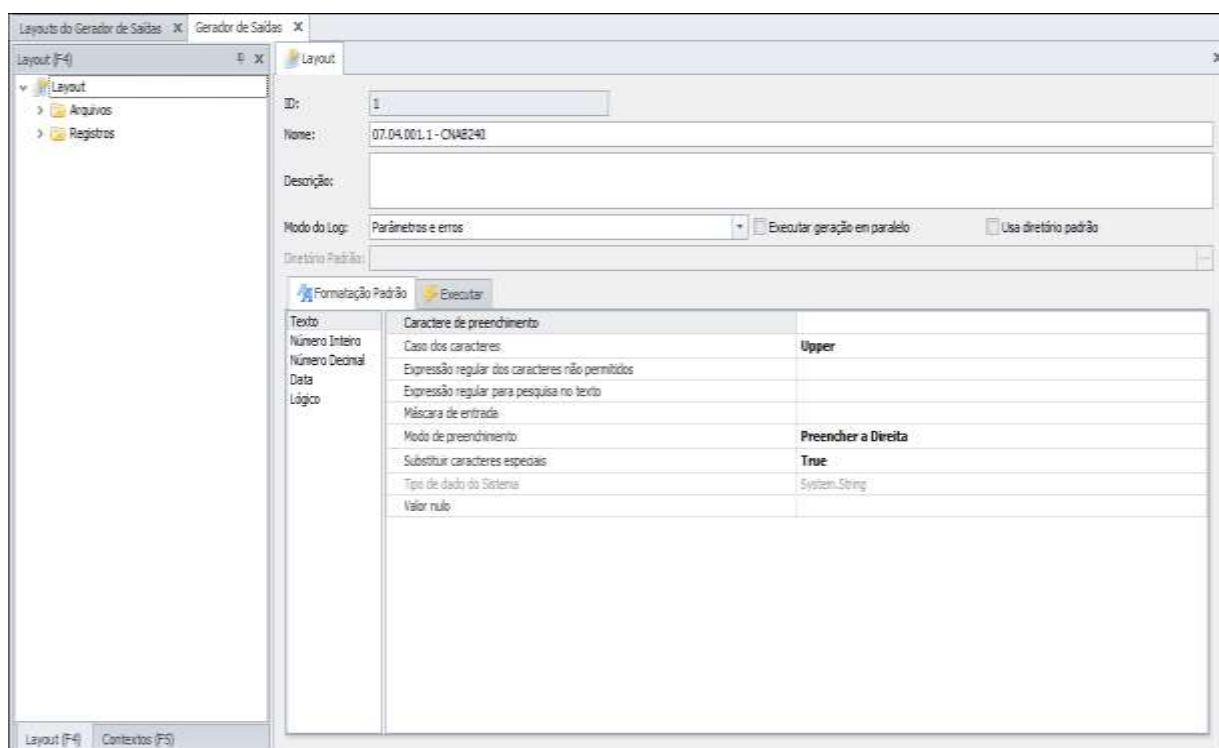
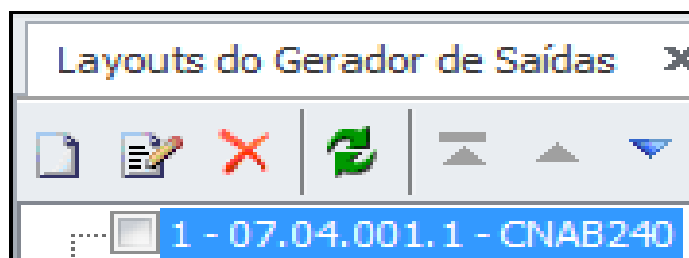
Quando se fala em layout refere-se a organização.

Definir o layout de um escritório, por exemplo é organizar as mesas e armários da melhor maneira possível para que se possa aproveitar o espaço da sala.

No caso de rotinas legais e da integração, layout é um arquivo que define a organização das informações solicitadas: ordenação, hierarquia, formato de dados e formato do arquivo. Sempre que trabalha com exportação deve-se definir um layout para auxiliar na leitura das mesmas por uma pessoa ou outro sistema.

Em síntese, o layout contém os registros com as informações que serão geradas no arquivo. Determina a organização destas informações no arquivo gerado.

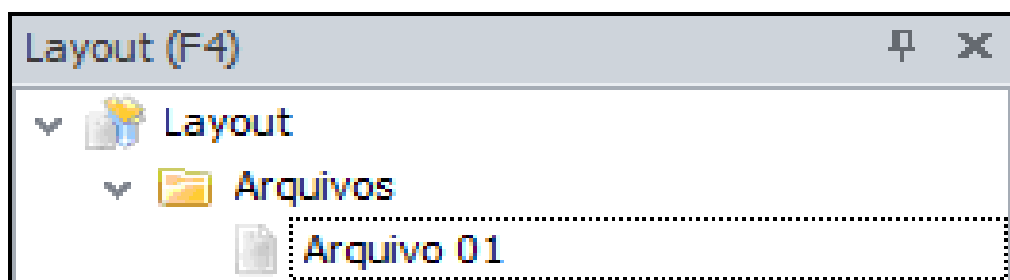
Exemplo de layout

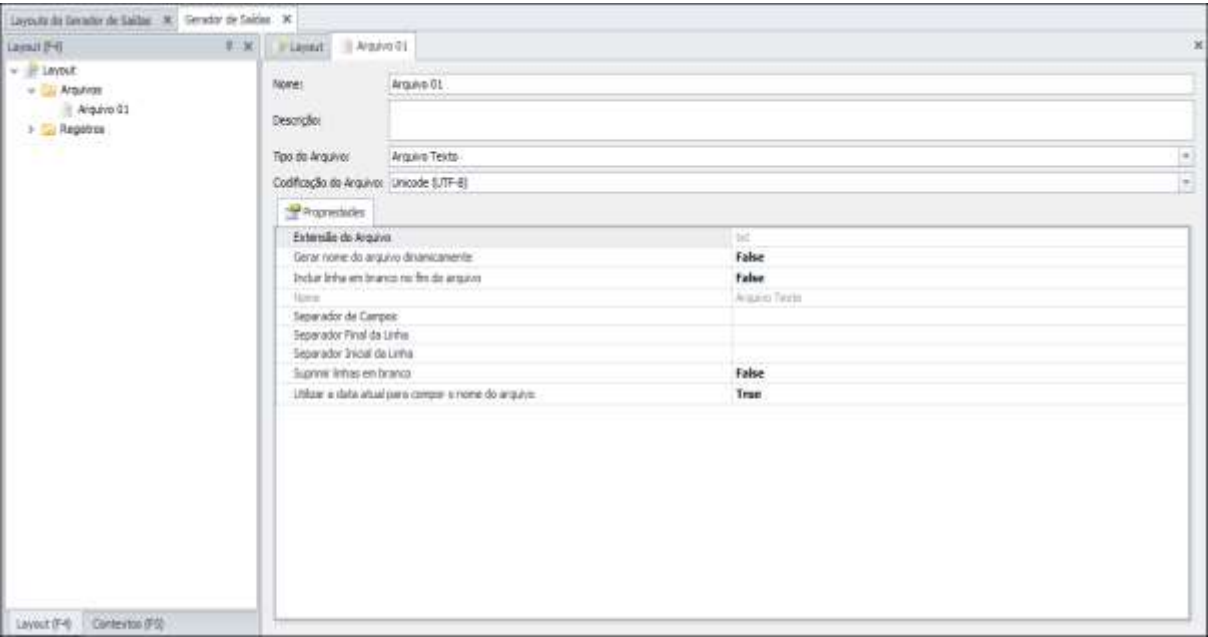


3.2. O que é arquivo

É o arquivo a ser gerado pelo Gerador de Saídas.

Exemplo de arquivo

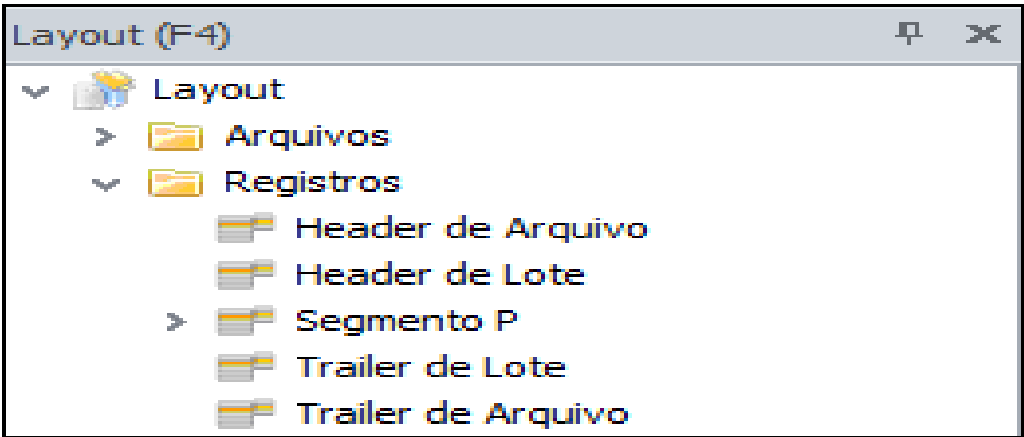


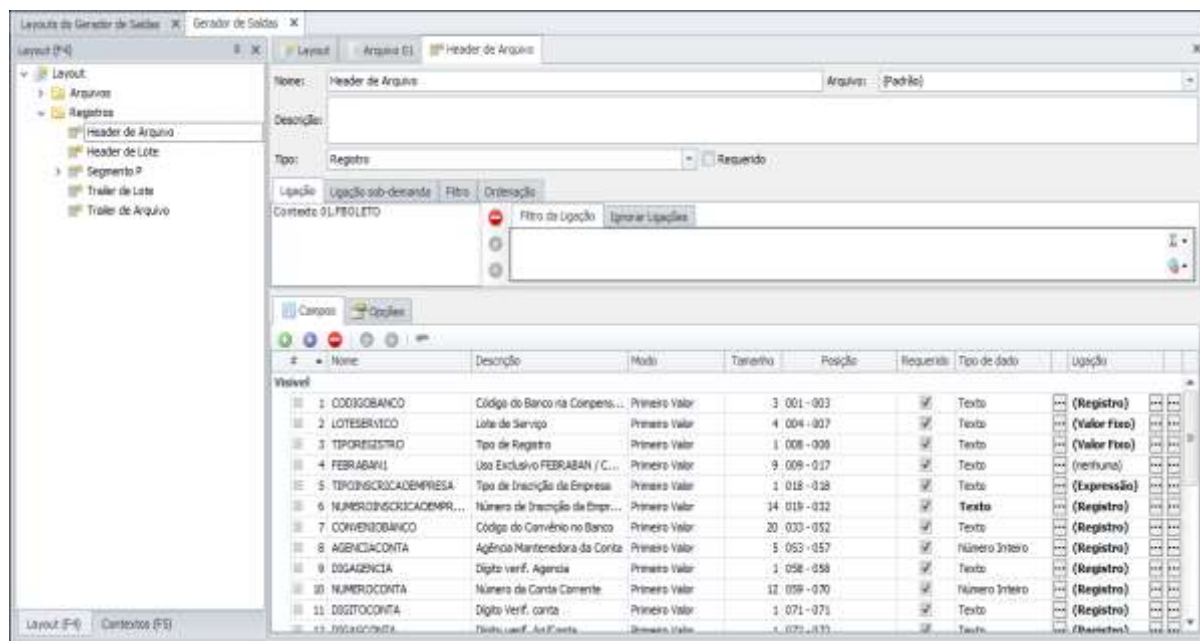


3.3. O que é registro

É a informação a ser gerada no arquivo.

Exemplo Registro





3.4. O que é campo

É a parte da informação (registro) a ser gerada no arquivo.

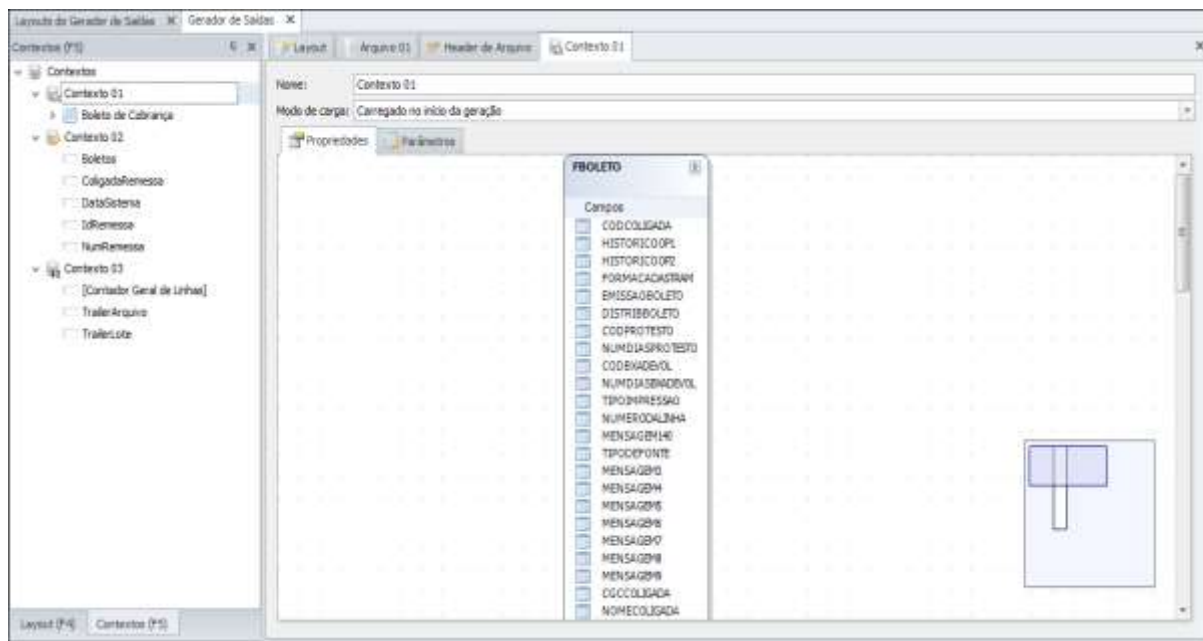
Exemplo de campo

#	Nome	Descrição	Modo	Tamanho	Posição	Requerido	Tipo de dado	Ligação
1	CODIGOBANCO	Código do Banco na Compens...	Primeiro Valor	3	001 - 003	<input checked="" type="checkbox"/>	Texto	... (Registro) ...
2	LOTESERVICO	Lote de Serviço	Primeiro Valor	4	004 - 007	<input checked="" type="checkbox"/>	Texto	... (Valor Fixo) ...
3	TIPOREGISTRO	Tipo de Registro	Primeiro Valor	1	008 - 008	<input checked="" type="checkbox"/>	Texto	... (Valor Fixo) ...
4	FEBRABAN1	Uso Exclusivo FEBRABAN / C...	Primeiro Valor	9	009 - 017	<input checked="" type="checkbox"/>	Texto	... (nenhuma) ...
5	TIPOINSCRICAOEMPRESA	Tipo de Inscrição da Empresa	Primeiro Valor	1	018 - 018	<input checked="" type="checkbox"/>	Texto	... (Expressão) ...
6	NUMEROINSCRICAOEMPR...	Número de Inscrição da Empr...	Primeiro Valor	14	019 - 032	<input checked="" type="checkbox"/>	Texto	... (Registro) ...
7	CONVENIOBANCO	Código do Convênio no Banco	Primeiro Valor	20	033 - 052	<input checked="" type="checkbox"/>	Texto	... (Registro) ...
8	AGENCIACONTA	Agência Mantenedora da Conta	Primeiro Valor	5	053 - 057	<input checked="" type="checkbox"/>	Número Inteiro	... (Registro) ...
9	DIGAGENCIA	Dígito verif. Agência	Primeiro Valor	1	058 - 058	<input checked="" type="checkbox"/>	Texto	... (Registro) ...
10	NUMEROCONTA	Número da Conta Corrente	Primeiro Valor	12	059 - 070	<input checked="" type="checkbox"/>	Número Inteiro	... (Registro) ...
11	DIGITOCONTA	Dígito Verif. conta	Primeiro Valor	1	071 - 071	<input checked="" type="checkbox"/>	Texto	... (Registro) ...

3.5. O que é contexto

É a origem dos dados, ou seja, origem das informações que serão geradas no arquivo.

Exemplo de contexto



4. Importante

Evite utilizar caractere especial na criação do layout (arquivos, registros, campos e contextos). Na geração irá acontecer a compilação do layout e esses caracteres especiais podem gerar dificuldades na compilação.

5. Configuração em ambiente 3 camadas

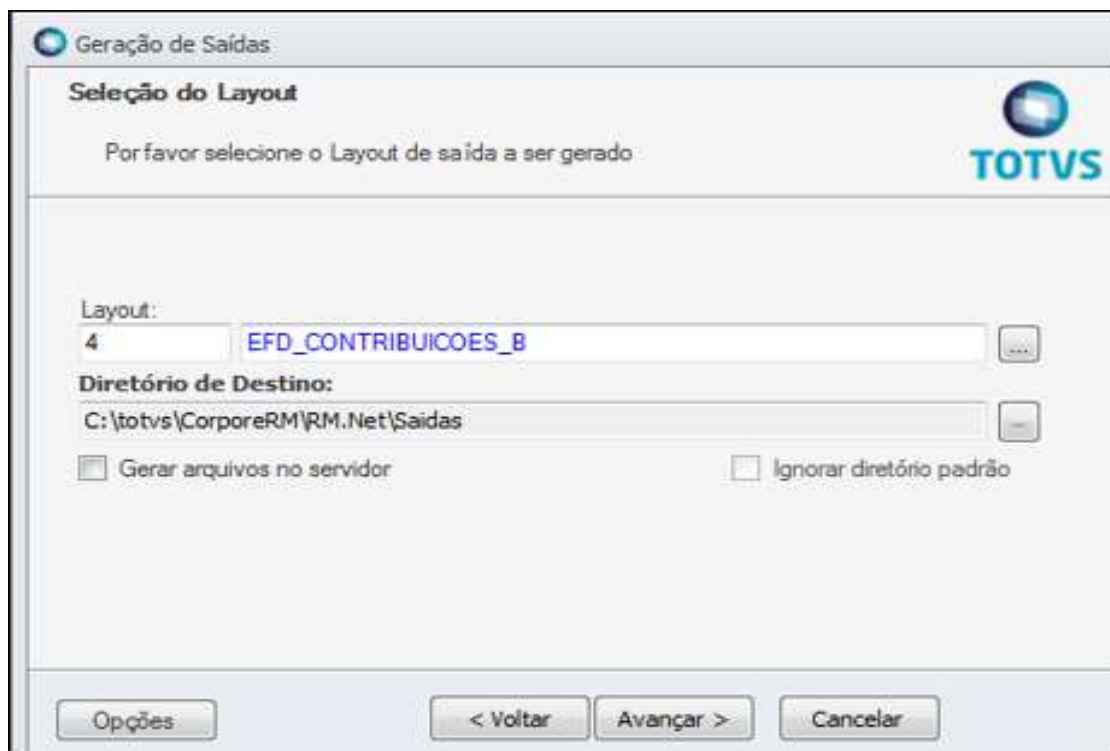
Para configurar o Gerador de Saídas em ambiente 3 camadas:

Primeiro passo é necessário inserir a Tag abaixo nos arquivos RM.Host.exe.Config e RM.Host.Service.Config.

```
<?xml version="1.0"?>
<configuration>
  <appSettings>
    <add key="FileServerPatch" value="//MAQUINA NA REDE\PASTA COMPARTILHADA" />
  </appSettings>
</configuration>
```

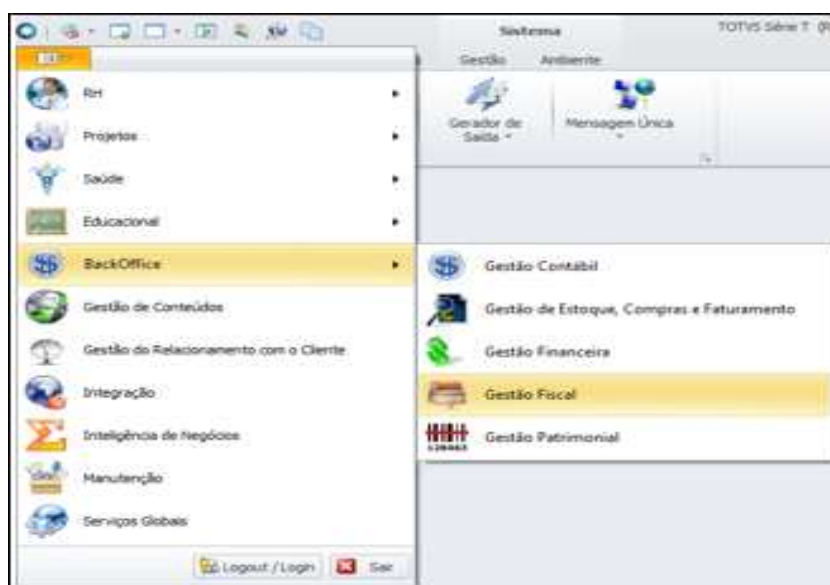
Segundo passo é criar um diretório em um dos servidores de rede e compartilhar estes diretórios, liberando acesso a estes diretórios ao usuário que executa o Host do sistema.

