

Guia do Usuário

Procedurais

Contabilização Universal

Versão 12.0 Release 27.02.2025

Menu de Procedurais

- [Gera movimento contabilidade com partida simples](#)
- [Gera movimento contabilidade com partida dobrada](#)

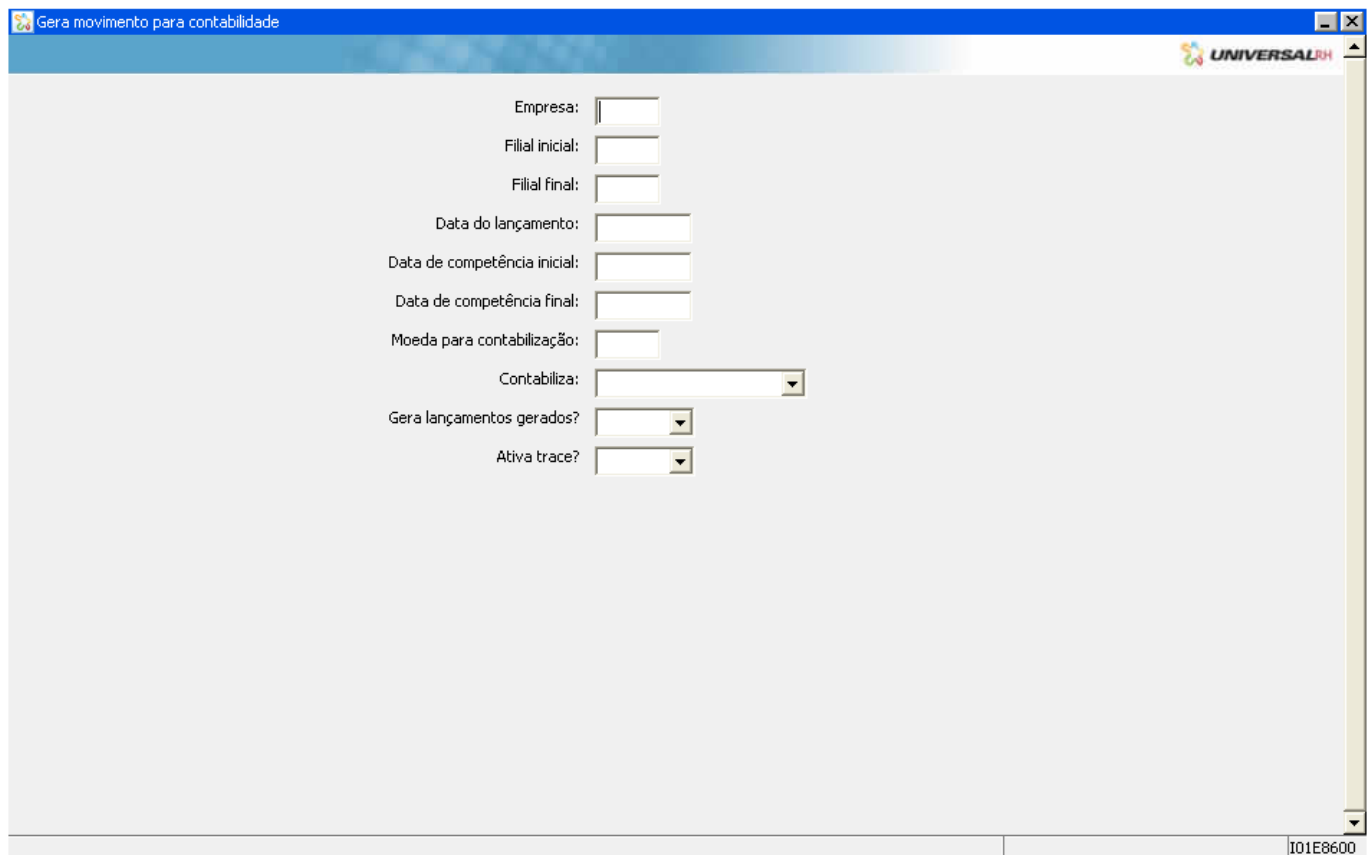


I018600 – Gera movimento para contabilidade

Código do Programa : I018600
Forms de Parâmetro : I01E8600
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Gera movimento para contabilidade
Específico :

Objetivo do Programa:

O objetivo do programa é gerar a Contabilização da Folha de Pagamento.



Descrição dos Campos de Parâmetros:

Competência inicial e final:

São campos de parâmetros destinados a receber os valores que determinam o intervalo de competências que deverão participar do processo. Serão selecionados os registros da tabela de movimentos_calculados ou elo_movimentos_calculados conforme parâmetro **Contabiliza** e conforme intervalo das datas de competência.

Data do lançamento:

4 - Integração Contabilidade (I01)

Campo de parâmetro destinado a receber o valor da data de Lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas Lançamentos_abertos e Lançamentos_gerados.

Contabiliza:

Campo de parâmetro destinado a receber a informação do modo de contabilização que deverá participar do processo.

As opções são:

- Reg Emprego;
- Elo Contrato.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

O usuário deverá implementar na folha, o cadastramento de:

Descrição de Históricos;

Descrição de Contas Contábeis;

Lançamentos Contábeis (esta transação tem por objetivo criar as regras de contabilização para qualquer evento que se julgar importante contabilizar).

Critério de Ordenação, Quebras, outras particularidades:

O critério de aglutinação será definido pelo usuário, usando principalmente, a transação Lançamento Contábil. Logo, este procedural apenas lê o que estiver cadastrado no Lançamento Contábil, para criar o movimento para contabilidade. Os atributos da transação de Lançamentos Contábeis usados para fazer o enquadramento serão os seguintes:

- EVENTO_INICIAL,
- EVENTO_FINAL,
- FILIAL_INICIAL,
- FILIAL_FINAL,
- CHAPA_INICIAL,
- CHAPA_FINAL,
- UORG_INICIAL,
- UORG_FINAL,
- MAO_DE_OBRA_INICIAL,
- MAO_DE_OBRA_FINAL.

Para que os lançamentos gerados sejam acumulados por Unidade de Negócio deverá ser customizado o Runtime [ACUMUNEG](#).

Para que os lançamentos gerados sejam acumulados por Estrutura Organizacional deverá ser customizado o Runtime [ACUMEORG](#).

Para que os lançamentos gerados sejam acumulados por Data de Pagamento deverá ser customizado o Runtime [ACUMDTPAG](#).