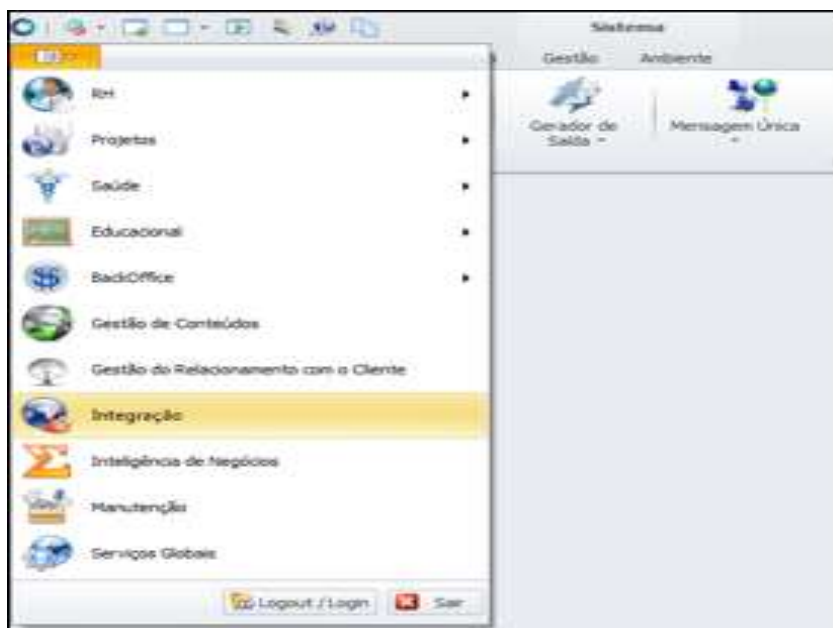
Terceiro passo é marcar a opção: "Gerar Arquivo em Servidor" na geração do layout. Esta opção será relembrada em tópico posterior.



6. Acessando o RM Gerador de Saídas

Acesse a MDI RM em sua área de trabalho. O RM Gerador de Saídas está disponível nos aplicativos no menu **TOTVS Linha RM / BackOffice / Gestão Fiscal e Integração**:

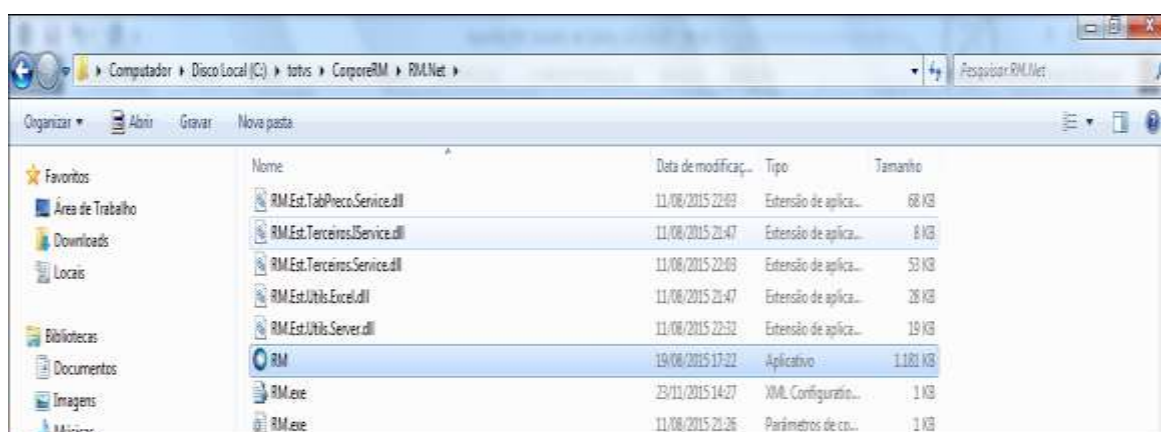




Através deste acesso, podemos visualizar os layouts de todos os módulos do RM.

Ou acesse o diretório de instalação do sistema:

C:\totvs\CorporeRM\RM.Net e executar o arquivo RM.exe.





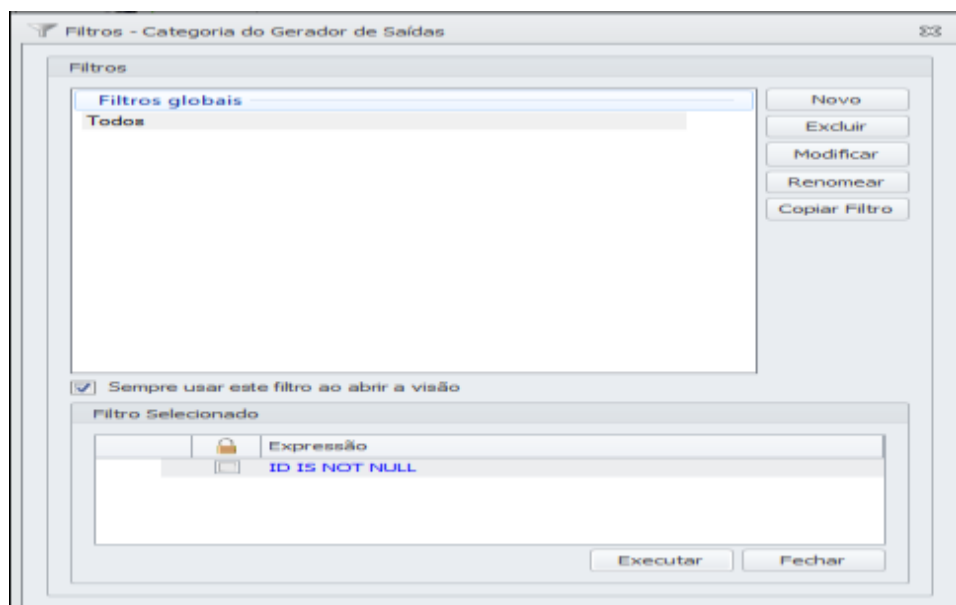
E ao acessar a MDI RM, localize no menu **TOTVS Linha RM / BackOffice / Gestão Fiscal ou Integração**.

7. Categoria de Layout

A categoria de layout serve para permitir acesso ao usuário. **Acesse o menu Integração / Gerador de Saída e clique em Categoria:**

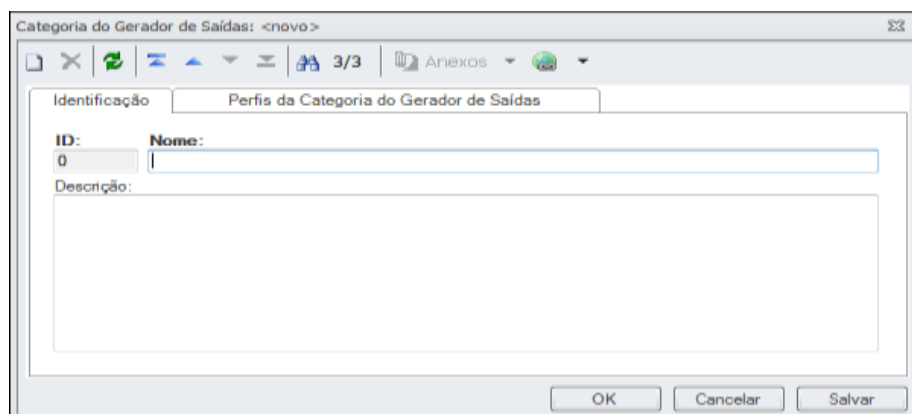


Crie um filtro ou execute o filtro existente:



Para criar nova categoria, clique no ícone "Incluir (Ctrl+Ins)", localizado na barra superior e a esquerda.

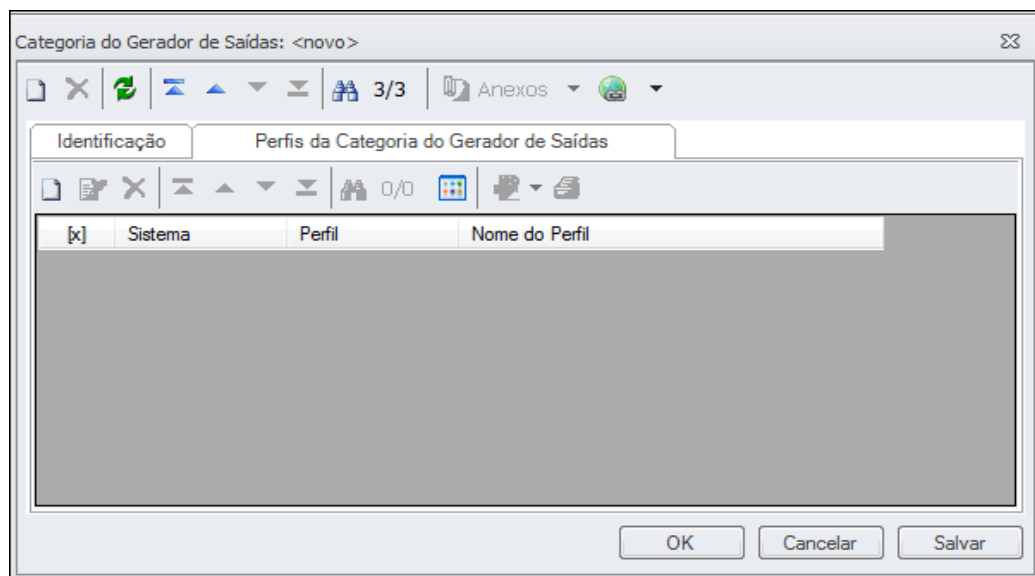
Preencha com as informações:



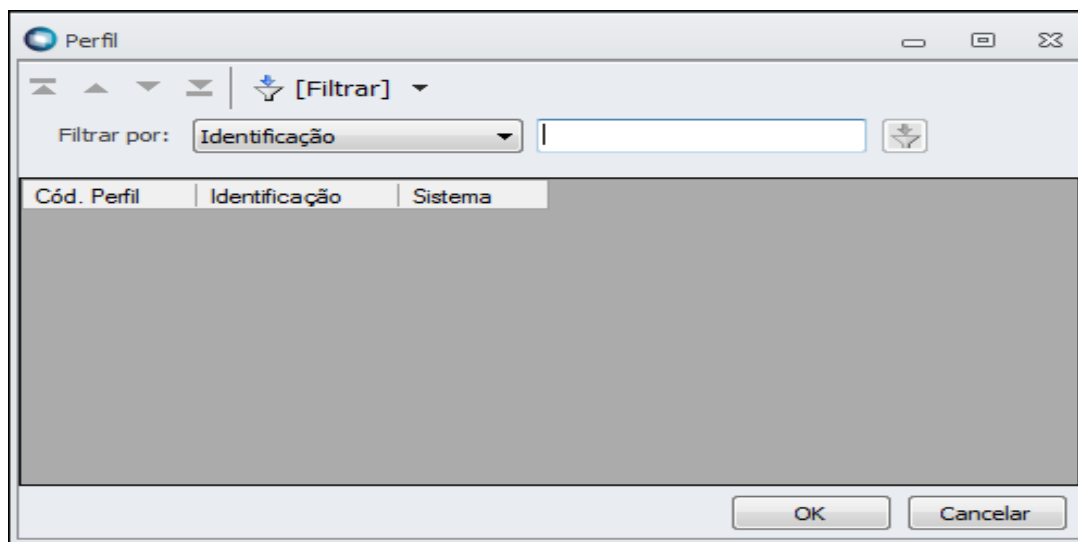
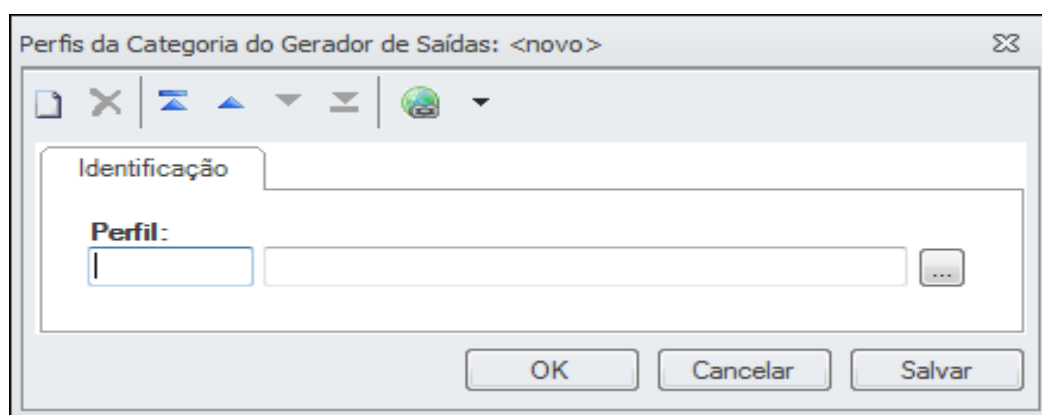
ID: Número identificador gerado pelo aplicativo de forma sequencial e automática.

Nome: Digite o nome da categoria.

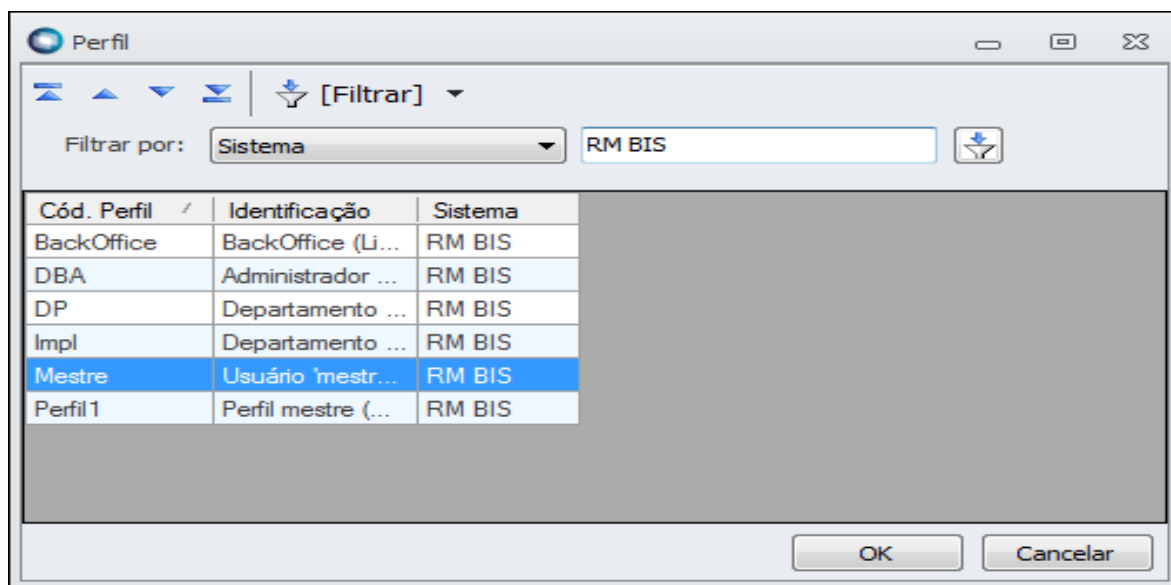
Descrição: Digite a descrição completa da categoria.



Para vincular o perfil, clique no ícone “Incluir (Ctrl+Ins)”, localizado na barra superior e a esquerda.



Filtre por sistema e vincule o perfil a categoria nova.

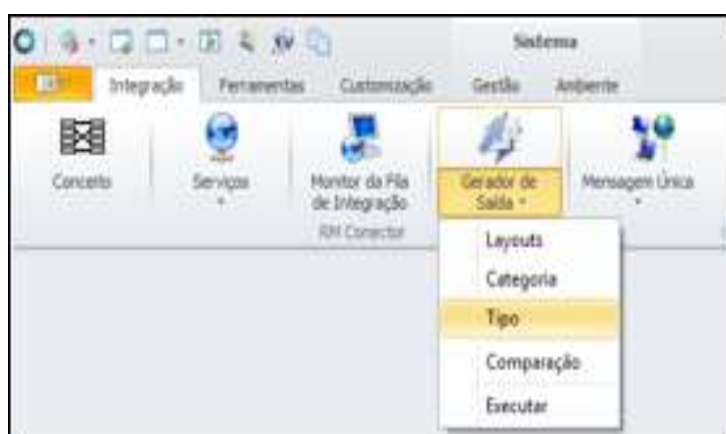


8. Tipo de Layout

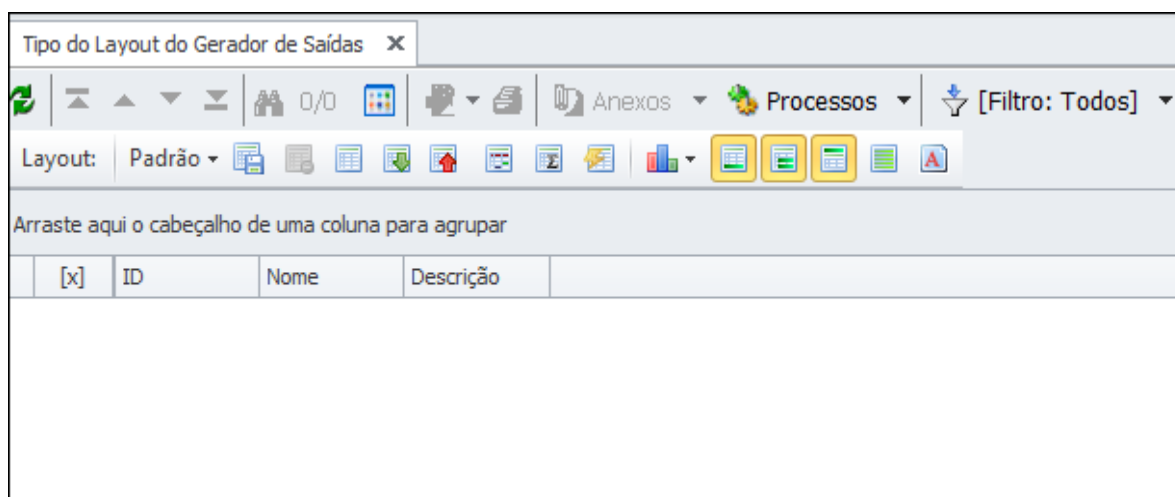
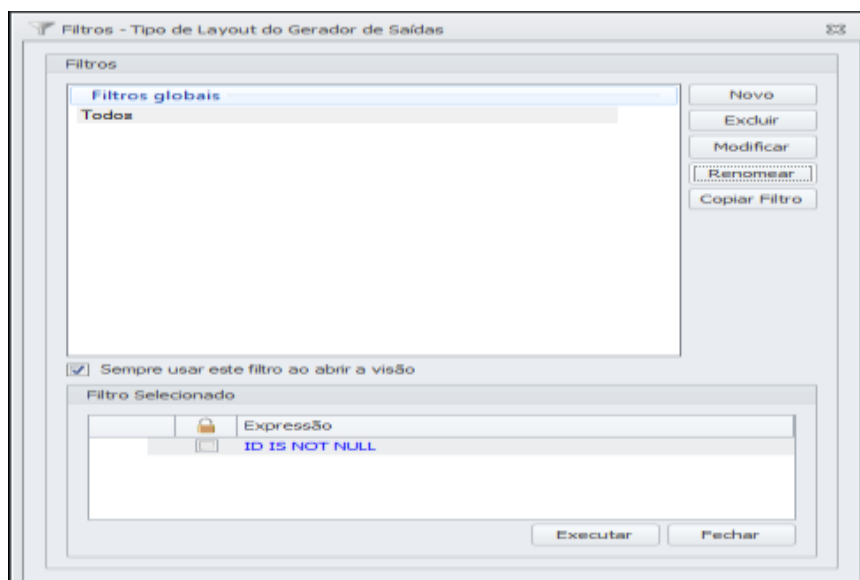
O tipo de layout não é cadastrado pelo usuário. São tipos de pré-definidos que são incluídos pelo Analista responsável de cada Produto via banco de dados no script de conversão da base. Os tipos são utilizados para separar os layouts por processos/aplicativos.