5 - Integração Contabilidade (I01)

Para que os lançamentos gerados sejam acumulados por Centro de Custo Contábil deverá ser customizado o Runtime [ACUMUCCCT](#).

Para que os Lançamentos Gerados sejam acumulados por Natureza do Custo deverá ser customizado o Runtime [ACUMUNATC](#).

Após o término do programa de criação do movimento será executado um procedimento que criará um arquivo formatado para carga na contabilidade. Este programa, no geral, é customizado para cada cliente.

O programa busca o centro de custo para ser gravado nas tabelas LANÇAMENTOS_GERADOS e abertos da seguinte forma:

Verifica se existe registro de centro de custo na tabela de CONTAS_CONTÁBEIS, se não houver registro nesta tabela, irá buscar o centro de custo conforme registro na tabela de LANÇAMENTOS_CONTÁBEIS (**Ccusto Cre** ou **Ccusto Deb**).

Em não existindo registro de centro de custo nas tabelas de CONTAS_CONTÁBEIS e LANÇAMENTOS_GERADOS irá verificar se existe rateio (registro de centro de custo na tabela de REG_MOVIMENTOS_CALCULADOS), se não houver registro irá buscar o centro de custo do funcionário.

Quebras:

- Empresa
- Dt-Lançamento
- Conta
- Centro de Custo
- Tipo Lançamento
- Unidade de Negócio
- Histórico Contábil

Obs.: Para fazer Restrição de Cálculo Contábil verificar as Variáveis Contábeis descritas na documentação do módulo de Procedimentos na transação Restrições de Cálculo ([GPR0090](#)).

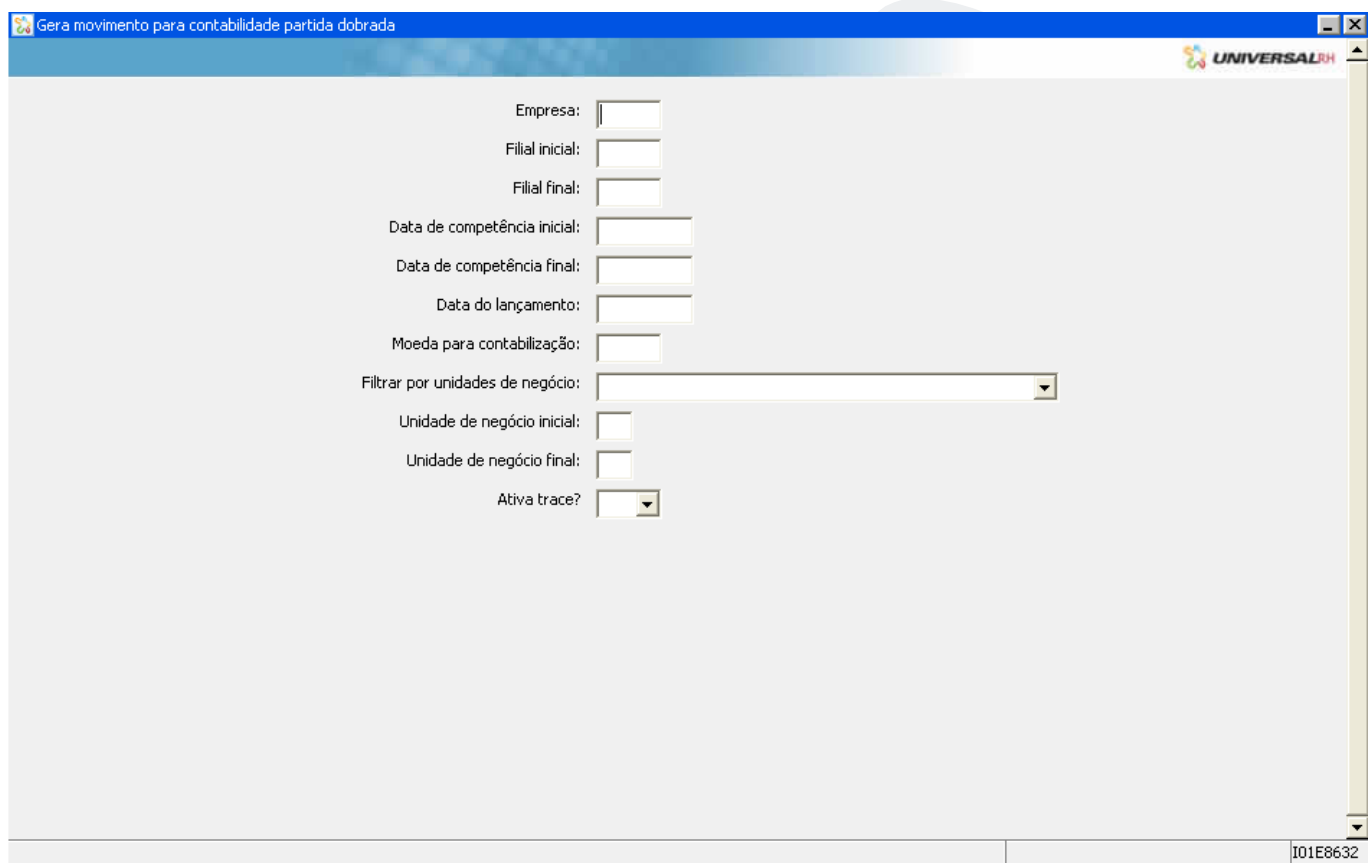
6 - Integração Contabilidade (I01)

I018632 – Gera movimento para contabilidade com partida dobrada

Código do Programa : I018632
Forms de Parâmetro : I01E8632
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Gera movimento para contabilidade com partida dobrada
Específico :

Objetivo do Programa:

O objetivo do programa é gerar um arquivo sequencial formatado para carga do movimento da folha na contabilidade.



Empresa:

Filial inicial:

Filial final:

Data de competência inicial:

Data de competência final:

Data do lançamento:

Moeda para contabilização:

Filtrar por unidades de negócio:

Unidade de negócio inicial:

Unidade de negócio final:

Ativa trace?

Descrição dos Campos de Parâmetros:

Competência inicial e final:

São campos de parâmetros destinados a receber os valores que determinam o intervalo de competências que deverão participar do processo. Serão selecionados os registros da tabela de movimentos calculados conforme intervalo das datas de competência.

Data do lançamento:

7 - Integração Contabilidade (I01)

Campo de parâmetro destinado a receber o valor da data de Lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas LANCAMENTOS_ABERTOS_DEB_CRE e LANCAMENTOS_GERADOS_DEB_CRE.

Filtrar por unidades de negócio:

Campo destinado a identificar a forma de seleção para unidade de negócio, tendo como opções:

- Incluídas no intervalo especificado
Seleciona os registros para integração para as unidades de negócio que satisfizerem o intervalo informado em **Unidade de negócio inicial:** e **Unidade de negócio final:**
- Sem unidade de negócio associada
Seleciona os registros onde a unidade de negócio não esteja associada à unidade organizacional.
- Todas
Seleciona todos os registros independente de terem ou não uma unidade de negócio associada à unidade organizacional.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

O programa é executado logo após a execução do programa I01P8600.

Para apresentação correta dos agrupamentos dos dados as filiais de débito e crédito devem ser iguais.

I018780 – Integra plano de contas da contabilidade

Código do Programa : I018780
Forms de Parâmetro :
Package : Não
Linguagem : PL/SQL
Título do Programa : Integra plano de contas da contabilidade
Específico :

Objetivo do Programa:

Criar uma visão (CONTAS_CONTABEIS) a partir do plano de contas da contabilidade.

Com isso, será padronizado o plano de contas existente na contabilidade para o sistema de RH.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

Este programa só deverá ser executado caso o cliente utilize o sistema de contabilidade fornecido pela empresa Domínio Informática, e ainda queira aproveitar o plano de contas deste sistema para o **UNIVERSALRH**.

Guia do Usuário

Procedurais Específicos

Contabilização Universal

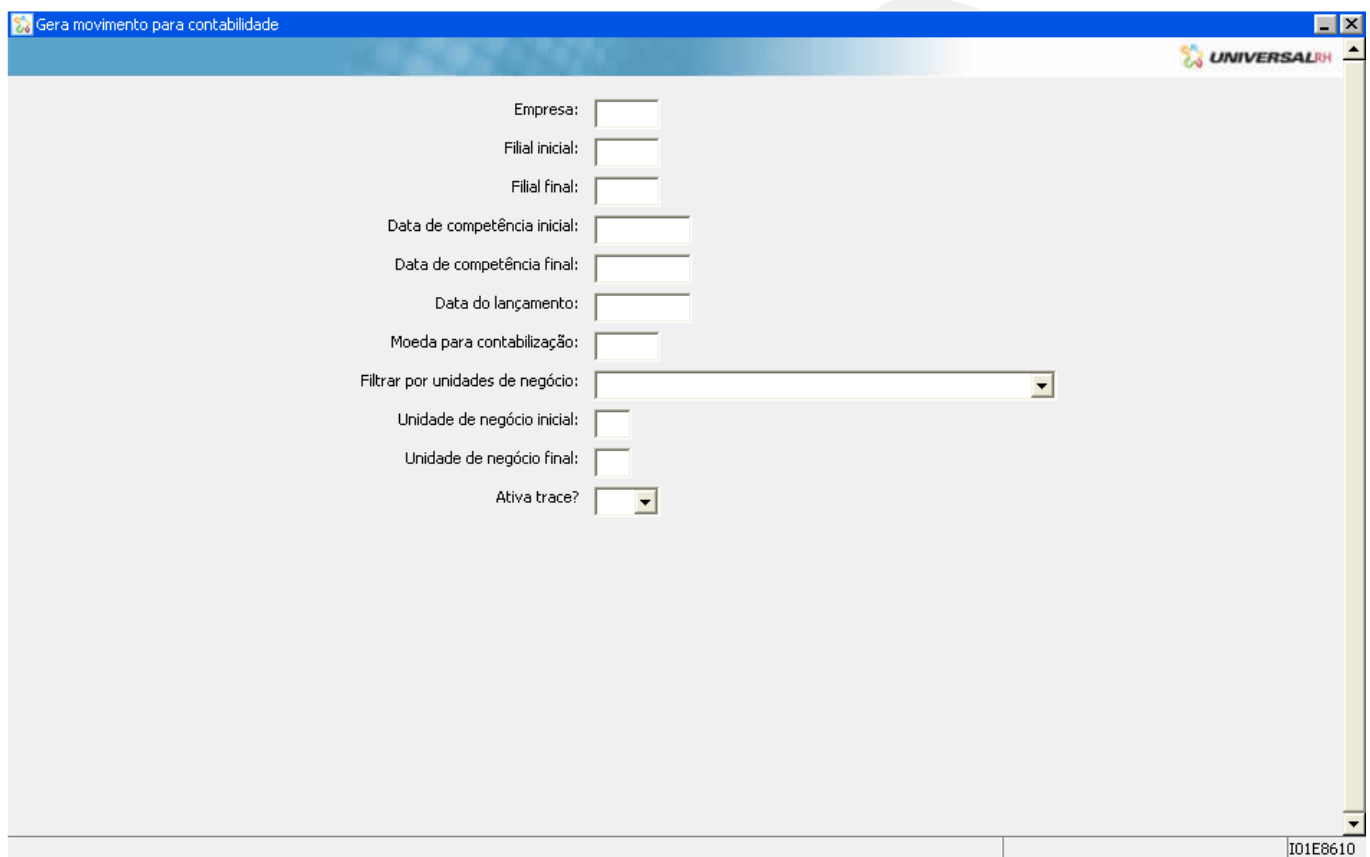
Versão 12.0 Release 27.02.2025

I018610 – Formatador de arquivo para a contabilidade (Albarus)

Código do Programa : I018610
Forms de Parâmetro : I01E8610
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Formatador de arquivo para a contabilidade
Específico : Albarus S/A

Objetivo do Programa:

O objetivo do programa é gerar um arquivo sequencial formatado para carga do movimento da folha na contabilidade. Sistema de Contabilidade Haiterion.



Empresa:

Filial inicial:

Filial final:

Data de competência inicial:

Data de competência final:

Data do lançamento:

Moeda para contabilização:

Filtrar por unidades de negócio:

Unidade de negócio inicial:

Unidade de negócio final:

Ativa trace?

I01E8610

Descrição dos Campos de Parâmetros:

Competência inicial e final:

São campos de parâmetros destinados a receber os valores que determinam o intervalo de competência que deverão participar do processo. Serão selecionados os registros da tabela de movimentos calculados conforme intervalo das datas de competência.

Data do lançamento:

11 - Integração Contabilidade (I01)

Campo de parâmetro destinado a receber o valor da data de lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas LANCAMENTOS_ABERTOS e LANCAMENTOS_GERADOS.