Exemplo

Quando for criada uma chamada para layouts do financeiro não serão apresentados outros tipos de layout.



Crie um filtro ou execute o filtro existente:

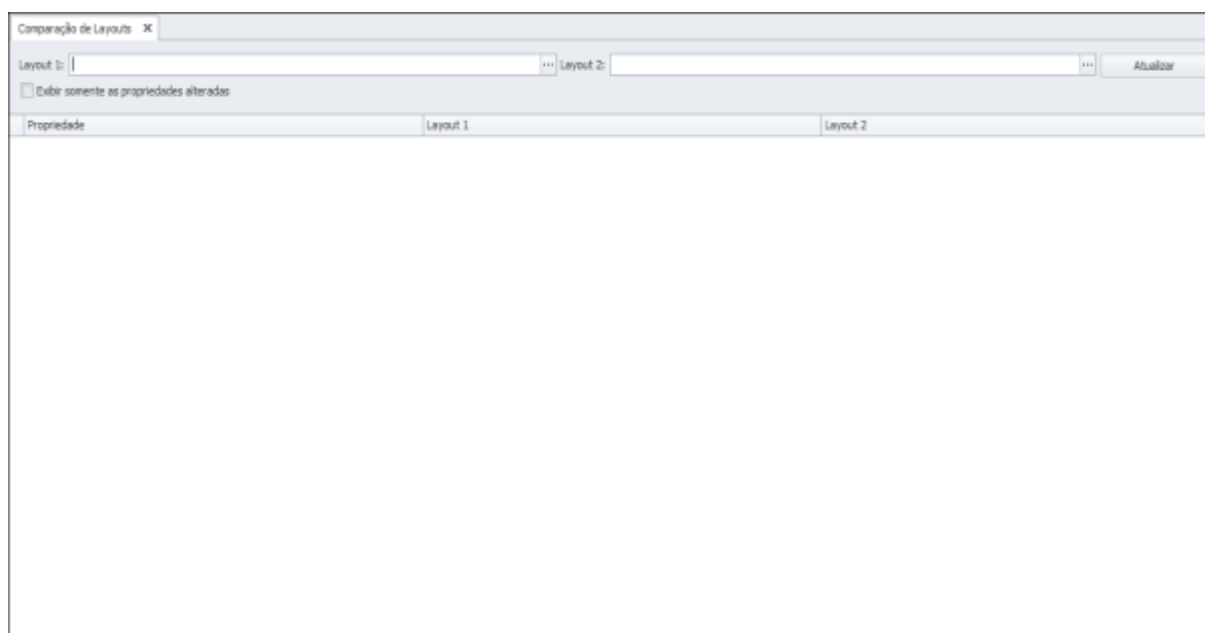


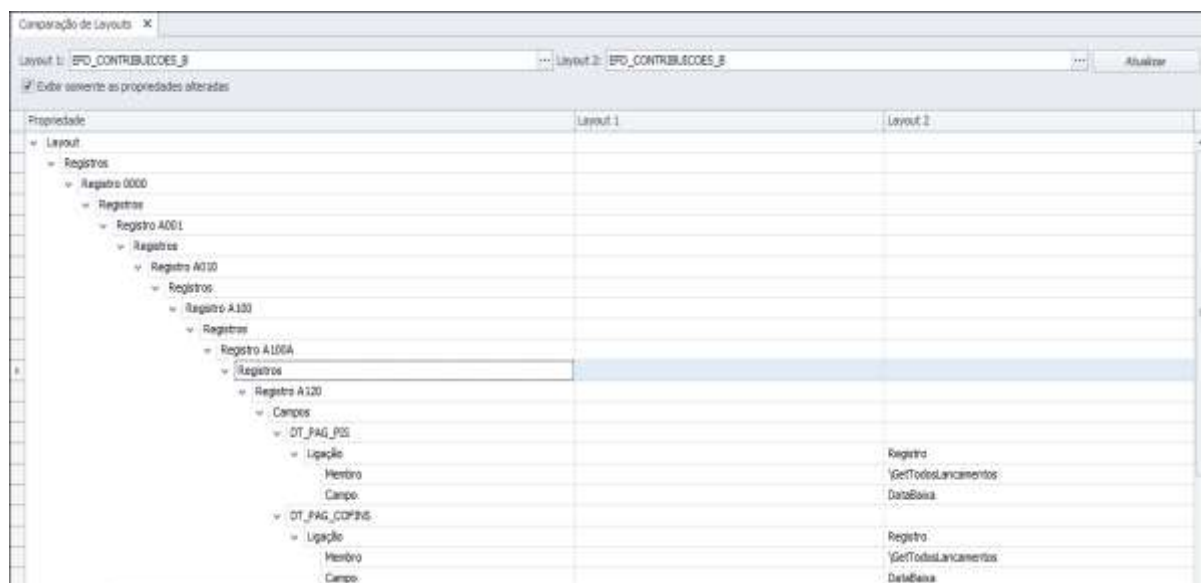
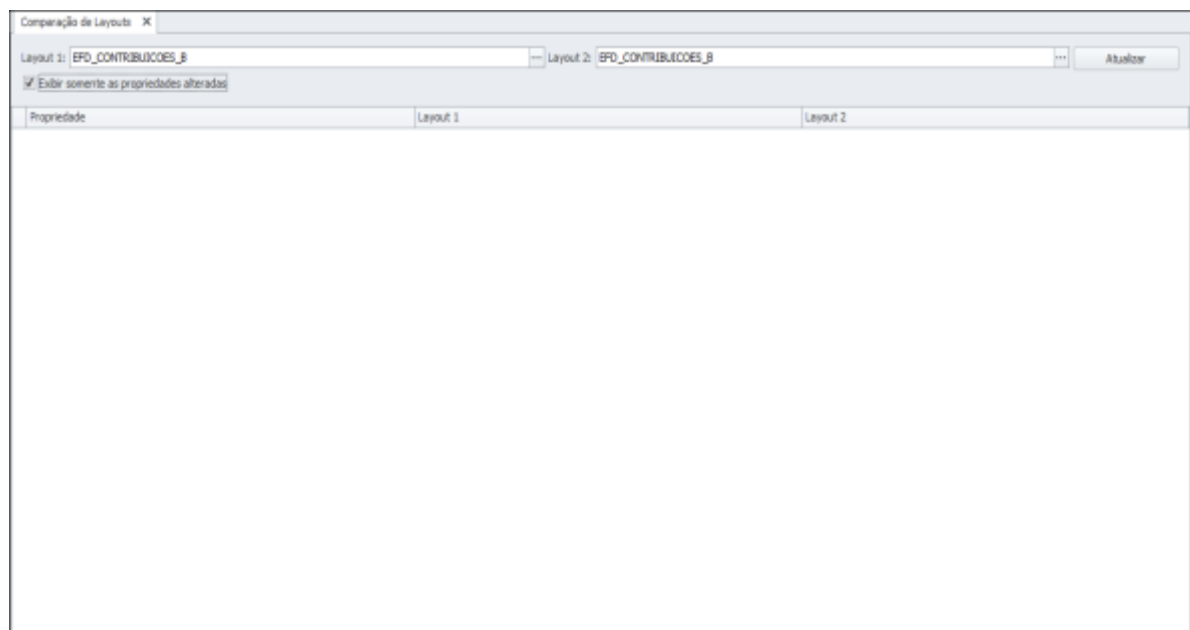
9. Comparação

Para verificar as alterações nos layouts.

Exemplo

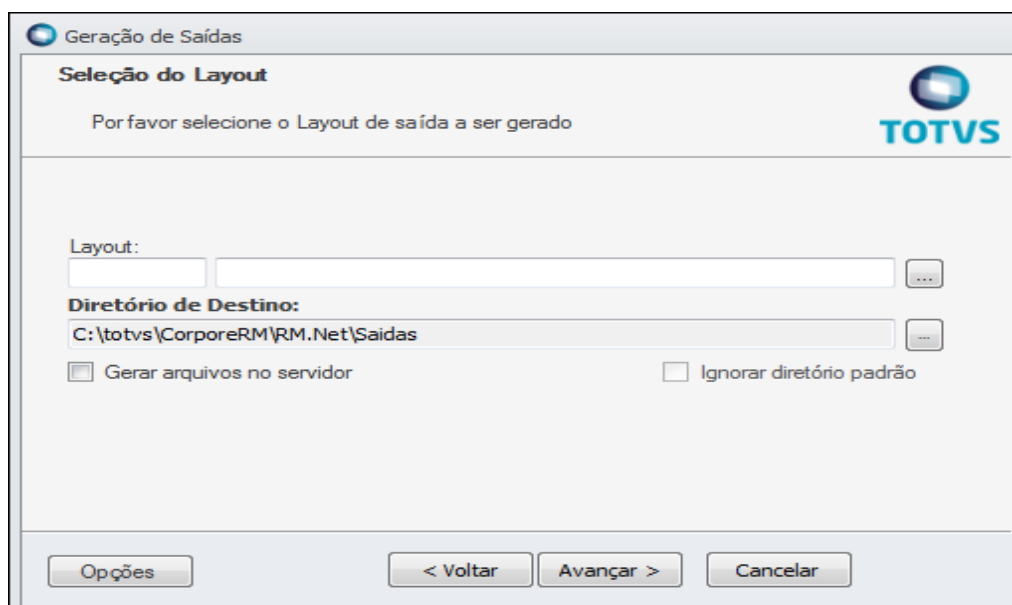
Layout EFD-Contribuições, ECF.





10. Executar

Para gerar o arquivo de exportação.



Geração de Saídas

Parâmetros da Geração

Por favor informe os parâmetros para a geração do Layout

Data Layout
21/05/2015

Filial Matriz
1 [Matriz](#)

Parâmetros ECF
1 [01/01/2014 - 31/12/2014](#)

Versão do layout
0001

☐ Retificadora

Número do Recibo Anterior

[Opções](#) [< Voltar](#) [Avançar >](#) [Cancelar](#)

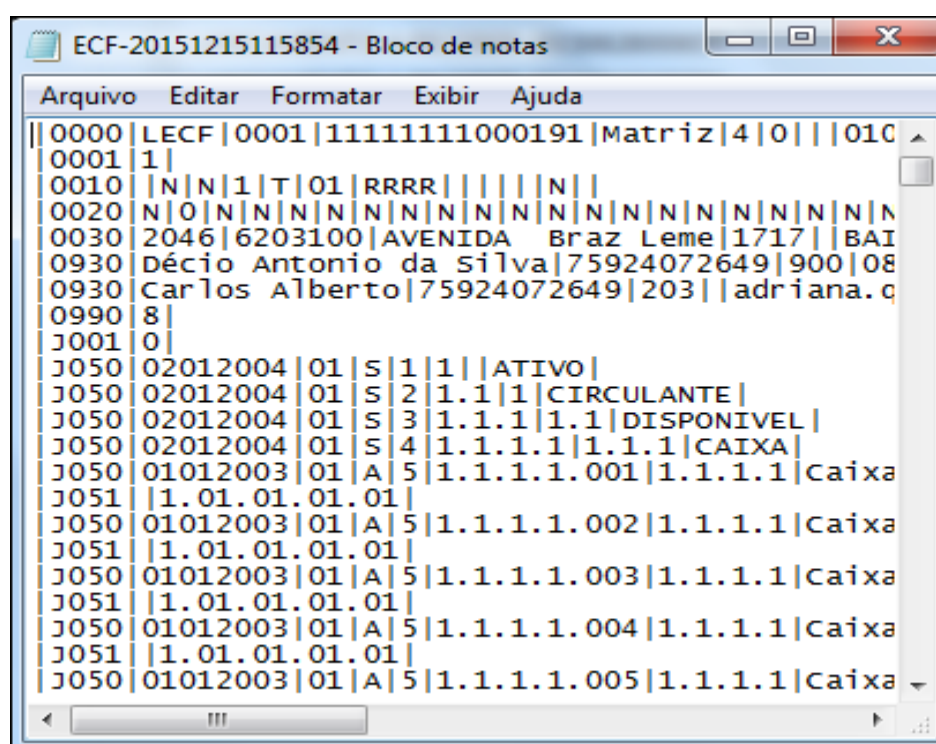
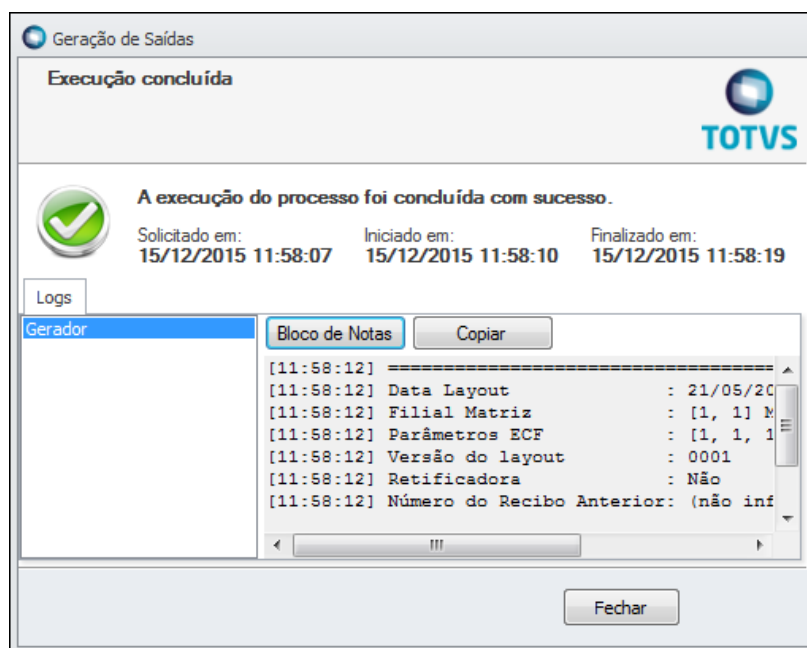
Geração de Saídas

Concluindo o Assistente ancestral de processos

Finalizado o processo

Para iniciar a execução e aceitar as configurações escolhidas, clique em 'Executar'.

[Opções](#) [< Voltar](#) [Executar](#) [Cancelar](#)

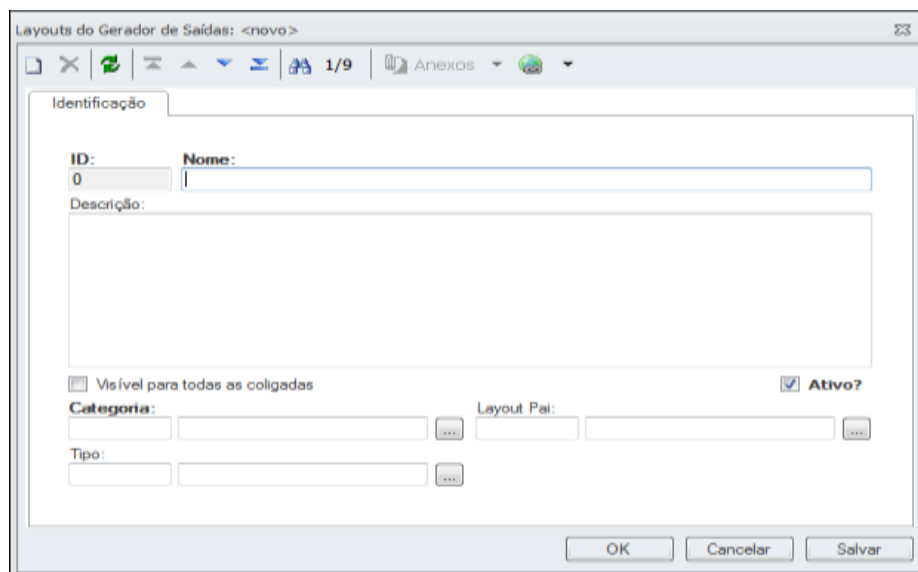


11. Layouts

11.1. Criando Layout

Para criar novo layout, clique no ícone "Incluir (Ctrl+Ins)", localizado na barra superior e a esquerda.

Preencha com as informações:



Layouts do Gerador de Saídas: <novo>

Identificação

ID: 0 Nome:

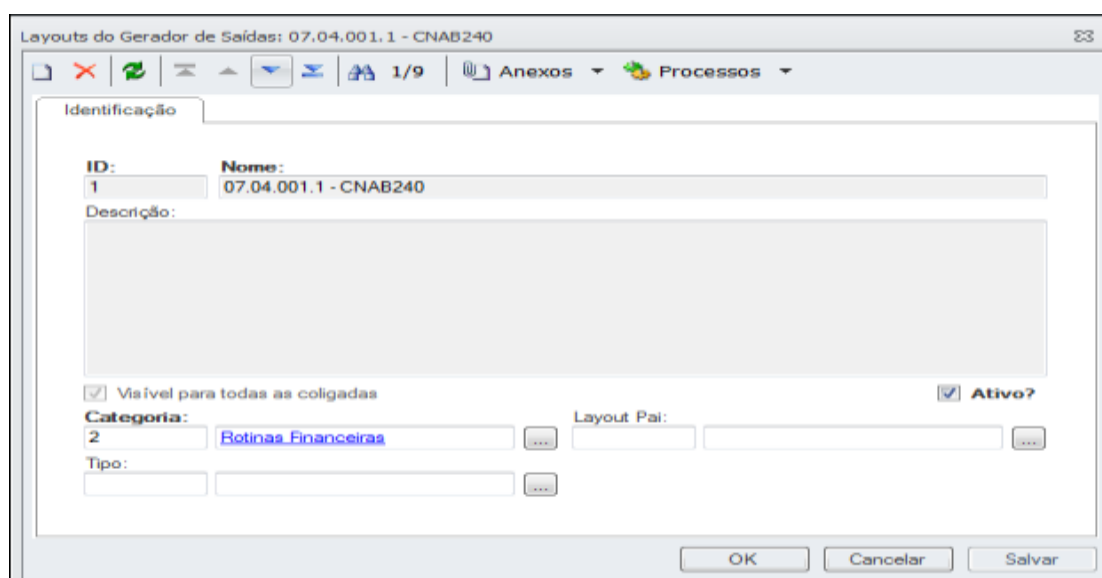
Descrição:

☐ Visível para todas as coligadas ☒ Ativo?

Categoria: Layout Pai:

Tipo:

OK Cancelar Salvar



Layouts do Gerador de Saídas: 07.04.001.1 - CNAB240

Identificação

ID: 1 Nome: 07.04.001.1 - CNAB240

Descrição:

☒ Visível para todas as coligadas ☒ Ativo?