Filtrar por unidades de negócio:

Campo destinado a identificar a forma de seleção para unidade de negócio, tendo como opções:

- Incluídas no intervalo especificado
Seleciona os registros para integração para as unidades de negócio que satisfizerem o intervalo informado em **Unidade de negócio inicial:** e **Unidade de negócio final:**.
- Sem unidade de negócio associada
Seleciona os registros onde a unidade de negócio não esteja associada à unidade organizacional.
- Todas
Seleciona todos os registros independente de terem ou não uma unidade de negócio associada à unidade organizacional.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

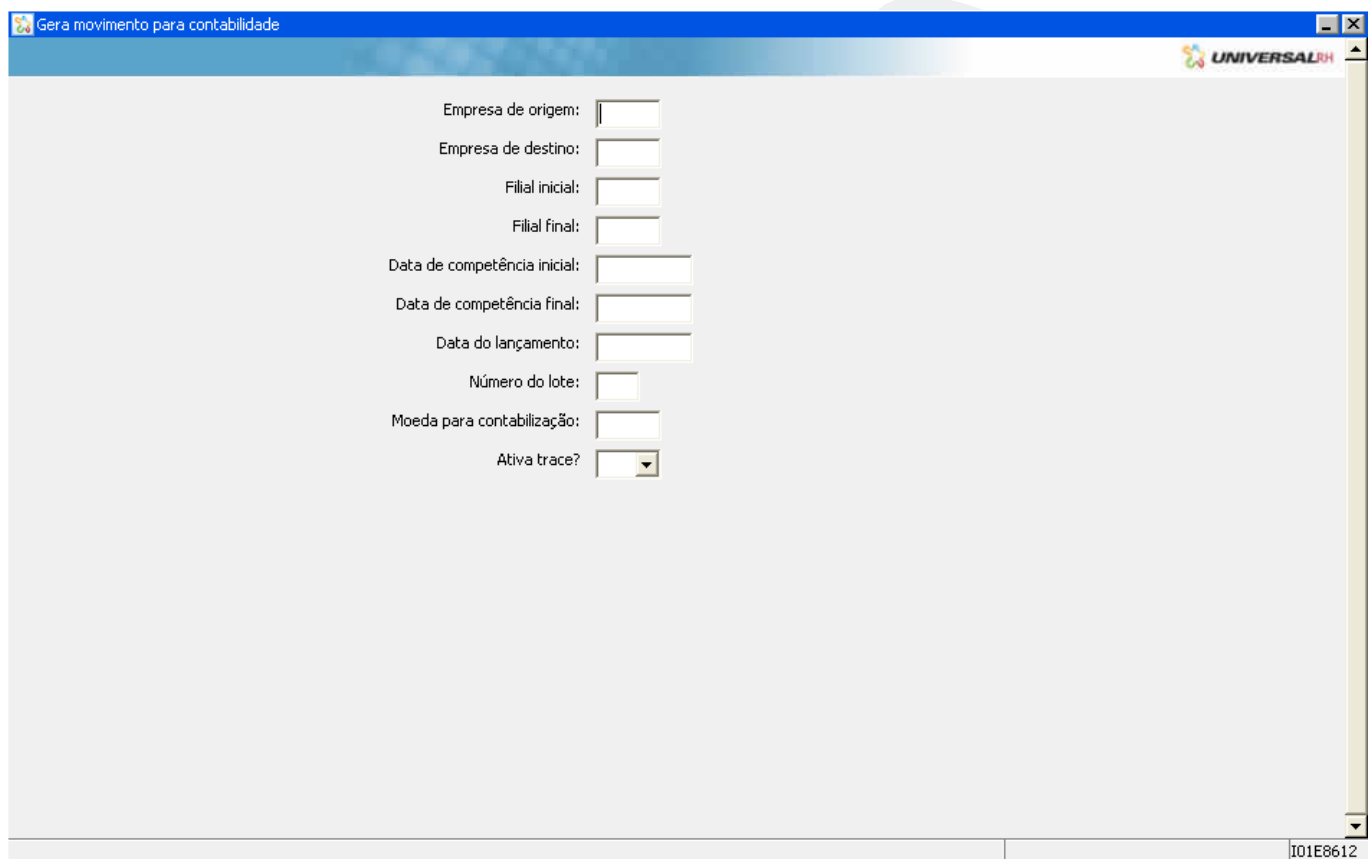
12 - Integração Contabilidade (I01)

I018612 – Formatador de arquivo para a contabilidade (Albarus-Cooperativa)

Código do Programa : I018612
Forms de Parâmetro : I01E8612
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Formatador de arquivo para a contabilidade (Albarus Cooperativa)
Específico : Albarus S/A

Objetivo do Programa:

O objetivo do programa é gerar um arquivo sequencial formatado para carga do movimento da folha na contabilidade da cooperativa.



Descrição dos Campos de Parâmetros:

Competência inicial e final:

São campos de parâmetros destinados a receber os valores que determinam o intervalo de competências que deverão participar do processo. Serão selecionados os registros da tabela de movimentos calculados conforme intervalo das datas de competência.

Data do lançamento:

13 - Integração Contabilidade (I01)

Campo de parâmetro destinado a receber o valor da data de lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas LANCAMENTOS_ABERTOS e LANCAMENTOS_GERADOS.

Número do lote:

Campo de parâmetro destinado a receber a informação do número do lote que deverá participar do processo. O número do lote será usado na gravação do arquivo formatador de saída.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

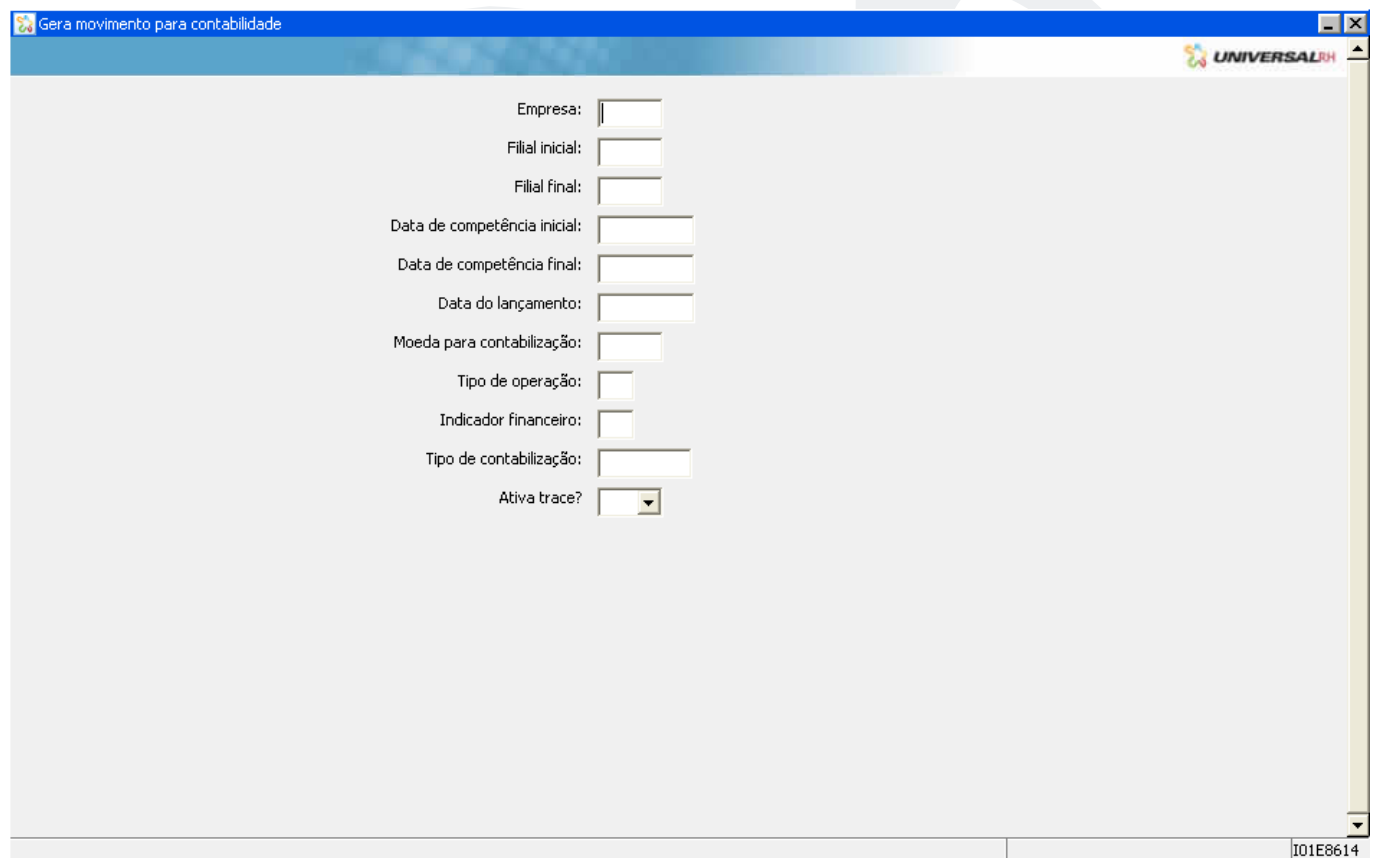
Este programa é executado logo após a execução do programa I01P8601.

I018614 – Formatador de arquivo para a contabilidade (Fockink)

Código do Programa : I018614
Forms de Parâmetro : I01E8614
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Formatador de arquivo para a contabilidade
Específico : Fockink Participações Ltda

Objetivo do Programa:

O objetivo do programa é de gerar a carga do movimento da Folha de Pagamento diretamente na tabela da contabilidade DOM_CONTA_CONTÁBIL (Sistema de Contabilidade da empresa Domínio), o campo TRANSACAO_CONTABIL da tabela DOM_CONTA_CONTÁBIL será preenchido pela sequência TRANSACAO_CTB (Sistema de Contabilidade da empresa Domínio).



Descrição dos Campos de Parâmetros:

Competência inicial e final:

São campos de parâmetros destinados a receber os valores que determinam o intervalo de competências que deverão participar do processo. Serão selecionados os registros da tabela de movimentos calculados conforme intervalo das datas de competência.

15 - Integração Contabilidade (I01)

Data de lançamento:

Campo de parâmetro destinado a receber o valor da data de lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas LANCAMENTOS_ABERTOS e LANCAMENTOS_GERADOS.

Tipo de operação:

Campo de parâmetro destinado a receber a informação do Tipo de Operação que deverá participar do processo. O Tipo de Operação informado será gravado na tabela DOM_LANCTO_CONTABIL.

Indicador financeiro:

Campo de parâmetro destinado a receber a informação do Indicador Financeiro que deverá participar do processo. O Indicador Financeiro informado será gravado na tabela DOM_LANCTO_CONTABIL.

Tipo de contabilização:

Campo de parâmetro destinado a receber a informação do Tipo de Contabilização que deverá participar do processo. O Tipo de Contabilização informado será gravado na tabela DOM_LANCTO_CONTABIL.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