Categoria: 2 Rotinas Financeiras Layout Pai:

Tipo:

OK Cancelar Salvar

ID: Número identificador gerado pelo aplicativo de forma sequencial e automática.

Nome: Digite o nome do seu arquivo de exportação.

Descrição: Digite a descrição completa do seu arquivo de exportação.

Visível para todas as coligadas: Flag (marque) este campo se o seu arquivo de exportação será visível em outras coligadas.

Ativo: Flag (marque) este campo se o layout está disponível. Caso, queira inativar desmarque este campo.

Categoria: Vincule o perfil do usuário de acesso ao layout.

Layout Pai: Crie layout pai para separar os tipos de layouts para organização dos layouts do Gerador de Saídas. Em síntese, serve para criar uma estrutura de árvore em sua visão para melhor organizar e visualizar seus layouts.

Tipo: O tipo de layout não é cadastrado pelo usuário. São tipos de pré-definidos que são incluídos pelo Analista responsável de cada Produto via banco de dados no script de conversão da base. Os tipos são utilizados para separar os layouts por processos/aplicativos.

Exemplo

Quando for criada uma chamada para layouts do financeiro não serão apresentados outros tipos de layout.

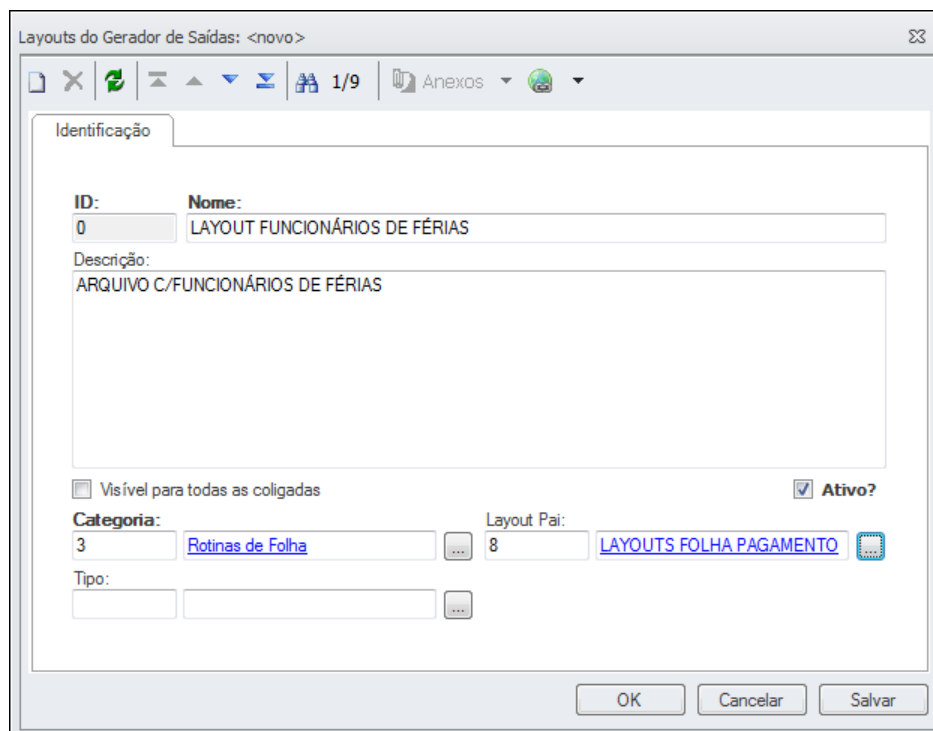
11.2. Construindo e Configurando Layout

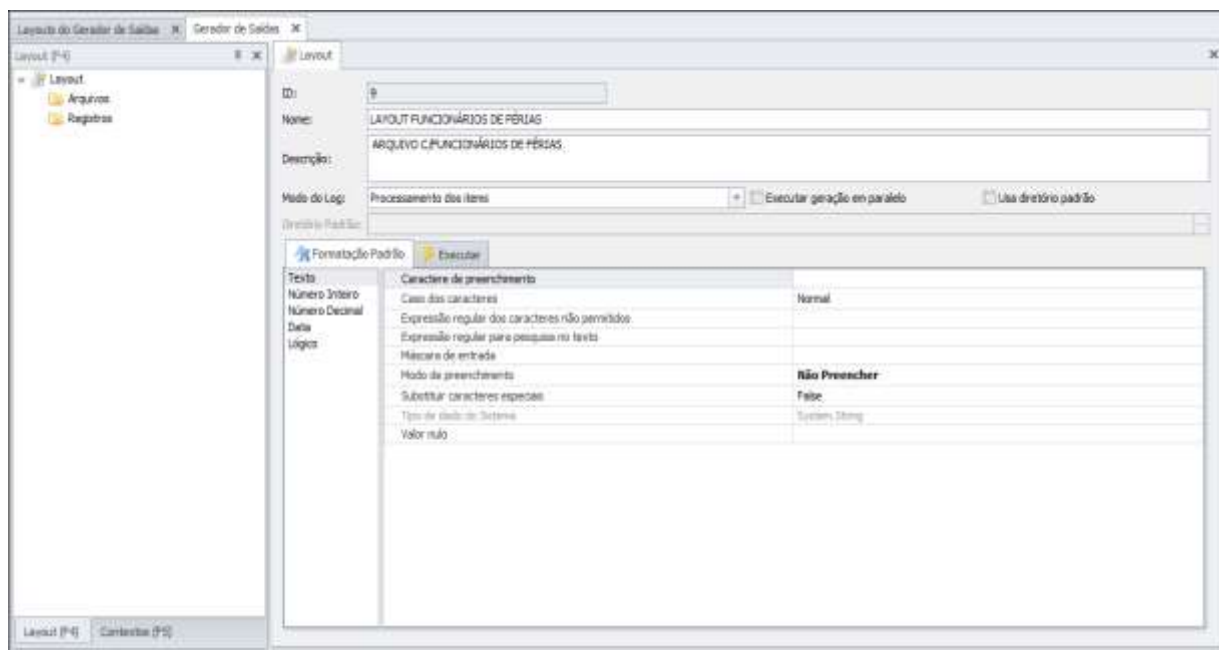
Para editar a estrutura do layout dê um duplo clique sobre o layout criado.



Preencha sempre os campos de descrição com o maior número de informações possíveis, pois desta forma a documentação do layout gerada pela ferramenta será mais completa.

As imagens abaixo apresentam as telas de cadastros, construção e configuração do layout.





Modo do Log: é possível parametrizar como será o log de geração.

Existem disponíveis os modos:

- Processamento de todos os itens.
- Processamento dos itens.
- Parâmetros e erros.
- Leitura dos itens.
- Leitura de todos os itens.
- Inicialização dos contextos.

Usa diretório padrão: definir que será usado o diretório padrão TOTVS: C:\totvs\CorporeRM\RM.Net\Sidas

11.2.1. Formatação Padrão

Em formatação padrão defina as regras para as condições de texto, número inteiro, número decimal, data e lógico.

Preenchendo essa formatação todos os campos criados no layout respeitarão essa formatação. A formatação deve ser feita separadamente pelo tipo do campo.

11.2.1.1. Texto

Formatação Padrão	Executar
Texto	Caractere de preenchimento
Número Inteiro	Caso dos caracteres
Número Decimal	Expressão regular dos caracteres não permitidos
Data	Expressão regular para pesquisa no texto
Lógico	Máscara de entrada
	Modo de preenchimento
	Substituir caracteres especiais
	Tipo de dado do Sistema
	Valor nulo

- Caracteres de p
- Exemplo: 0 (zer

- Caso dos caracteres: definir normal, upper (acima) ou lower (baixo).
- Expressão regular dos caracteres não permitidos: é possível definir uma expressão regular para remover caracteres que não podem sair no layout.
- Expressão regular para pesquisa no texto: possui a função de obter uma parte do texto desejado.
- Máscara de entrada: a máscara definida no layout é aplicada no arquivo todo e se for definida nos registros será aplicada apenas na linha definida.
- Modo de preenchimento: existem 3 modos de preenchimento: não preencher, preencher a esquerda, preencher a direita. Para todo o layout ou registro o caractere de preenchimento respeita esse modo de preenchimento.
- Substituir caracteres especiais: não substitui a função, remove os caracteres especiais. Exemplo: se vier @, /, a ferramenta irá eliminar.
- Tipo de dado do sistema: apenas para consulta. A ferramenta já apresenta o tipo de dado automaticamente. O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Exemplo: System.Int32.
- Valor nulo: definir o caractere que o Gerador de Sidas irá acrescentar quando for nula a informação.

11.2.1.2. Número Inteiro

Formatação Padrão	Executar
Texto	Caractere de preenchimento
Número Inteiro	Modo de preenchimento
Número Decimal	Separador de Milhares
Data	Tipo de dado do Sistema
Lógico	Valor nulo

- Caractere de p
- Exemplo: 0 (zer

- Modo de preenchimento: existem 3 modos de preenchimento: não preencher, preencher a esquerda, preencher a direita. Para todo o layout ou registro o caractere de preenchimento respeita esse modo de preenchimento.

- Separador de milhares: Lembrando sempre que o SQL Server internamente armazena o separador decimal como ponto (.) e o separador de milhar como vírgula (,).
- Tipo de dado do sistema: apenas para consulta. A ferramenta já apresenta o tipo de dado automaticamente. O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Exemplo: System.Int32.
- Valor nulo: definir o caractere que o Gerador de Saídas irá acrescentar quando for nula a informação.

11.2.1.3. Número Decimal

Formatação Padrão		Executar
Texto	Caractere de preenchimento	
Número Inteiro	Casas Decimais	2
Número Decimal	Modo de Arredondamento	Arredondar
Data	Modo de preenchimento	Não Preencher
Lógico	Separador de Milhares	
	Separador Decimal	,
	Tipo de dado do Sistema	System.Decimal
	Valor nulo	

- **Caractere de p**

Exemplo: 0 (ze

- **Casas decimais:** definir a quantidade de casas decimais.
- **Modo de arredondamento:** definir se irá arredondar ou truncar (A função TRUNC remove a parte fracionária do número. INT arredonda números para baixo até o inteiro mais próximo com base no valor da parte fracionária do número. Trunca um número até um inteiro, removendo a parte decimal ou fracionária do número.)
- **Modo de preenchimento:** existem 3 modos de preenchimento: não preencher, preencher a esquerda, preencher a direita. Para todo o layout ou registro o caractere de preenchimento respeita esse modo de preenchimento.
- **Separador de milhares:** Lembrando sempre que o SQL Server internamente armazena o separador decimal como ponto (.) e o separador de milhar como vírgula (,). Defina o separador de milhares.
- **Separador decimal:** Lembrando sempre que o SQL Server internamente armazena o separador decimal como ponto (.) e o separador de milhar como vírgula (,). Defina o separador decimal.
- **Tipo de dado do sistema:** Apenas para consulta. A ferramenta já apresenta o tipo de dado automaticamente. O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Exemplo: System.Int32.
- **Valor nulo:** Definir o caractere que o Gerador de Saídas irá acrescentar quando for nula a informação.

11.2.1.3. Data

Formatação Padrão		Executar
Texto	Caractere de preenchimento	
Número Inteiro	Máscara de Formatação	dd/MM/yyyy
Número Decimal	Modo de preenchimento	Não Preencher
Data	Tipo de dado do Sistema	System.DateTime
Lógico	Valor nulo	

- **Caractere de p**

Exemplo: 0 (zero), espaço, letra.

- **Máscara de formatação:** definir o tipo e formato para data.
- **Modo de preenchimento:** existem 3 modos de preenchimento: não preencher, preencher a esquerda, preencher a direita. Para todo o layout ou registro o caractere de preenchimento respeita esse modo de preenchimento.
- **Tipo de dado do sistema:** apenas para consulta. A ferramenta já apresenta o tipo de dado automaticamente. O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Exemplo: System.Int32.
- **Valor nulo:** definir o caractere que o Gerador de Saídas irá acrescentar quando for nula a informação.

11.2.1.4. Lógico

Formatação Padrão		Executar
Texto	Caractere de preenchimento	
Número Inteiro	Modo de preenchimento	Não Preencher
Número Decimal	Texto Falso	False
Data	Texto Verdadeiro	True
Lógico	Tipo de dado do Sistema	System.Boolean
	Valor nulo	

- **Caractere de p**

Exemplo: 0 (zer

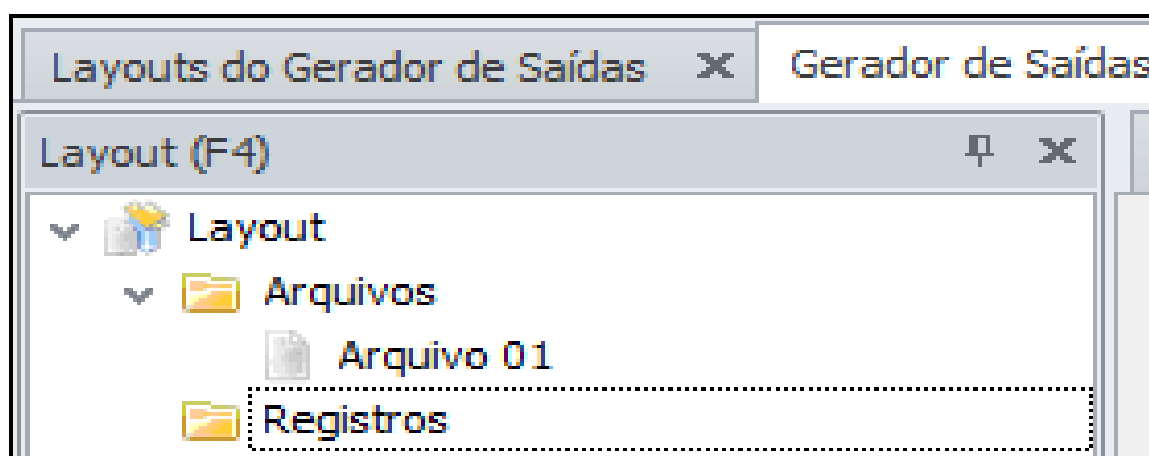
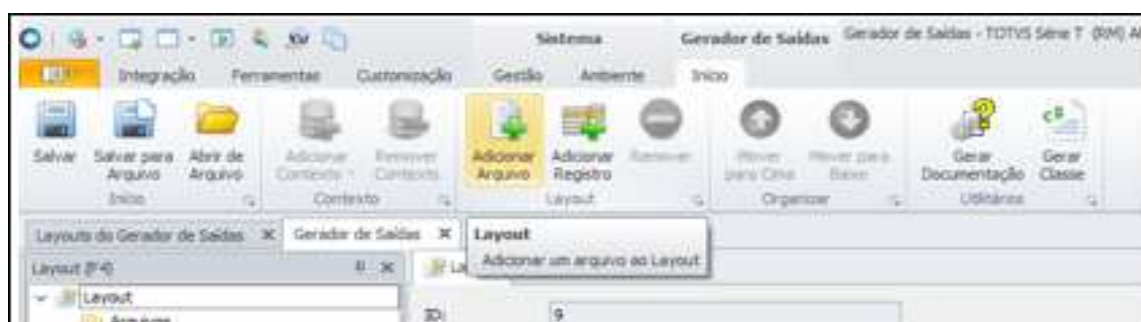
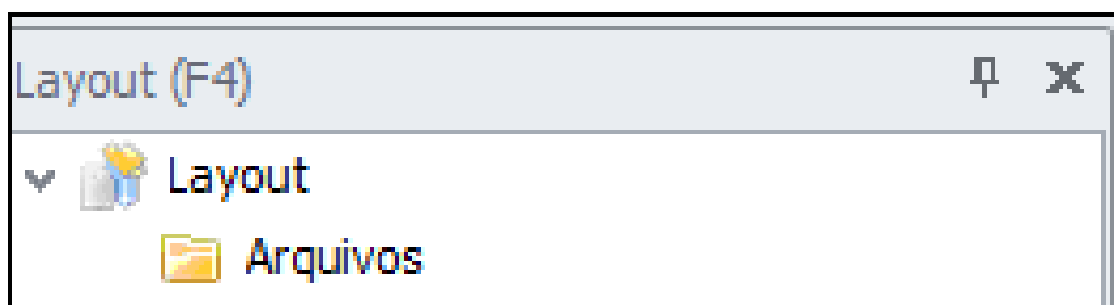
- **Modo de preenchimento:** existem 3 modos de preenchimento: não preencher, preencher a esquerda, preencher a direita. Para todo o layout ou registro o caractere de preenchimento respeita esse modo de preenchimento.
- **Texto falso:** definir a condição false.
- **Texto verdadeiro:** definir a condição true.
- **Tipo de dado do sistema:** apenas para consulta. A ferramenta já apresenta o tipo de dado automaticamente. O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL).

Exemplo: System.Int32.

- **Valor nulo:** definir o caractere que o Gerador de Saídas irá acrescentar quando for nula a informação.

11.3. Adicionando Arquivo ao Layout

Acesse a pasta “Layout (F4)” e selecione “Arquivos” a esquerda e nesta pasta, no menu Início clique no ícone “Adicionar Arquivo”:



Arquivo

- **Nome:** definir o nome do arquivo do layout.

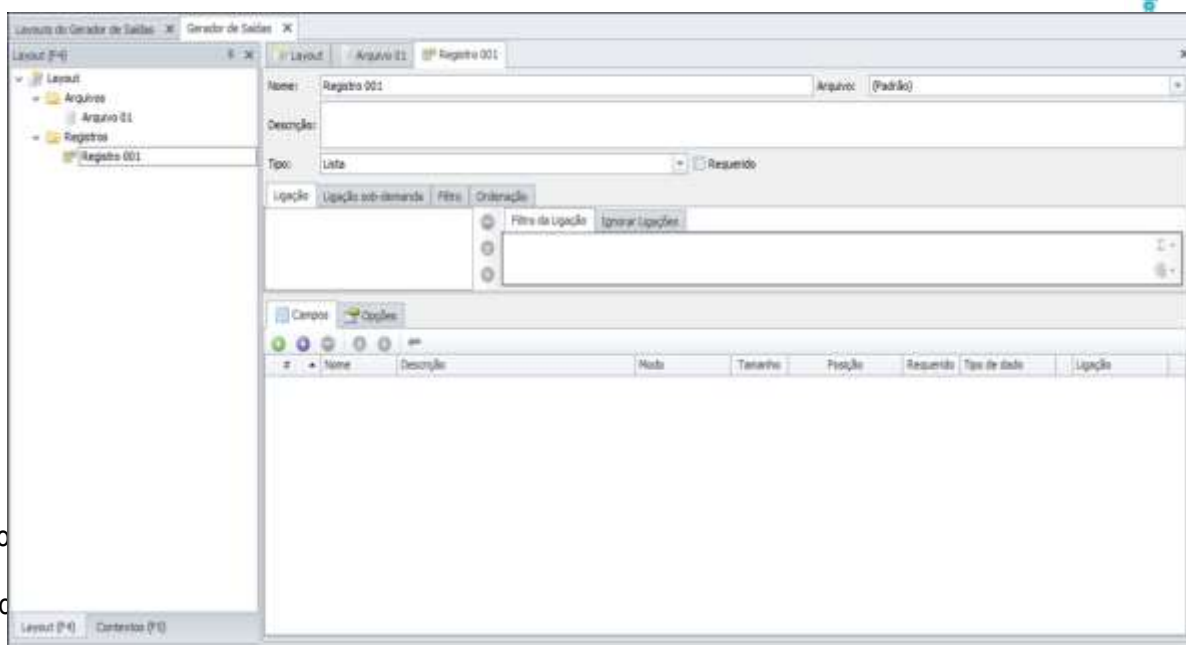
- Descrição: definir a descrição completa do arquivo do layout.
- Tipo do arquivo: definir a extensão do arquivo a ser gerado. Até o momento é gerado em .txt.
- Codificação do arquivo: ATENÇÃO: para a codificação do arquivo, dependendo da codificação escolhida os caracteres especiais podem apresentar formato inválido na geração do layout. O único tipo do arquivo disponível para geração do arquivo é o arquivo texto.

Propriedades

- Extensão do arquivo: formato padrão TOTVS em .txt.
- Nome: nome do formato padrão TOTVS em .txt.
- Separador final da linha: definir o separador para o final da linha.
- Separador inicial da linha: definir o separador para o início da linha.

11.4. Adicionando Registros ao Layout

Acesse a pasta “Layout (F4)” e selecione “Registros” a esquerda e nesta pasta, no menu Início clique no ícone “Adicionar Registro”:



Registro 001

- Nome: nome do
- Arquivo: arquivo

Descrição: descrição completa do registro.

- Tipo:

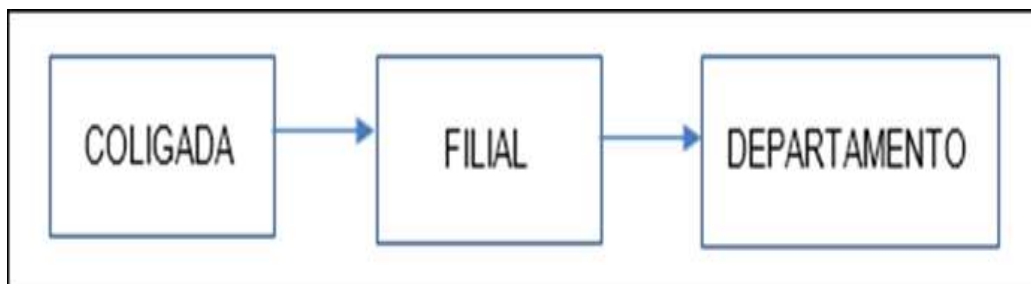
Lista: repete mais de uma vez. Imprime todos os registros, exemplo: CLI/FOR (lista).

Registro: não repete. Exemplo: cabeçalho (coligada).

Agrupador: registro oculto. Não aparece no layout. Deve ser utilizado baseado na hierarquia das informações.

Exemplo:

Para obter as informações da COLIGADA e dos DEPARTAMENTOS é necessário utilizar a tabela de FILIAL.



As informações da Filial não são necessárias no layout, então a tabela de FILIAL será um registro agrupador.

Requerido: informe se o campo é obrigatório ou não.

Ligação: possui o contexto padrão por registro.

Ligação sob demanda: é quando não possui a informação na ligação. E o usuário não consegue adicionar no conjunto de dados. Exemplo: as mensagens personalizadas no arquivo bancário.

Filtro: definir o filtro.

Ordenação: definir a ordem no layout.

Filtro da ligação: selecionar o campo do contexto que será utilizado no filtro e montar a expressão. Linguagem C#.

Exemplo

Util.ValueIn(CODCFO,"C01397"). Não é necessário replicar o filtro para os registros filhos.

Campos: incluir os campos a serem gerados no layout.

Opções:

Gravar arquivo parcial em disco: será descontinuado, pois o modo de carga do layout executa de forma mais eficiente esta função.

Gravar arquivo parcial em disco dos registros filhos:

- **Gerar somente se possuir algum registro filho:** será descontinuado, pois o modo de carga do layout executa de forma mais eficiente esta função.

iv. Id do Registro:

Legenda dos botões na aba campos:

- O botão “verde” adiciona um novo registro.
- O botão “azul” um novo registro invisível.
- O botão “vermelho” remove um registro selecionado.
- As “setas” servem para movimentar os campos no registro.
- O último botão exibe o nome da ligação dos campos para conhecimento do usuário.

Associando contexto ao registro

Para adicionar um contexto a um registro selecione o contexto desejado e arraste para o registro correspondente. Caso seja necessário é possível criar filtros e determinar uma ordenação para o registro.

Para adicionar um campo ao seu registro basta clicar no botão adicionar na guia de campos.

Para parametrizar um campo no layout preencha com os dados descritos a seguir:

- Nome: nomear o nome do campo. Não utilize espaços ou caracteres especiais.
- Descrição: descreva o campo do seu registro.
- Requerido: informe se o campo é obrigatório ou não.

Modo:

- Chave primária: utilizado quando o dado não pode ser repetido no registro.
- Primeiro valor: retornam o primeiro e último valor de uma série quando utilizados em conjunto com a chave primária.
- Último valor: retornam o primeiro e último valor de uma série quando utilizados em conjunto com a chave primária.
- Sequencial: incrementa o valor em 1 a cada registro impresso.
- Contar: conta a quantidade de dados impressos nos registros. Utilizado em conjunto com a chave primária.
- Somar: utilizado para somar valores. Utilizado em conjunto com a chave primária.
- Média: calcula a média dos valores de um registro. Utilizado em conjunto com a chave primária.
- Mínimo: retornam o menor e o maior valor de uma série quando utilizados em conjunto com a chave primária.
- Máximo: retornam o menor e o maior valor de uma série quando utilizados em conjunto com a chave primária.
- Tamanho: especifique o tamanho do campo. Pode consultar direto no banco de dados.

Posição:

Tipo de dado:

- Texto: O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Definir se é tipo texto.
- Número Inteiro: O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Definir se é tipo número inteiro.
- Número Decimal: O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Definir se é tipo número decimal.
- Data: O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Definir se é tipo data.
- Lógico: O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Definir se é tipo lógico.

Ligação:

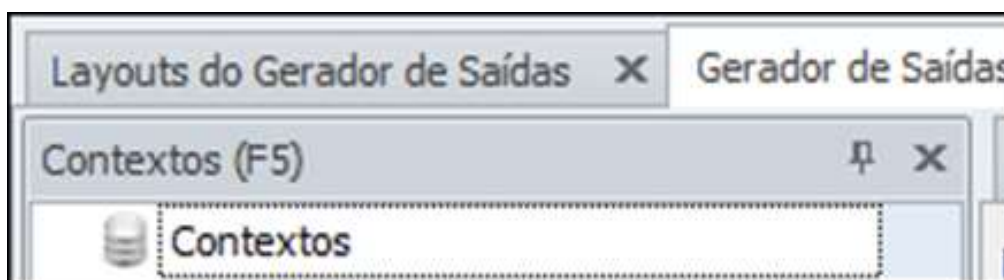
- Propriedades avançadas do registro: a edição das propriedades avançadas do campo: neste local são acessadas todas as parametrizações do campo, pode-se personalizar a formatação do campo, visualizar a ligação de dados e ainda pode-se fazer conversão de valores: “de-para”, assim como diversas outras parametrizações.

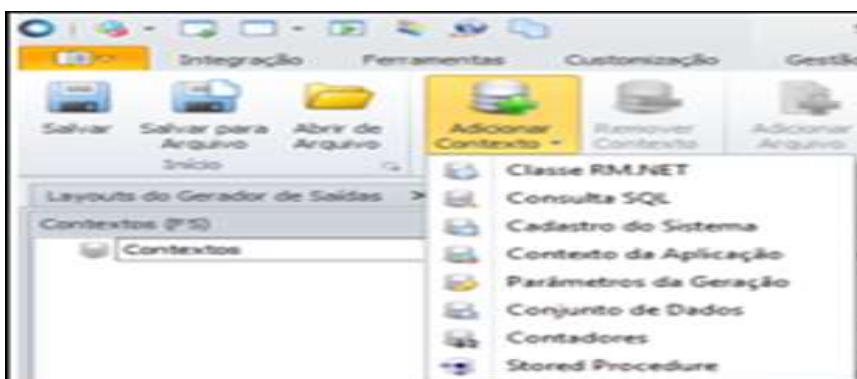
Para associar uma coluna do contexto a um campo:

- Clique sobre o nome do contexto criado, arraste e solte a coluna desejada no campo desejado.
- Observe que a coluna ligação estará com a palavra (registro) e ao passar o mouse sobre o campo é apresentada a ligação de dados.
- É possível ainda ter os seguintes tipos de campos a ligação de dados: contexto, expressão, valor fixo e fórmula.

11.5. Adicionando Contexto ao Layout

Acesse a pasta “Contextos (F5)” e selecione “Contextos” a esquerda e nesta pasta, no menu Início clique no ícone “Adicionar Contexto”:





Após adicionar um dos tipos de contexto e necessário dar um nome para este contexto e seleccionar o modo de carga do contexto.

11.6. Tipos de Contexto:

11.6.1. Classe RM.NET:

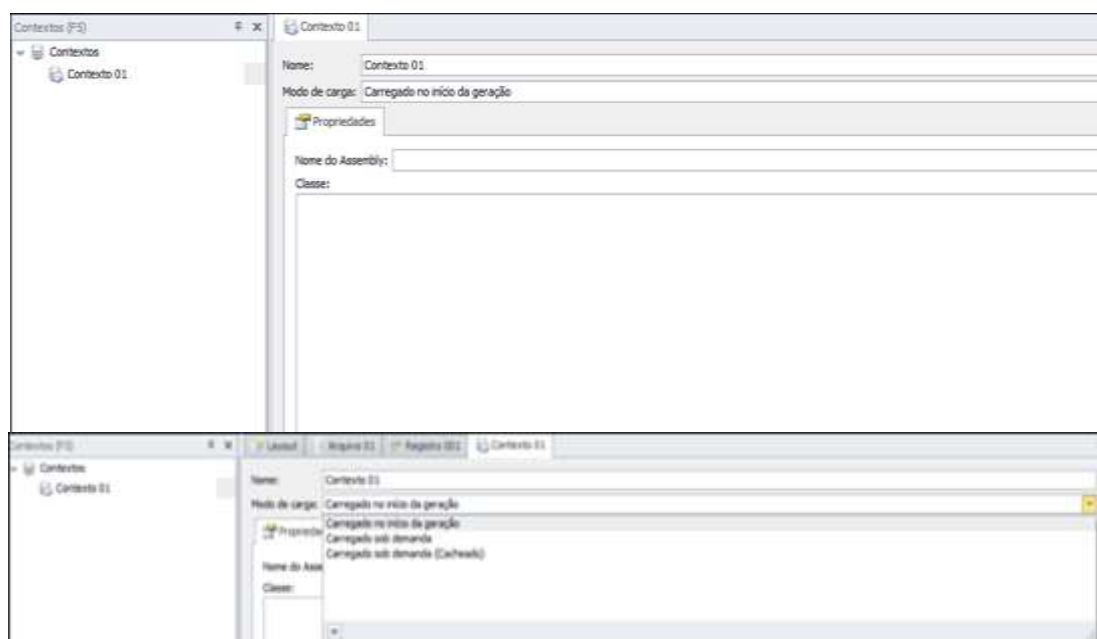
O contexto Classe RM.NET é um contexto “Programado” utilizado somente internamente aqui na TOTVS.

É um contexto que retorna os dados conforme a regra de negócios criada pelo Desenvolvimento do Sistema.

Não é possível a criação de contextos deste tipo pelos analistas que desenvolvem os layouts. Para saber quais os dados que um contexto Classe RM.NET(DLL) retorna é necessário verificar com o Departamento de Suporte (com os responsáveis que criaram o contexto).

Exemplo

Se foi criado um contexto do financeiro, os analistas do financeiro que desenvolveram o contexto deverão informar o que ele retorna.



Nome: definir o nome do contexto.

Modo de carga:

Carregado no início da geração: no início da geração do layout carregar todos os dados. Em layouts muito grandes a geração pode ter o desempenho comprometido. Em síntese, carrega todos os dados e depois a estrutura.

Carregado sob demanda: carrega o contexto sob demanda, somente no momento que é requerido pelo layout. De forma simples, não armazena. Se for necessário carrega de novo.

Carregado sob demanda (Cacheado): carrega sob demanda e só descarrega após utilizar as informações em todos os layouts. Em síntese, mantém na memória.

Exemplo

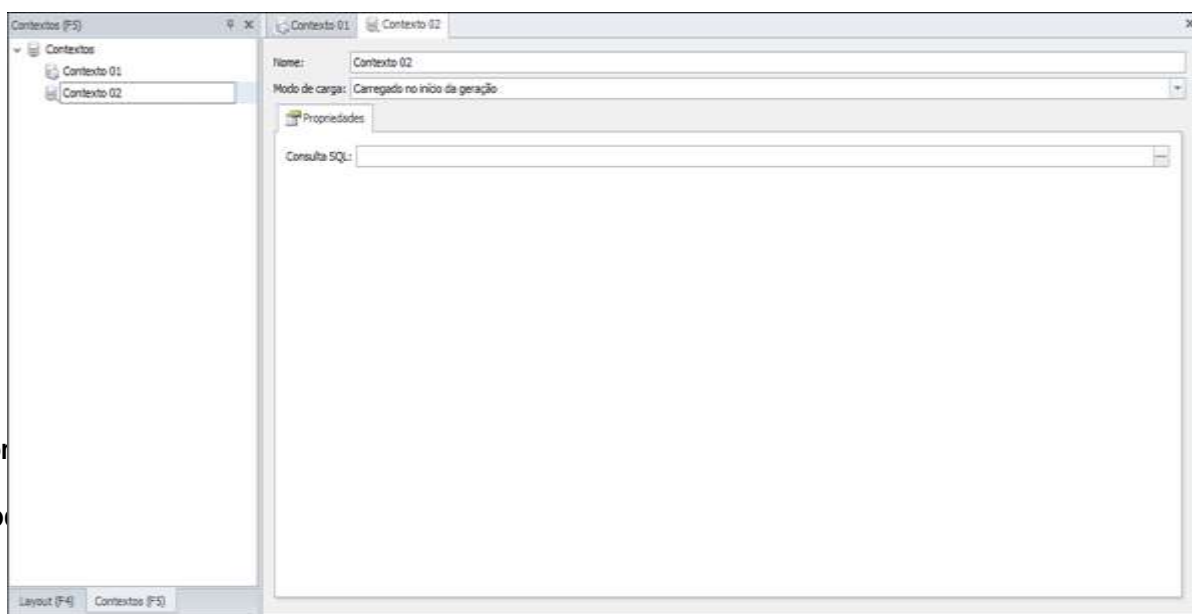
No layout com a seguinte estrutura:

- Registro 01
- Registro 01.01
- Registro 01.02
- Registro 02
- Registro 03

É utilizado a informação dos clientes/fornecedores no registro 01, registro 02, este contexto será carregado na geração do registro 01 e será descarregado somente após a utilização das informações no registro 02.

11.6.1.1. Consulta SQL:

Os parâmetros da SQL deverão ser associados ao parâmetro da geração do layout. Para fazer isso é necessário clicar com o botão esquerdo do mouse sobre o parâmetro, segurar e arrastar o parâmetro para o parâmetro da SQL e soltar o botão do mouse.



Carregado no início da geração: no início da geração do layout carregar todos os dados. Em layouts muito grandes a geração pode ter o desempenho comprometido. Em síntese, carrega todos os dados e depois a estrutura.

Carregado sob demanda: carrega o contexto sob demanda, somente no momento que é requerido pelo layout. De forma simples, não armazena. Se for necessário carrega de novo.

Carregado sob demanda (Cacheado): carrega sob demanda e só descarrega após utilizar as informações em todos os layouts. Em síntese, mantém na memória.

Exemplo

No layout com a seguinte estrutura:

- Registro 01
- Registro 01.01
- Registro 01.02
- Registro 02
- Registro 03

É utilizado a informação dos clientes/fornecedores no registro 01, registro 02, este contexto será carregado na geração do registro 01 e será descarregado somente após a utilização das informações no registro 02.

Propriedades:

Consulta SQL: Linguagem de Consulta Estruturada ou SQL, é a linguagem de pesquisa declarativa padrão para banco de dados relacional (base de dados relacional).

Exemplo

Crie a consulta SQL:

```
SELECT TOP 3 IDLAN,  
            NUMERODOCUMENTO,  
            CODCOLIGADA  
FROM FLAN  
WHERE FLAN.DATAEMISSAO > 01 / 01 / 2009  
      AND FLAN.PAGREC = 1  
      AND FLAN.VALORORIGINAL = 0
```

Definições

TABELA FLAN: Aplicativo Gestão Financeira – Lançamentos Financeiros.

SELECT: O comando SELECT permite recuperar os dados de um objeto do banco de dados, como uma tabela, view e, em alguns casos, uma stored procedure (alguns bancos de dados permitem a criação de procedimentos que retornam valor).

TOP 3: TOP(Transact-SQL). Limita as linhas retornadas em um conjunto de resultados de consulta a um número ou percentual de linhas no SQL Server 2014. Quando TOP é usado junto com a cláusula ORDER BY, o conjunto de resultados é limitado ao primeiro número N de linhas ordenadas; caso contrário, ele retorna o primeiro número N de linhas em uma ordem indefinida. Use esta cláusula para especificar o número de linhas retornado de uma instrução SELECT ou afetado por uma instrução INSERT, UPDATE, MERGE ou DELETE.

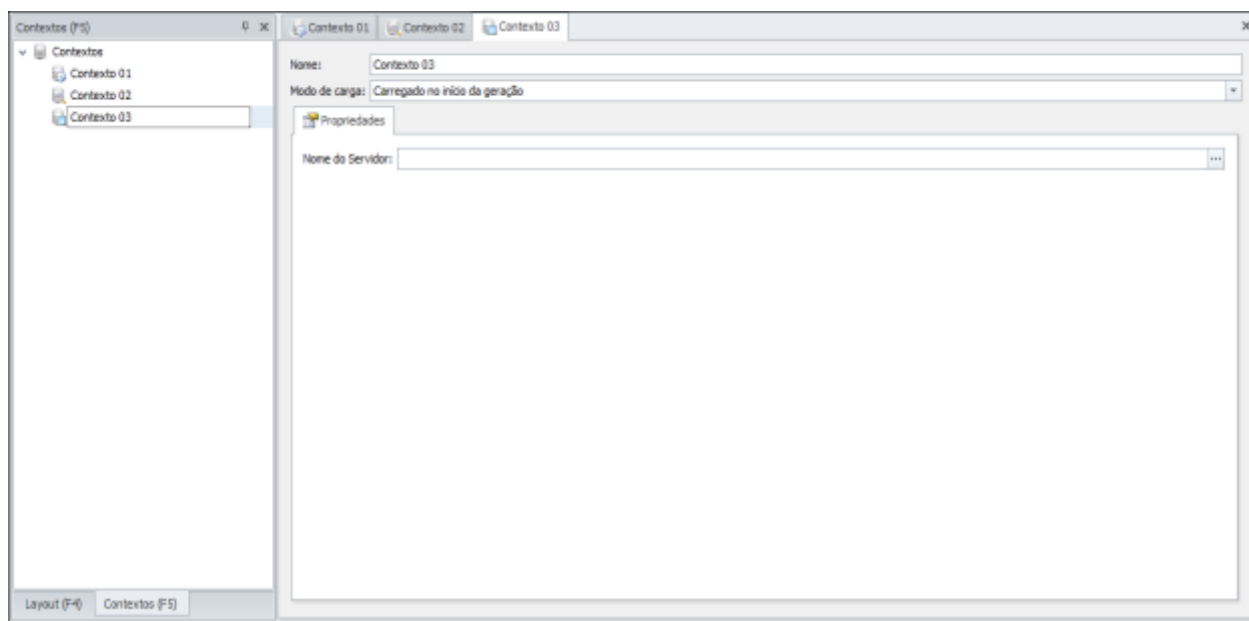
Limita as linhas retornadas em um conjunto de resultados de consulta a um número ... de TOP (10) afetar 10 linhas, dessas linhas, 7 podem ser atualizadas e 3 ...

FROM: Indicar a tabela. O caractere * representa todos os campos. Apesar de prático, este caractere não é muito utilizado, pois, para o SGBD é mais rápido receber o comando com todos os campos explicitados. O uso do * obriga o servidor a consultar quais são os campos antes de efetuar a busca dos dados, criando mais um passo no processo SQL passar condições de filtragem.

WHERE: Condição da consulta SQL. A cláusula Where permite ao comando.