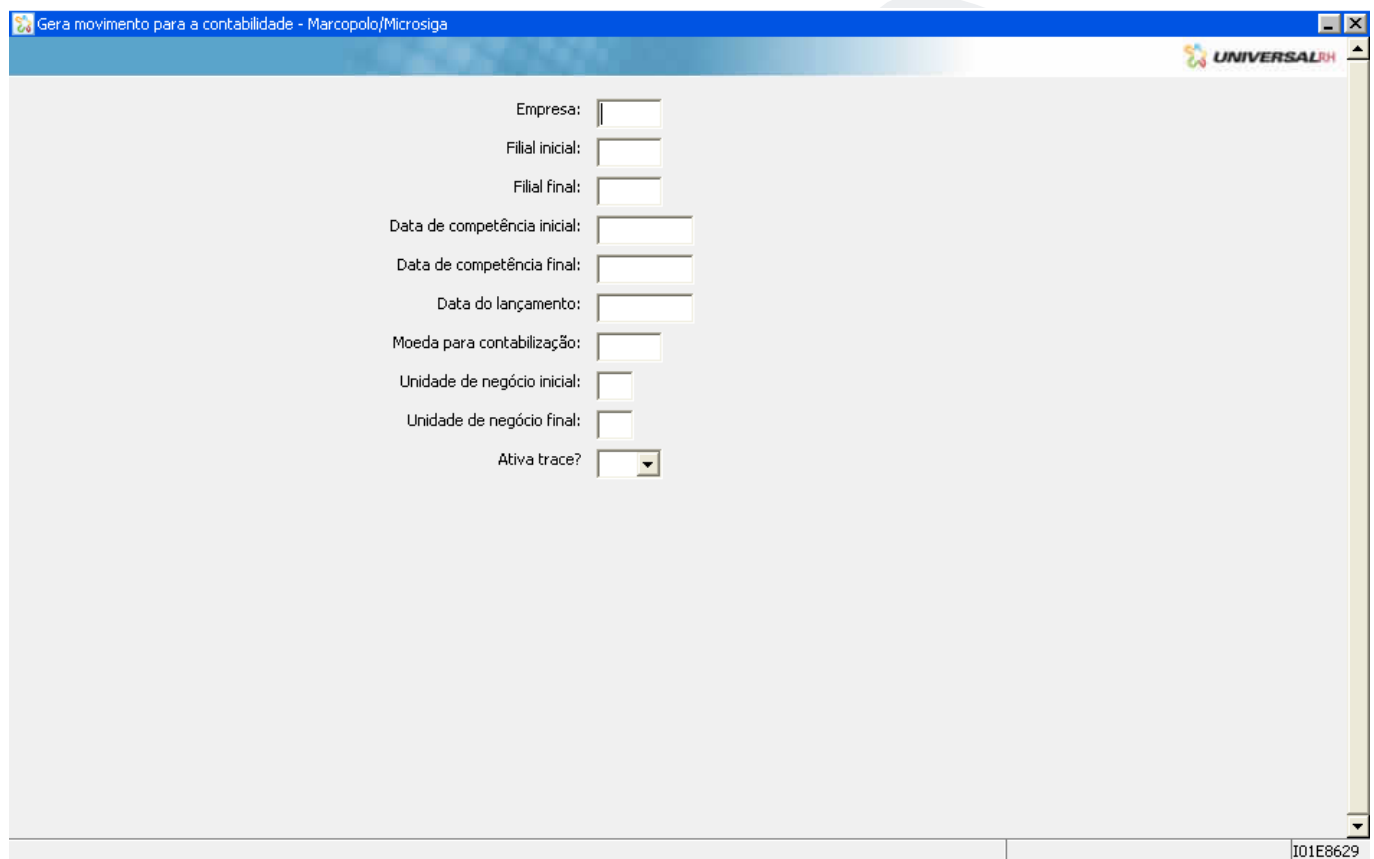
Este programa é executado logo após a execução do programa I01P8600.SQL.

I018629 – Formatador de arquivo para a contabilidade (Marcopolo / Microsiga)

Código do Programa : I018629
Forms de Parâmetro : I01E8629
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Formatador de arquivo para a contabilidade
Específico : Marcopolo S/A

Objetivo do Programa:

O objetivo do programa é gerar um arquivo sequencial formatado para carga do movimento da folha de pagamento. Sistema de Contabilidade Microsiga.



Descrição dos Campos de Parâmetros:

Competência inicial e final:

São campos de parâmetros destinados a receber os valores que determinam o intervalo de competências que deverão participar do processo. Serão selecionados os registros da tabela de movimentos calculados conforme intervalo das datas de competência.

Data do lançamento:

17 - Integração Contabilidade (I01)

Campo de parâmetro destinado a receber o valor da data de Lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas LANCAMENTOS_ABERTOS e LANCAMENTOS_GERADOS.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

Este programa é executado logo após a execução do programa I01P8600.SQL.

Empresa

Filial

Data de Lançamento

Centro de Custo

Unidade de Negócio

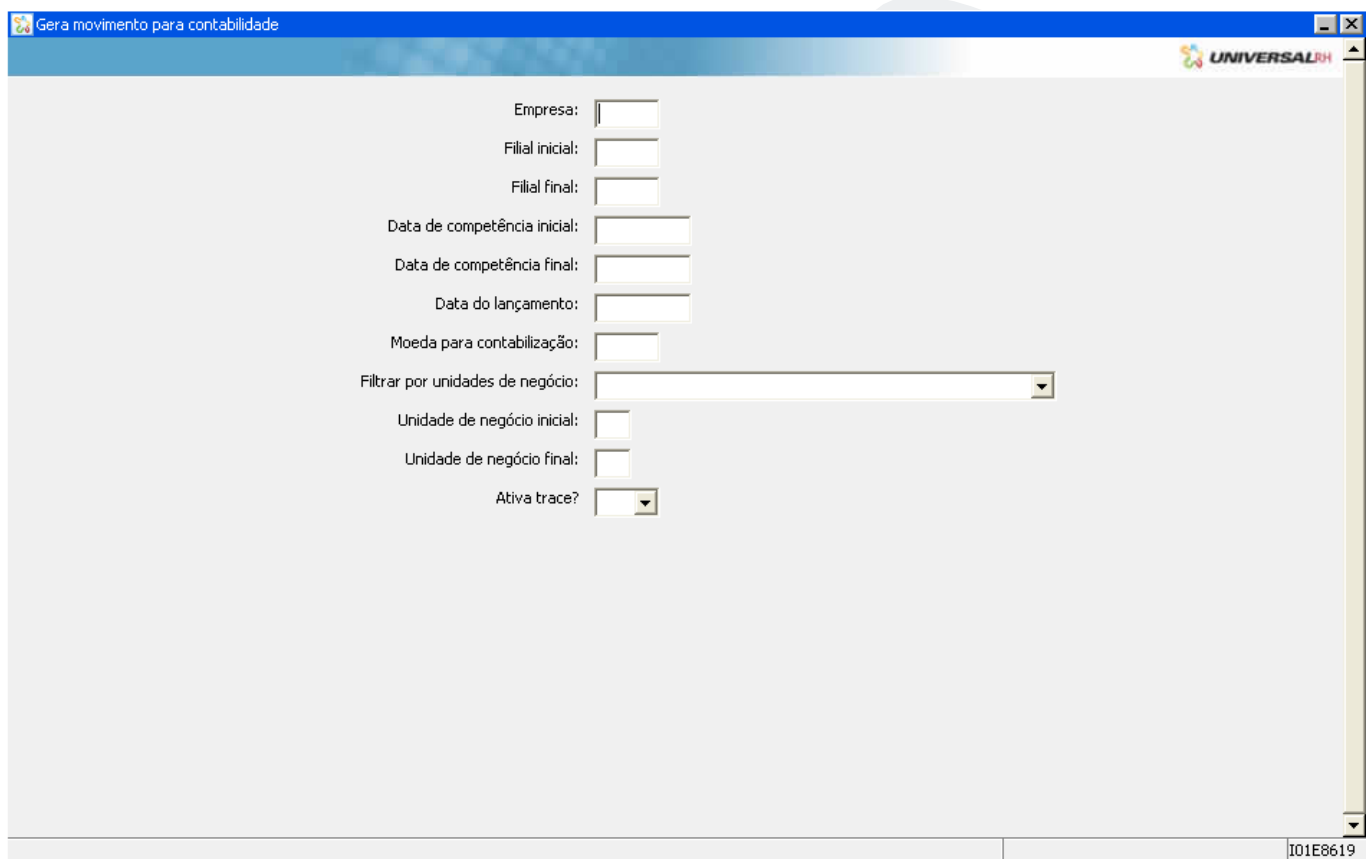
Código da Conta Contábil

I018619 – Formatador de arquivo para a contabilidade (Marcopolo)

Código do Programa : I018619
Forms de Parâmetro : I01E8619
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Formatador de arquivo para a contabilidade
Específico : Marcopolo S/A

Objetivo do Programa:

O objetivo do programa é gerar um arquivo sequencial formatado para carga do movimento da folha de pagamento. Sistema de Contabilidade Magnus.