11.6.1.2. Cadastro do Sistema (Tela do Sistema):

Selecionar o cadastro do sistema que será utilizado como contexto do layout. Localizar o cadastro do sistema no lookup, pesquisar e selecionar o resultado desejado. É possível a associação de um parâmetro ao contexto.



Nome: definir o nome do contexto.

Mode de Carga:

Carregado no início da geração: no início da geração do layout carregar todos os dados. Em layouts muito grandes a geração pode ter o desempenho comprometido. Em síntese, carrega todos os dados e depois a estrutura.

Carregado sob demanda: carrega o contexto sob demanda, somente no momento que é requerido pelo layout. De forma simples, não armazena. Se for necessário carrega de novo.

Carregado sob demanda (Cacheado): carrega sob demanda e só descarrega após utilizar as informações em todos os layouts. Em síntese, mantém na memória.

Exemplo

No layout com a seguinte estrutura:

- Registro 01
- Registro 01.01
- Registro 01.02
- Registro 02
- Registro 03

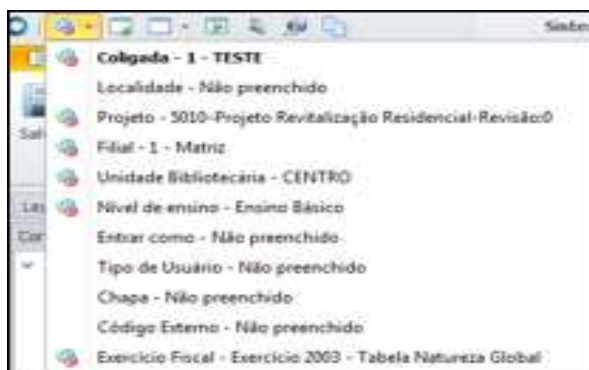
É utilizado a informação dos clientes/fornecedores no registro 01, registro 02, este contexto será carregado na geração do registro 01 e será descarregado somente após a utilização das informações no registro 02.

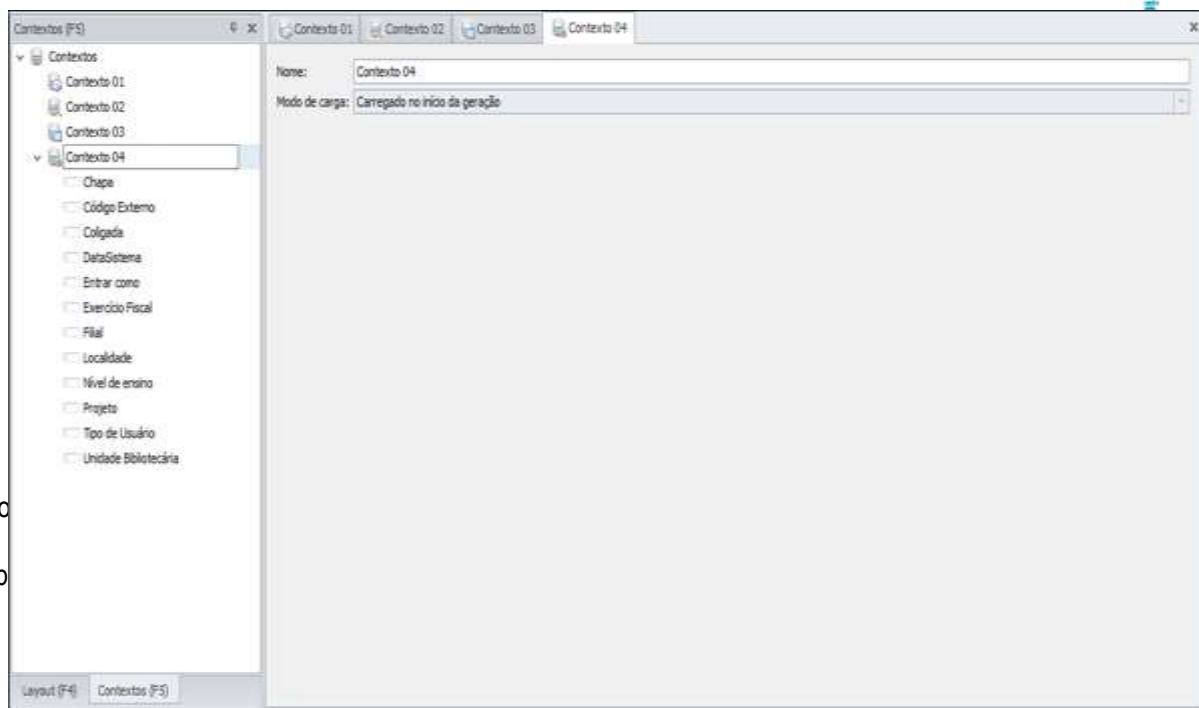
11.6.1.3. Contexto da Aplicação:

Esse contexto retorna os contextos de aplicação (exemplo abaixo) disponíveis no aplicativo.

Exemplo

Contextos de aplicação:





Nome: definir o nome do

Modo de carga: não é p

Propriedades:

Adicionar:

Texto: permite que o operador digite informação do tipo texto a ser gerada no layout.

Número Inteiro: O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Definir se é tipo número inteiro.

Número Decimal: O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada. Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Definir se é tipo número decimal.

Data: Permite que o operador digite a data a ser gerada no layout.

Lógico: O tipo de dados do resultado é determinado pela aplicação das regras de precedência de tipos de dados em relação aos tipos de dados de expressões de entrada.

Para obter mais informações, consulte Precedência de tipo de dados (Transact-SQL). Definir se é tipo lógico. Permite que o operador digite o tipo lógico a ser gerado no layout.

Lista: Criar uma lista a ser disponibilizada na visão da geração do layout e permitir ao usuário a seleção dos itens dessa lista.

Lookup: Definir qual tabela do aplicativo estará disponível na visão da geração do layout para o operador filtrar e selecionar o item a ser gerado neste layout. Selecionar as colunas na seção campos. Selecionar a coluna que será utilizada na descrição do filtro na seção descrição e o código que será passado como parâmetro da action para o layout na seção código. Nas seções descrição e código estarão disponíveis as colunas disponíveis na seção campos.

Filtro: Permite que o usuário use os filtros. Em configuração, selecionar a tabela que deverá ser utilizada no filtro dos parâmetros do layout. O filtro será utilizado no layout após a criação.

Remover: Excluir as opções de parâmetros.

Nome: Nomear o parâmetro.

Título: Definir a denominação resumida do parâmetro.

Descrição: Definir a descrição completa do parâmetro.

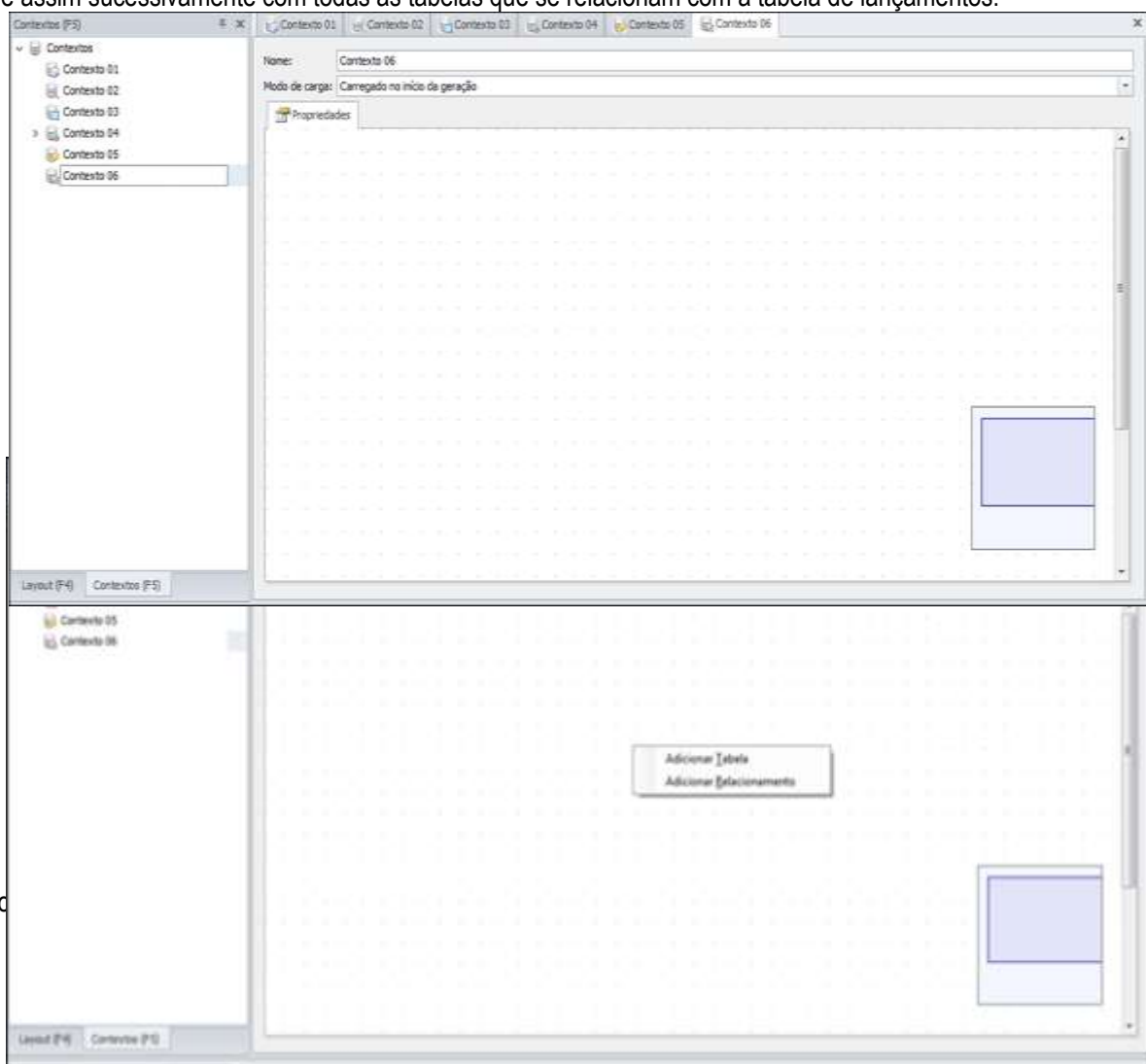
Obrigatório: Definir se o parâmetro a ser preenchido na geração do layout será obrigatório.

11.6.1.4. Conjunto de Dados:

O contexto conjunto de dados refere-se às tabelas do sistema disponíveis no banco de dados. Quando utilizamos o contexto conjunto de dados não é necessário adicionar várias tabelas e relacioná-las para obtermos as informações como fonte de dados para utilização no layout, pois todas as tabelas que se relacionam no banco com esta tabela estão disponíveis no contexto.

Exemplo

Selecionamos a tabela de lançamentos FLAN temos disponível no contexto as informações por exemplo da tabela de coligadas GCOLIGADA e assim sucessivamente com todas as tabelas que se relacionam com a tabela de lançamentos.



Nome: definir o nome do

Mode de Carga:

Carregado no início da geração: No início da geração do layout carregar todos os dados. Em layouts muito grandes a geração pode ter o desempenho comprometido. Em síntese, carrega todos os dados e depois a estrutura.

Carregado sob demanda: carrega o contexto sob demanda, somente no momento que é requerido pelo layout. De forma simples, não armazena. Se for necessário carrega de novo.

Carregado sob demanda (Cacheado): carrega sob demanda e só descarrega após utilizar as informações em todos os layouts. Em síntese, mantém na memória.

Exemplo

No layout com a seguinte estrutura:

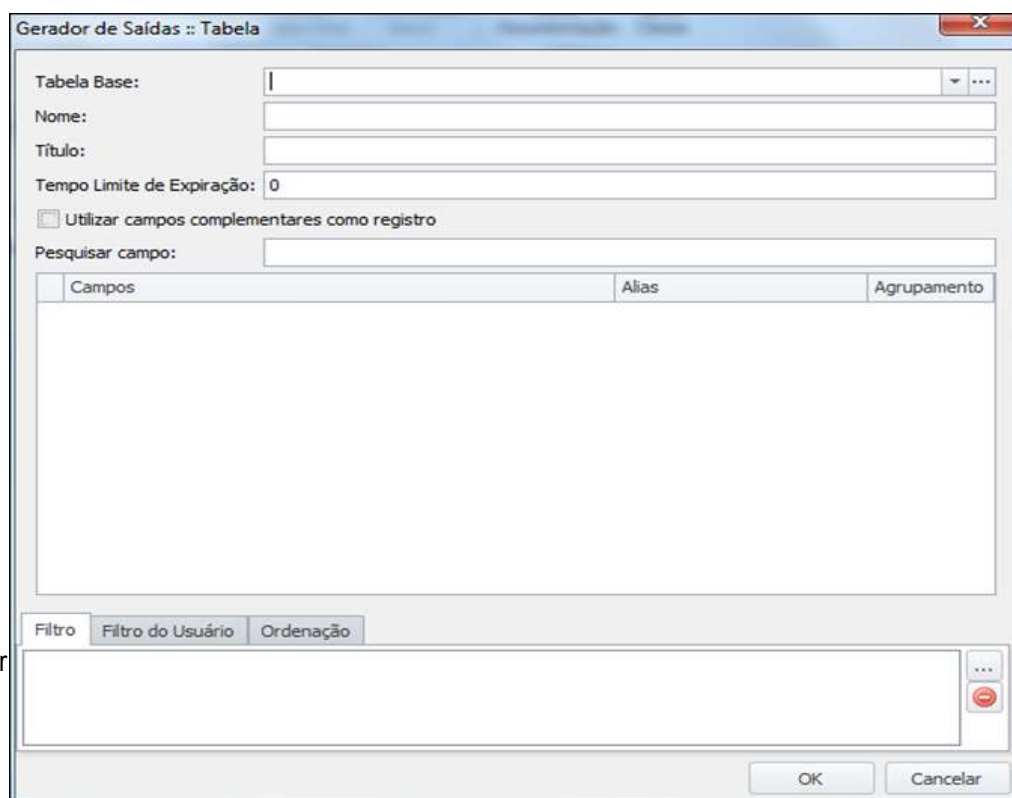
- Registro 01
- Registro 01.01
- Registro 01.02
- Registro 02
- Registro 03

É utilizado a informação dos clientes/fornecedores no registro 01, registro 02, este contexto será carregado na geração do registro 01 e será descarregado somente após a utilização das informações no registro 02.

Propriedades:

Botão direito do mouse selecione:

Adicionar Tabela:



Gerador de Saídas :: Tabela

Tabela Base:

Nome:

Título:

Tempo Limite de Expiração:

☐ Utilizar campos complementares como registro

Pesquisar campo:

Campos	Alias	Agrupamento
--------	-------	-------------

Filtro Filtro do Usuário Ordenação

OK Cancelar

Tabela Base: pesquisar

Nome: o nome da tabela do aplicativo é preenchido automaticamente. Exemplo: PFUNC.

Título: a denominação da tabela. Exemplo: Funcionários.

Utilizar campos complementares como registro: definir se será usado campo complementar.

Pesquisar campo: filtrar o campo a ser usado no layout.

Filtro: alterar ou remover o filtro.

Ícones (a direita):

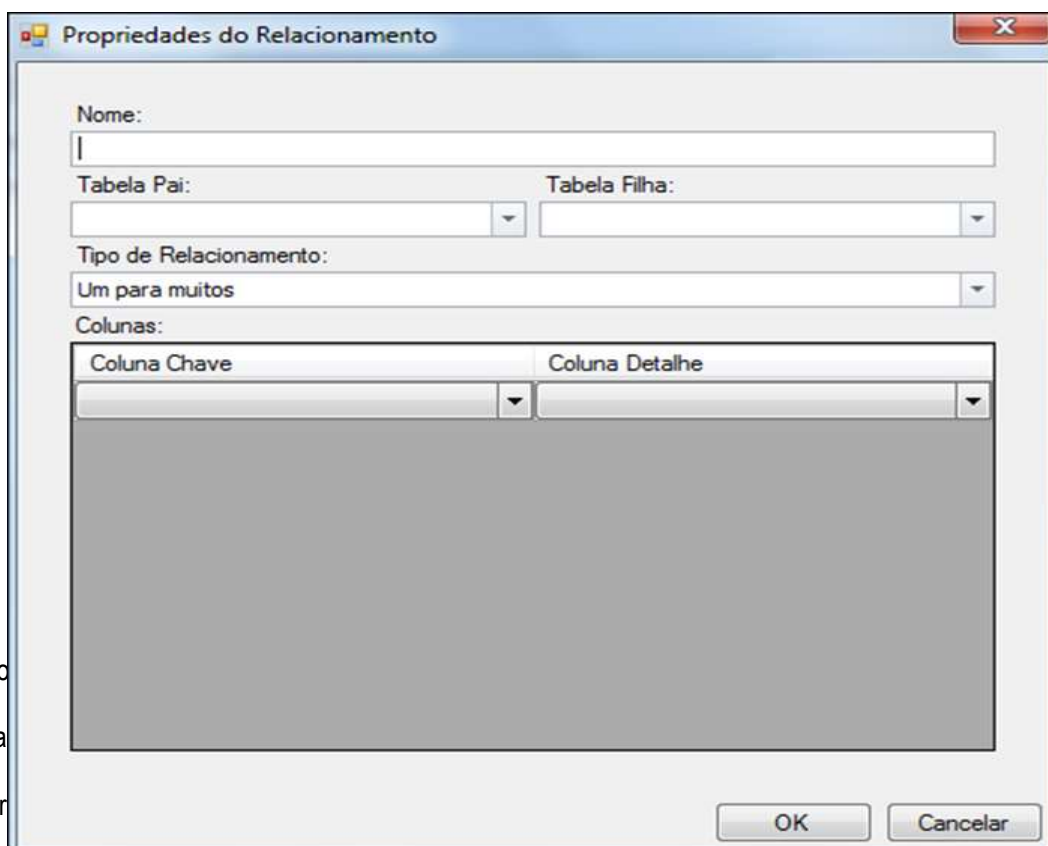
Alterar Filtro.

Excluir Filtro.

Filtro do Usuário: definir a tabela base do filtro do usuário.

Ordenação: ordenação dos campos.

Adicionar Relacionamento:



The image shows a dialog box titled "Propriedades do Relacionamento" (Relationship Properties). It contains the following fields and controls:

- Nome:** A text input field.
- Tabela Pai:** A dropdown menu.
- Tabela Filha:** A dropdown menu.
- Tipo de Relacionamento:** A dropdown menu with the selected value "Um para muitos".
- Colunas:** A section with two columns: "Coluna Chave" and "Coluna Detalhe". Each column has a dropdown menu.
- At the bottom right, there are "OK" and "Cancelar" buttons.

Nome: definir o nome do

Tabela Pai: selecionar a

Tabela Filha: selecionar

Colunas:

Coluna Chave: definir o campo para a coluna chave.

Coluna Detalhe: definir o campo para a coluna detalhe.

Conceitos:

Uma chave primária é utilizada para identificar de forma única cada linha numa tabela. Pode fazer parte do próprio registo atual ou pode ser um campo artificial (um que não tenha nada que ver com o registo atual). Uma chave primária pode ser composta por um ou mais campos numa tabela.

Sob o ponto de vista de um banco de dados relacional, referem-se aos conjuntos de um ou mais campos, cujos valores, considerando a combinação de valores em caso de mais de uma chave primária, nunca se repetem na mesma tabela e, desta forma, podem ser usadas como um índice de referência para criar relacionamentos com as demais tabelas do banco de dados (daí vem o nome banco de dados relacional). Portanto, uma chave primária nunca pode ter valor nulo, nem repetição. Simplificando, quando a chave primária é simples, ou seja, é formada por um único campo da tabela, esse campo não pode ter dois ou mais registros de mesmo valor e também não pode conter nenhum registro nulo. Se a chave primária é composta, ou seja, formada por mais de um campo, os valores de cada campo podem se repetir, mas nunca a combinação desses valores.

Exemplo: a tabela 'Livros_Autores' tem como chave primária (cod_livro, cod_autor). Podem existir nessa tabela os registros:

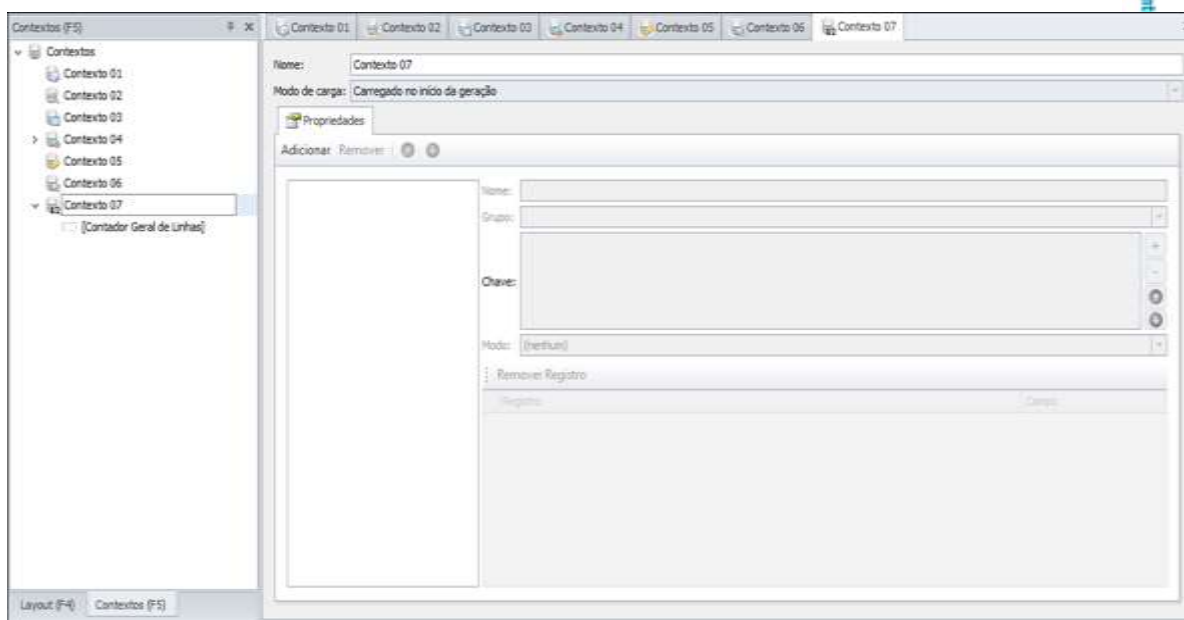
- (5, 9);
- (5, 10);
- (4, 9);
- (9, 5).

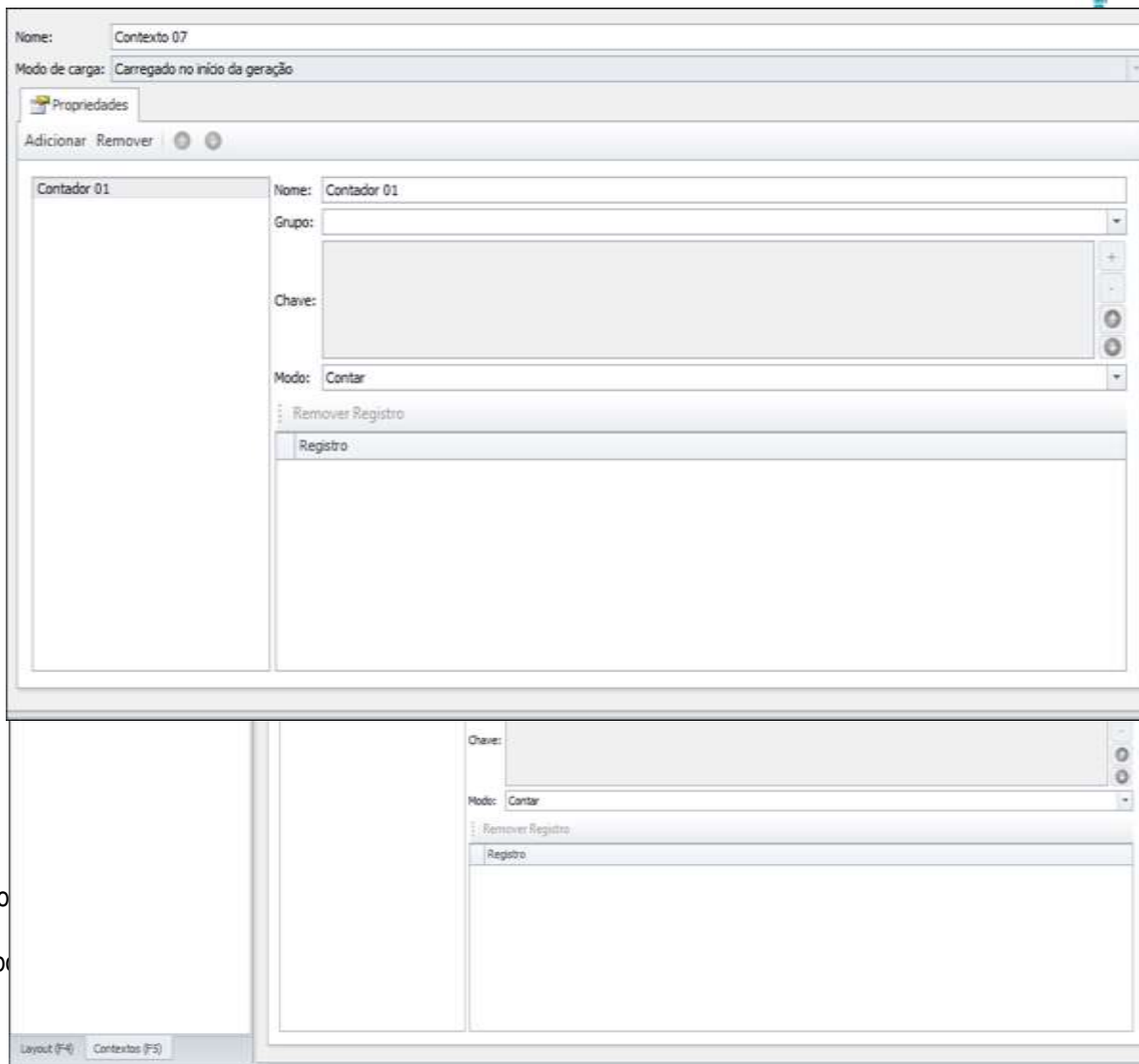
Mas não podem existir dois registros (5, 9).

11.6.1.5. Contadores:

Contexto utilizado para gerar contadores do layout. Existe um tipo de contar nativo que gera um contador sequencial de linhas.

Para criar um contador de um registro específico é necessário adicionar um novo contador e adicionar a ele o registro que deseja contar.





Nome: definir o nome do

Mode de carga: não é p

Propriedades:

Adicionar:

Nome: definir o nome do contador

Modo:

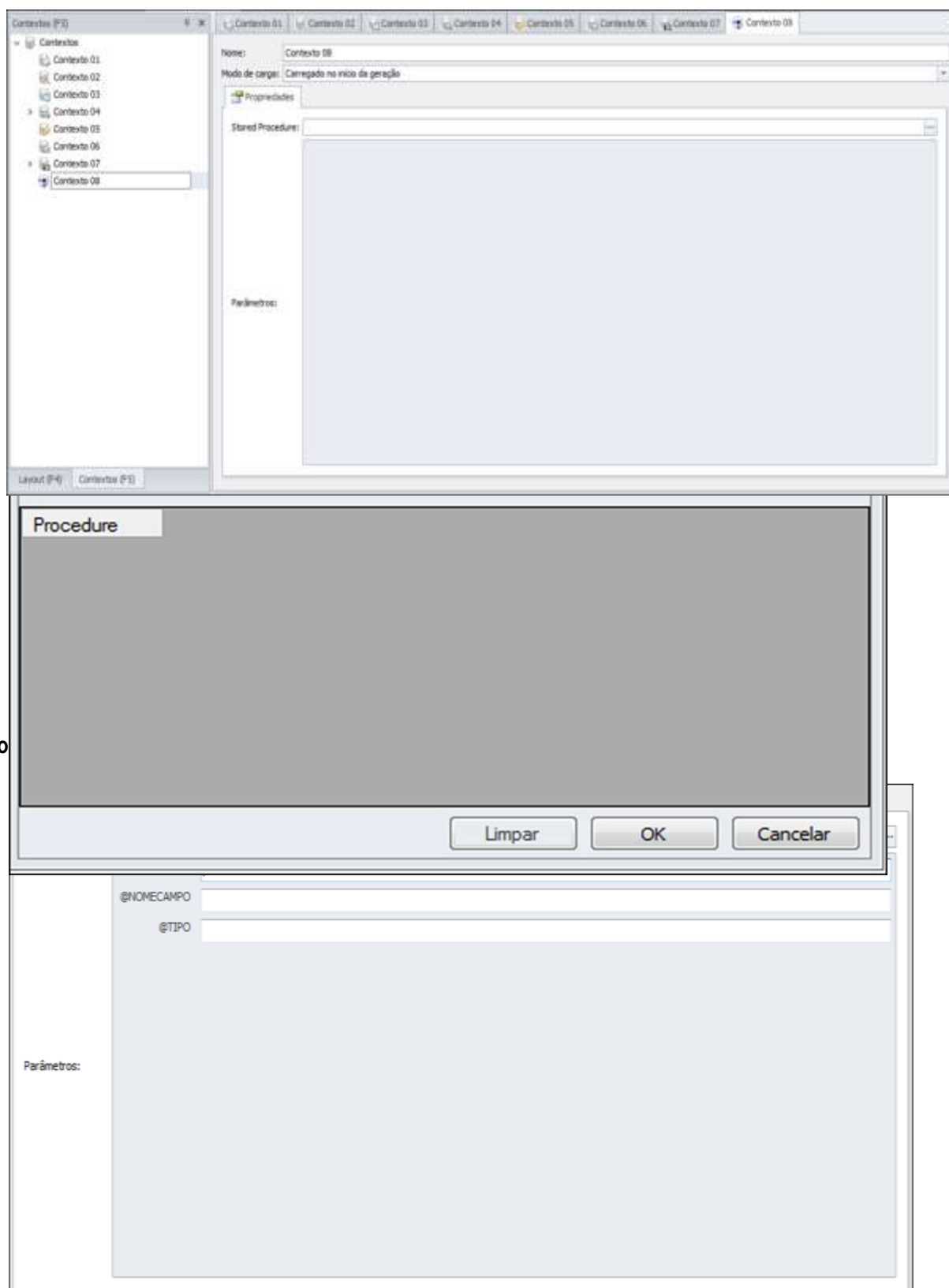
- Contar
- Somar
- Mínimo
- Máximo

Registro: para associar o registro ao modo do contador.

11.6.1.6. Stored Procedure:

Selecionado a Procedure que será utilizada como contexto do layout.

O lookup busca todas as Procedures que estão criadas na base de dados. Caso a Procedure tenha parâmetros é necessário informar valores válidos de parâmetros para alimentar o contexto para a montagem do layout. Após a montagem do layout é necessário inserir valores inválidos nos parâmetros para que o contexto não seja carregado somente com os dados gerados para a montagem do layout. Os parâmetros da Procedure devem ter o mesmo nome dos parâmetros criados no layout.



Contextos (F3)

Contexto 01 Contexto 02 Contexto 03 Contexto 04 Contexto 05 Contexto 06 Contexto 07 Contexto 08

Nome: Contexto 08

Modo de carga: Carregado no início da geração

Propriedades

Stored Procedure:

Parâmetros:

Limpar OK Cancelar

Procedure

@NOMECAMPO

@TIPO

Parâmetros:

Exemplo Desenvolvido

Nome: definir o nome do contexto.

Mode de Carga:

Carregado no início da geração: no início da geração do layout carregar todos os dados. Em layouts muito grandes a geração pode ter o desempenho comprometido. Em síntese, carrega todos os dados e depois a estrutura.

Carregado sob demanda: carrega o contexto sob demanda, somente no momento que é requerido pelo layout. De forma simples, não armazena. Se for necessário carrega de novo.

Carregado sob demanda (Cacheado): carrega sob demanda e só descarrega após utilizar as informações em todos os layouts. Em síntese, mantém na memória.

Exemplo

No layout com a seguinte estrutura:

- Registro 01
- Registro 01.01
- Registro 01.02
- Registro 02
- Registro 03

É utilizado a informação dos clientes/fornecedores no registro 01, registro 02, este contexto será carregado na geração do registro 01 e será descarregado somente após a utilização das informações no registro 02.

Propriedades:

- Stored Procedure: associar a stored procedure.
- Parâmetros: definir os parâmetros da stored procedure.

11.6.1.7. Criação de Layout do Gerador de Saídas com Parâmetros

Esta informação ajudará na criação de um Layout do Gerador de Saídas com Parâmetros.

Exemplo:

Sentença SQL utilizada:

```
SELECT          PFUNC.CHAPA,          PFUNC.NOME,          PFDEPEND.NOME AS 'NOME
DEPENDENTE',PFDEPEND.DTNASCIMENTO, PFDEPEND.SEXO
FROM PFUNC INNER JOIN
PFDEPEND ON PFUNC.CODCOLIGADA = PFDEPEND.CODCOLIGADA
AND PFUNC.CHAPA = PFDEPEND.CHAPA
```

WHERE PFUNC.CODSITUACAO = :CODSITUACAO
AND PFDEPEND.SEXO = :SEXO

Contexto Consulta SQL:

Ao adicionar os parâmetros na sentença SQL, é necessário informá-los na tela do Contexto.

Nome:

Modo de carga: Carregado no início da geração

Propriedades

Nome	Tipo	Ligação
CODSITUACAO	System.String	
SEXO	System.String	

Neste exemplo, vamos u

Assim deve ser adicionado ao Layout um contexto de parâmetros da Geração:



Desta Forma é apresentada as configurações para este parâmetro que será apresentado ao gerar o Layout desejado:

Nome:

Modo de carga: Carregado no início da geração

Propriedades

Adicionar - Remover

Nome	Título	Descrição
CODSITUACAO	SITUACAO FUNCIONARIO	PARAMETROS PARA FILTRAR O CODIGO DO FUNCIONARIO

☒ Obrigatório

Gerador de Saida

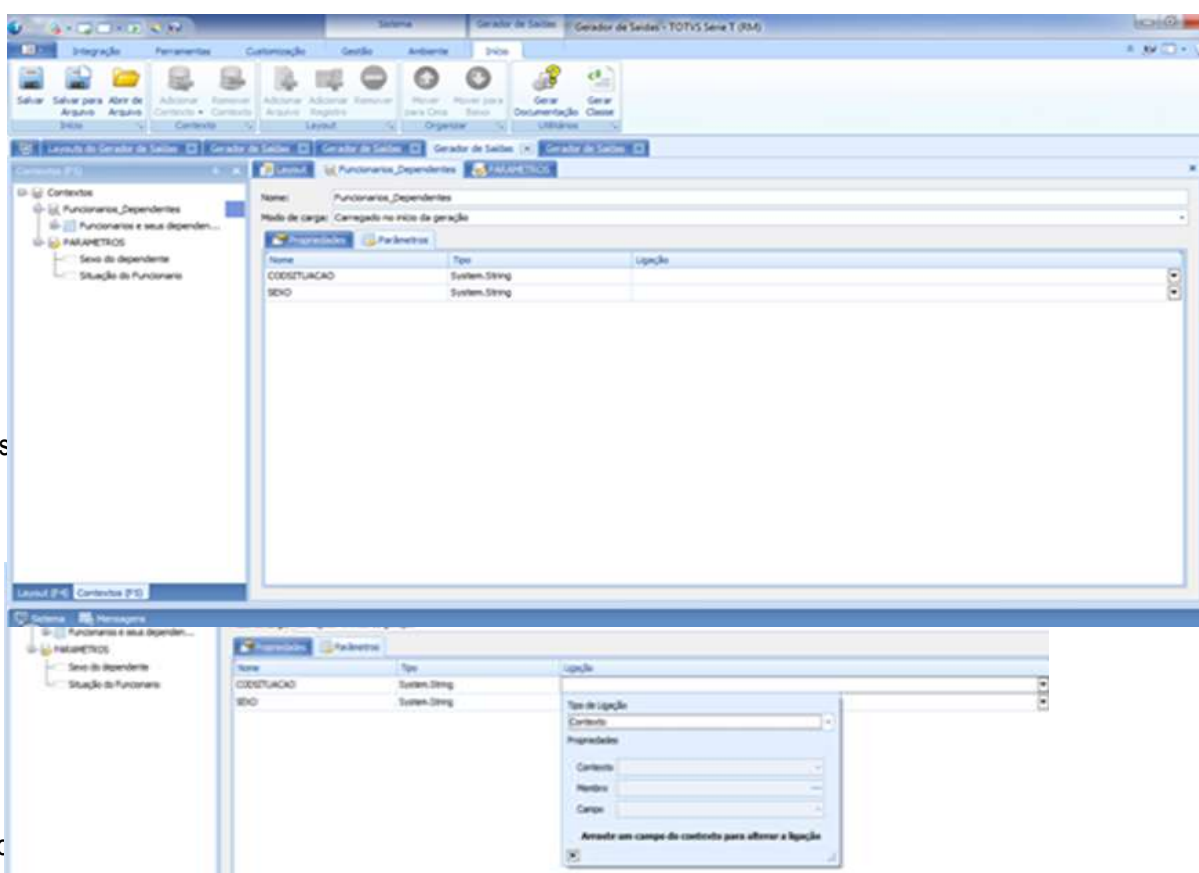


Neste exemplo foram configurados os dois parâmetros da sentença:

- CODSITUACAO – Situação do Funcionário;
- SEXO – Sexo dos Dependentes;

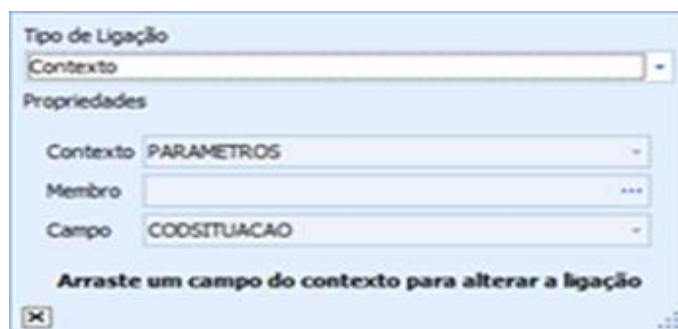
Após esta configuração dos parâmetros, se faz necessário configurar a ligação deste parâmetro com o contexto da sentença SQL.

Nas configurações do contexto da SQL na guia dos parâmetros temos esta opção como print abaixo:



Para que o contexto dos
de ligação:

Consulta Após arrastar c



Assim após estas parametrizações, o Layout ao ser Gerado solicitará a passagem dos parâmetros:

- CODSITUACAO;
- SEXO.

Observação

Repare que os nomes de dependentes são apenas do Sexo Feminino como solicitamos no parâmetro:



11.6.1.8. Operações com Layout

Para efetuar as operações com os layouts, selecione o layout desejado e clique em processos.

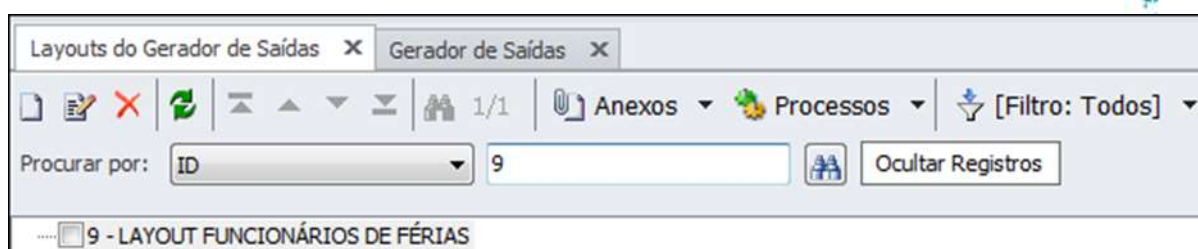
As operações são:

Geração de saídas: é utilizada para a geração do layout. Por padrão o layout é gravado em: C:\totvs\CorporeRM\RM.Net\Saídas ou no diretório parametrizado para geração do layout.