Empresa:

Filial inicial:

Filial final:

Data de competência inicial:

Data de competência final:

Data do lançamento:

Moeda para contabilização:

Filtrar por unidades de negócio:

Unidade de negócio inicial:

Unidade de negócio final:

Ativa trace?

I01E8619

Descrição dos Campos de Parâmetros:

Competência inicial e final:

São campos de parâmetros destinados a receber os valores que determinam o intervalo de competências que deverão participar do processo. Serão selecionados os registros da tabela de movimentos calculados conforme intervalo das datas de competência.

Data do lançamento:

19 - Integração Contabilidade (I01)

Campo de parâmetro destinado a receber o valor da data de Lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas LANCAMENTOS_ABERTOS e LANCAMENTOS_GERADOS.

Filtrar por unidades de negócio:

Campo destinado a identificar a forma de seleção para unidade de negócio, tendo como opções:

- Incluídas no intervalo especificado
Seleciona os registros para integração para as unidades de negócio que satisfizerem o intervalo informado em **Unidade de negócio inicial:** e **Unidade de negócio final:**
- Sem unidade de negócio associada
Seleciona os registros onde a unidade de negócio não esteja associada à unidade organizacional.
- Todas
Seleciona todos os registros independente de terem ou não uma unidade de negócio associada à unidade organizacional.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

Este programa é executado logo após a execução do programa I01P8600.SQL.

Empresa
Filial
Data de Lançamento
Centro de Custo
Unidade de Negócio
Código da Conta Contábil

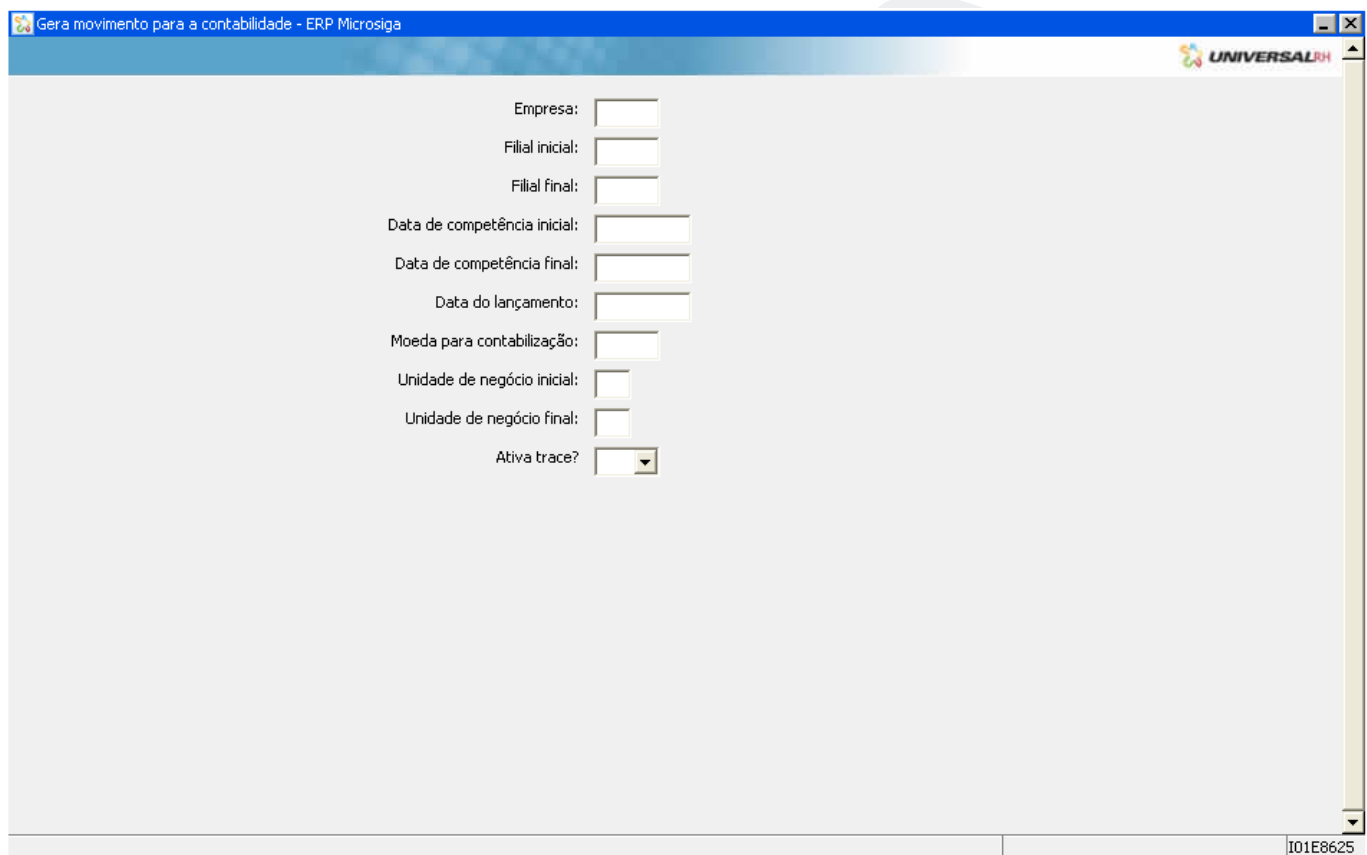
20 - Integração Contabilidade (I01)

I018625 – Formatador de arquivo para a contabilidade (Marcopolo)

Código do Programa : I018625
Forms de Parâmetro : I01E8625
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Formatador de arquivo para a contabilidade
Específico : Marcopolo S/A

Objetivo do Programa:

O objetivo do programa é gerar um arquivo sequencial formatado para carga do movimento da folha de pagamento. Sistema de Contabilidade Microsiga.