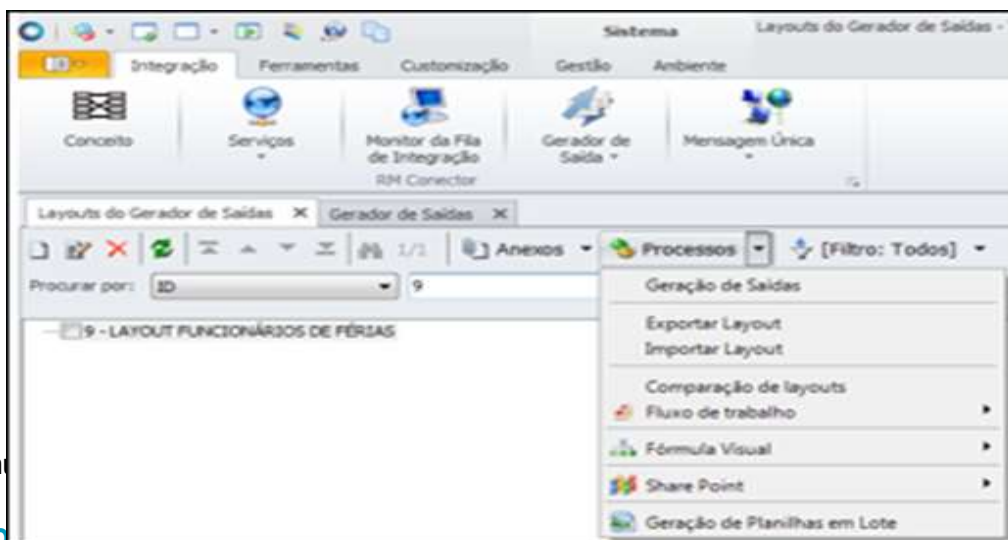
Export layout.

Importar layout.

Comparação de layout (tópico 9 – Sumário): recurso muito útil para comparar diferenças entre dois layouts. Para utilizar esta funcionalidade clique no botão comparação dos layouts e selecione os layouts, que deseja comparar e atualizar a versão. Se, a opção exibir somente as propriedades alteradas estiver marcada será visualizada somente a diferença entre os layouts.



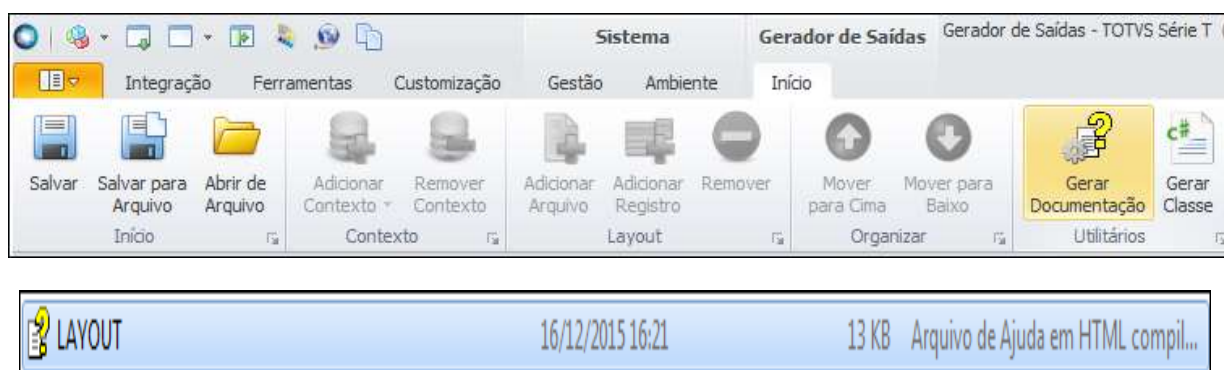
Gerador de Saídas



Os demais itens do menu

11.7. Gerar Do

Para gerar a documentação do layout criado, edite o layout e clique no ícone Gerar Documentação e escolha o diretório de sua preferência para salvar o arquivo de documentação.

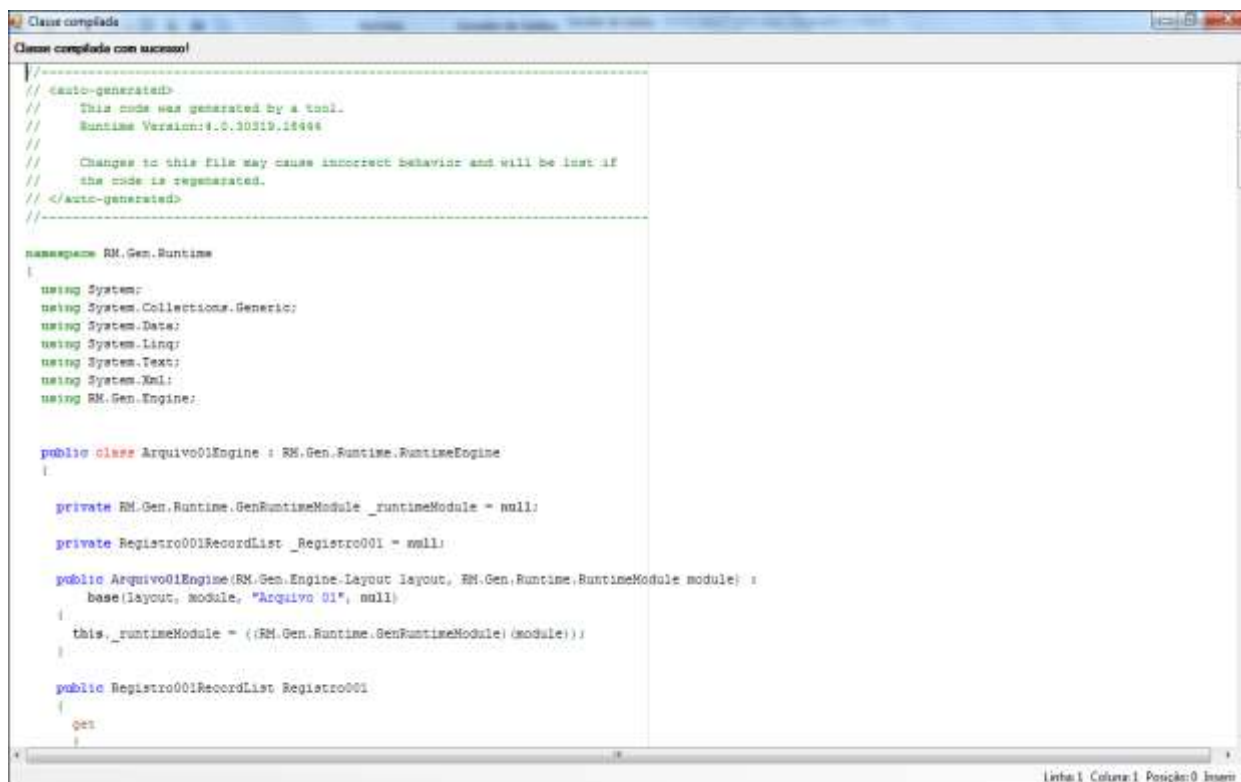




11.8. Gerar Classe

Esta opção gera código fonte para o Desenvolvedor utilizar para debugar em caso de problema.





```
// Classe compilada
//
// Classe compilada com sucesso!
//
// -----
// <auto-generated>
// This code was generated by a tool.
// Runtime Version: 4.0.30319.16446
//
// Changes to this file may cause incorrect behavior and will be lost if
// the code is regenerated.
// </auto-generated>
// -----
//
namespace RM.Gen.Runtime
{
    using System;
    using System.Collections.Generic;
    using System.Data;
    using System.Linq;
    using System.Text;
    using System.Xml;
    using RM.Gen.Engine;

    public class Arquivo01Engine : RM.Gen.Runtime.RuntimeEngine
    {
        private RM.Gen.Runtime.GenRuntimeModule _runtimeModule = null;
        private Registro001RecordList _Registro001 = null;

        public Arquivo01Engine(RM.Gen.Engine.Layout layout, RM.Gen.Runtime.RuntimeModule module) :
            base(layout, module, "Arquivo 01", null)
        {
            this._runtimeModule = (RM.Gen.Runtime.GenRuntimeModule)(module);
        }

        public Registro001RecordList Registro001
        {
            get
            {
            }
        }
    }
}
```

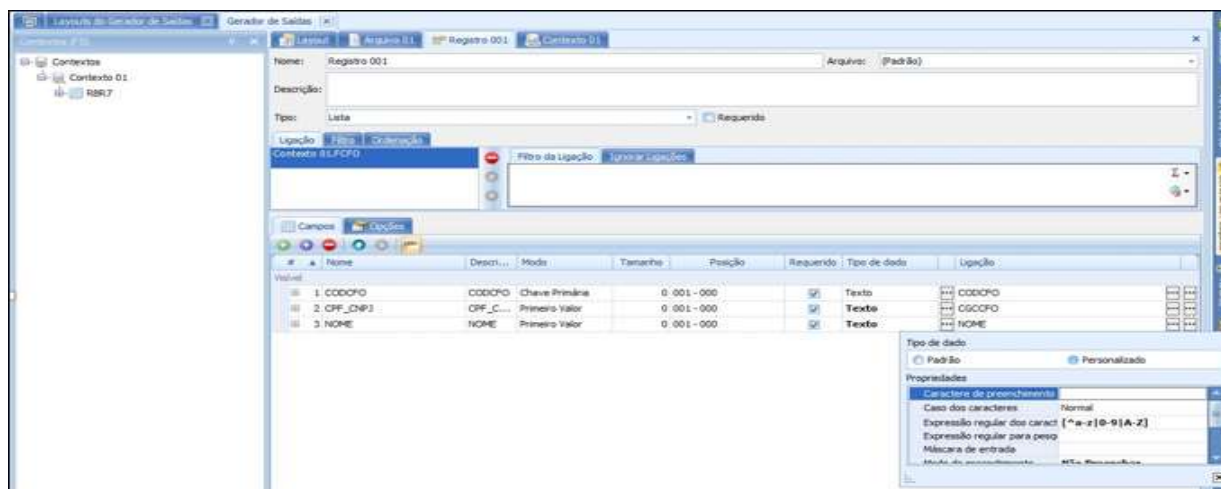
12. Informações Complementares:

12.1. Remoção de Caracteres em Campos do Gerador de Saídas

Abaixo segue um exemplo de como remover caracteres de um campo do tipo CPF/CNPJ como mostrado no exemplo.

Foi utilizado o campo CNPJ para demonstração, assim:

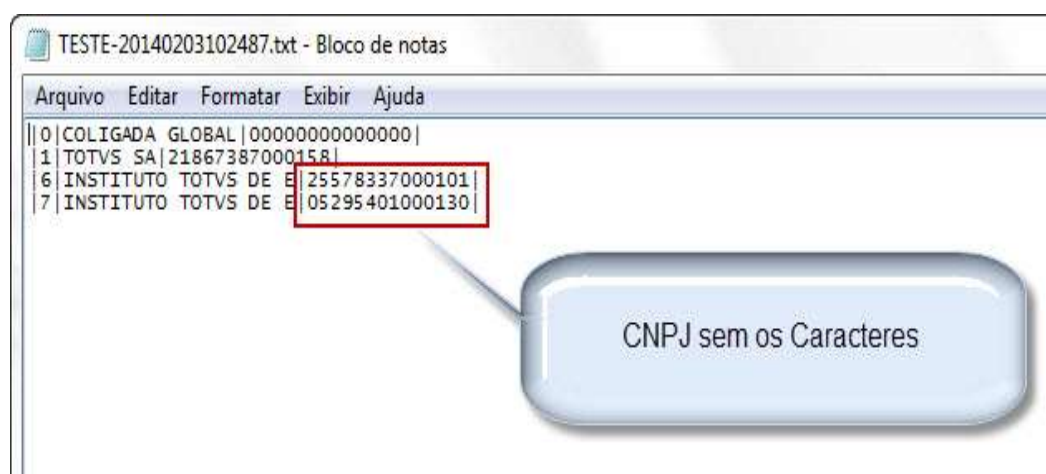
Acessando o Registro do Layout, temos o campo CNPJ que corresponde ao campo da base de dados (GCOLIGADA.CGC). Assim clicamos no Botão “Editar Propriedades Avançadas do Registro” como mostrado no Print abaixo:



São apresentadas as opções de definição deste registro.

Assim selecionamos a opção "PERSONALIZADO" e na parte "EXPRESSÃO REGULAR DOS CARACTERES" colocamos o Valor "[^a-z|0-9|A-Z]"

Após isto obtemos o resultado abaixo:

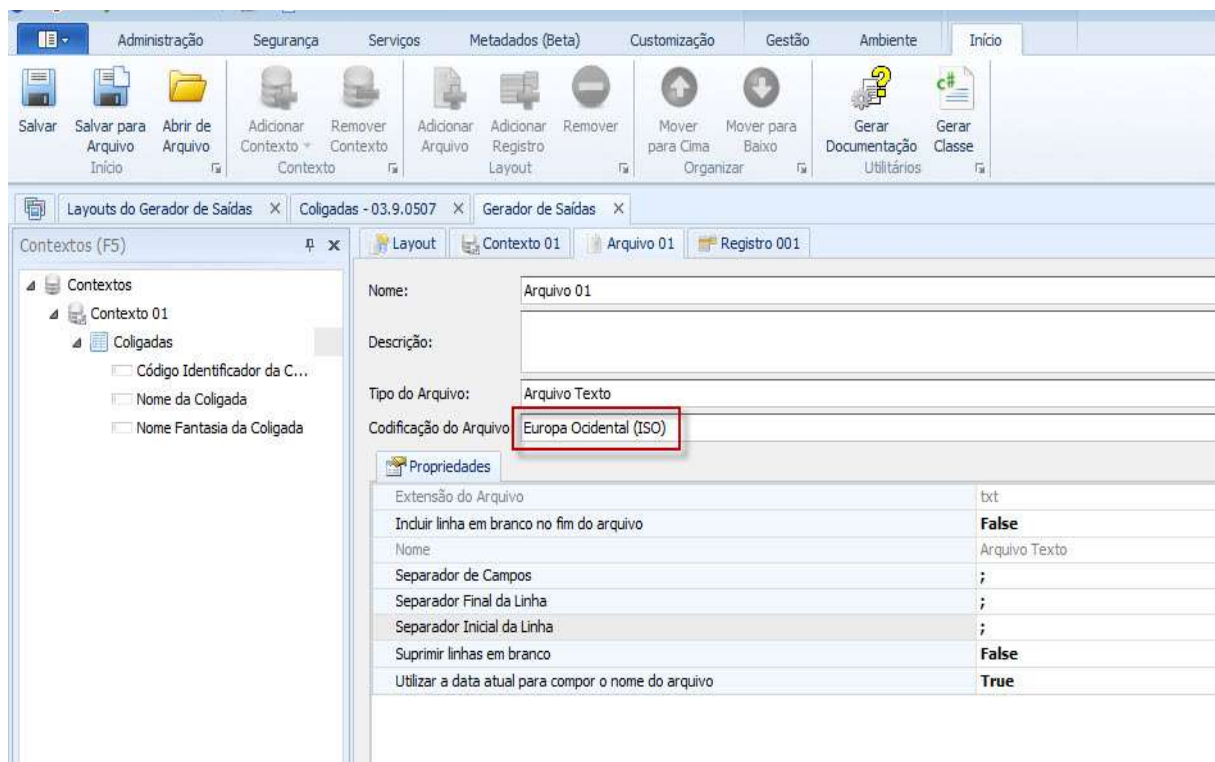


Observação

Observação: para permitir o uso de letras acentuadas e espaçamento, colocamos o valor "[^a-z|0-9|A-Z|s|áéíóúâêîôûãõ]".

Incluindo o Caractere Ç

Para incluir o caractere Ç na impressão de layout do gerador de saídas deverá ser incluído no campo: codificação do Arquivo selecionando a opção Europa Ocidental (ISO).



12. Exercício

12.2. Layout de Saldo de Férias Funcionários

Crie um novo layout com base nas informações abaixo:

- Acesse Gerador de Saídas e clique em layouts.
- Clique em incluir para criar novo layout.
- Preencha os campos da capa do novo layout:
 - ID: (gerado pelo aplicativo)
 - Nome: Layout Saldo de Férias Funcionários
 - Descrição: Layout Saldo de Férias Funcionários por Coligada
 - Categoria: RH
 - Layout Pai: Layouts Folha Pagamento
 - Tipo: (deixar em branco)
- Salve a capa do layout e duplo clique para editar o novo layout.

- Selecione a subpasta Layout (F4), selecione arquivos e clique no ícone adicionar arquivo:

- Nome: Saldo de Férias Funcionários
- Descrição: Saldo de Férias Funcionários TOTVS S.A.
- Tipo de Arquivo: Arquivo Texto
- Codificação do arquivo: (padrão)

Propriedades:

- Extensão do arquivo: txt
- Gerar nome do arquivo dinamicamente: true
- Incluir linha em branco no fim do arquivo: true
- Nome: (gerado pelo aplicativo)
- Separador de campos: (deixar em branco)
- Separador final de linha: (deixar em branco)
- Separador inicial de linha: (deixar em branco)
- Suprimir linhas em branco: (deixar em branco)
- Utilizar a data atual para compor o nome do arquivo: false
- Selecione a subpasta Layout (F4), selecione registros e clique no ícone adicionar registro:

Nome: Férias Funcionários

Arquivo: Saldo de Férias Funcionários

Descrição: Saldo de Férias Funcionários TOTVS S.A.

Tipo: Lista

Requerido: (deixar desmarcado)

Ligação: arrastar os contextos usados neste exercício (quando solicitado).

Campos:

- Nome: COLIGADA, CHAPA, NOME, SALARIO, SALDOFERIAS.
- Descrição: COLIGADA, CHAPA, NOME, SALARIO, SALDOFERIAS.
- Modo: Chave primária, chave primária, primeiro valor, primeiro valor, primeiro valor, primeiro valor.

- Tamanho: definir o tamanho que o campo ocupa no layout.
- Posição: conferir a posição do campo no layout (conforme tamanho).
- Requerido: (deixar em branco)
- Tipo de dado: (definir para cada campo solicitado neste exercício)
- Ligação: conferir a associação feita no tópico f. Ligação.
 - Ligação sob demanda: (deixar em branco)
 - Filtro: (deixar em branco)
 - Ordenação: (deixar em branco)
 - Filtro da ligação: selecionar o campo do contexto que será utilizado no filtro e montar a expressão.
 - Ignorar ligações: (deixar em branco)
 - Funções: (deixar em branco)
 - Parâmetros: (deixar em branco)

Selecione a subpasta Contextos (F5), selecione contextos e clique no ícone adicionar contexto “Contexto da Aplicação”:

- Retorne no registro;
- Arraste os campos do contexto (COLIGADA, CHAPA) para os campos do registro.

Selecione a subpasta Contextos (F5), selecione contextos e clique no ícone adicionar contexto “Conjunto de Dados”:

- Retorne no registro;
- Arraste os campos do contexto (NOME, SALARIO, SALDOFERIAS) para os campos do registro.

Salve o layout;

Acesse menu Integração / Gerador de Saída / Layouts, selecione (marque) o layout a ser gerado, clique em Processos / Geração de Saídas:

- Clique em avançar;
- Clique em executar;
- Clique em fechar;
- Localize o layout gerado e confira as informações;
- Altere a forma de apresentação do layout em formatação padrão layout.