Descrição dos Campos de Parâmetros:

Competência inicial e final:

São Campos de parâmetros destinados a receber os valores que determinam o intervalo de competências que deverão participar do processo. Serão selecionados os registros da tabela de movimentos calculados conforme intervalo das datas de competência.

Data do lançamento:

21 - Integração Contabilidade (I01)

Campo de parâmetro destinado a receber o valor da data de Lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas LANCAMENTOS_ABERTOS e LANCAMENTOS_GERADOS.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

Este programa é executado logo após a execução do programa I01P8600.SQL.

Empresa

Filial

Data de Lançamento

Centro de Custo

Unidade de Negócio

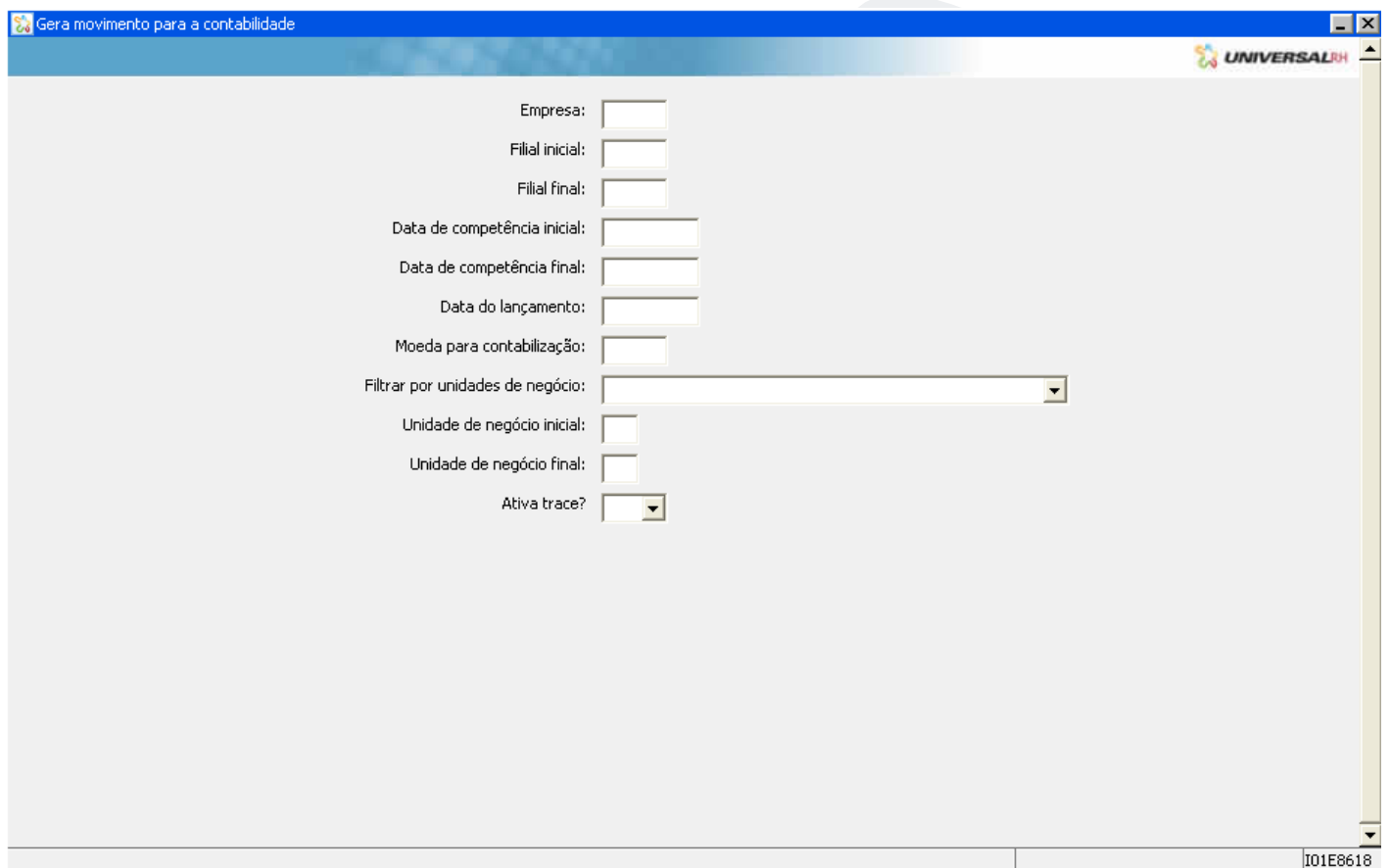
Código da Conta Contábil

I018618 – Formatador de arquivo para a contabilidade (SESI)

Código do Programa : I018618
Forms de Parâmetro : I01E8618
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Formatador de arquivo para a contabilidade
Específico : SESI

Objetivo do Programa:

O objetivo do programa é gerar um arquivo sequencial formatado para carga do movimento da folha de pagamento. Sistema de Contabilidade Direção.



Empresa:

Filial inicial:

Filial final:

Data de competência inicial:

Data de competência final:

Data do lançamento:

Moeda para contabilização:

Filtrar por unidades de negócio:

Unidade de negócio inicial:

Unidade de negócio final:

Ativa trace?

Descrição dos Campos de Parâmetros:

Competência inicial e final:

São campos de parâmetros destinados a receber os valores que determinam o intervalo de competência que deverão participar do processo. Serão selecionados os registros da tabela de movimentos calculados conforme intervalo das datas de competência.

Data do lançamento:

23 – Integração Contabilidade (101)

Campo de parâmetro destinado a receber o valor da data de lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas LANCAMENTOS_ABERTOS e LANCAMENTOS_GERADOS.

Filtrar por unidades de negócio:

Campo destinado a identificar a forma de seleção para unidade de negócio, tendo como opções:

- Incluídas no intervalo especificado
Selecione os registros para integração para as unidades de negócio que satisfizerem o intervalo informado em **Unidade de negócio inicial:** e **Unidade de negócio final:**.
- Sem unidade de negócio associada
Selecione os registros onde a unidade de negócio não esteja associada à unidade organizacional.
- Todas
Selecione todos os registros independente de terem ou não uma unidade de negócio associada à unidade organizacional.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

Este programa é executado logo após a execução do programa I01P8600.SQL.

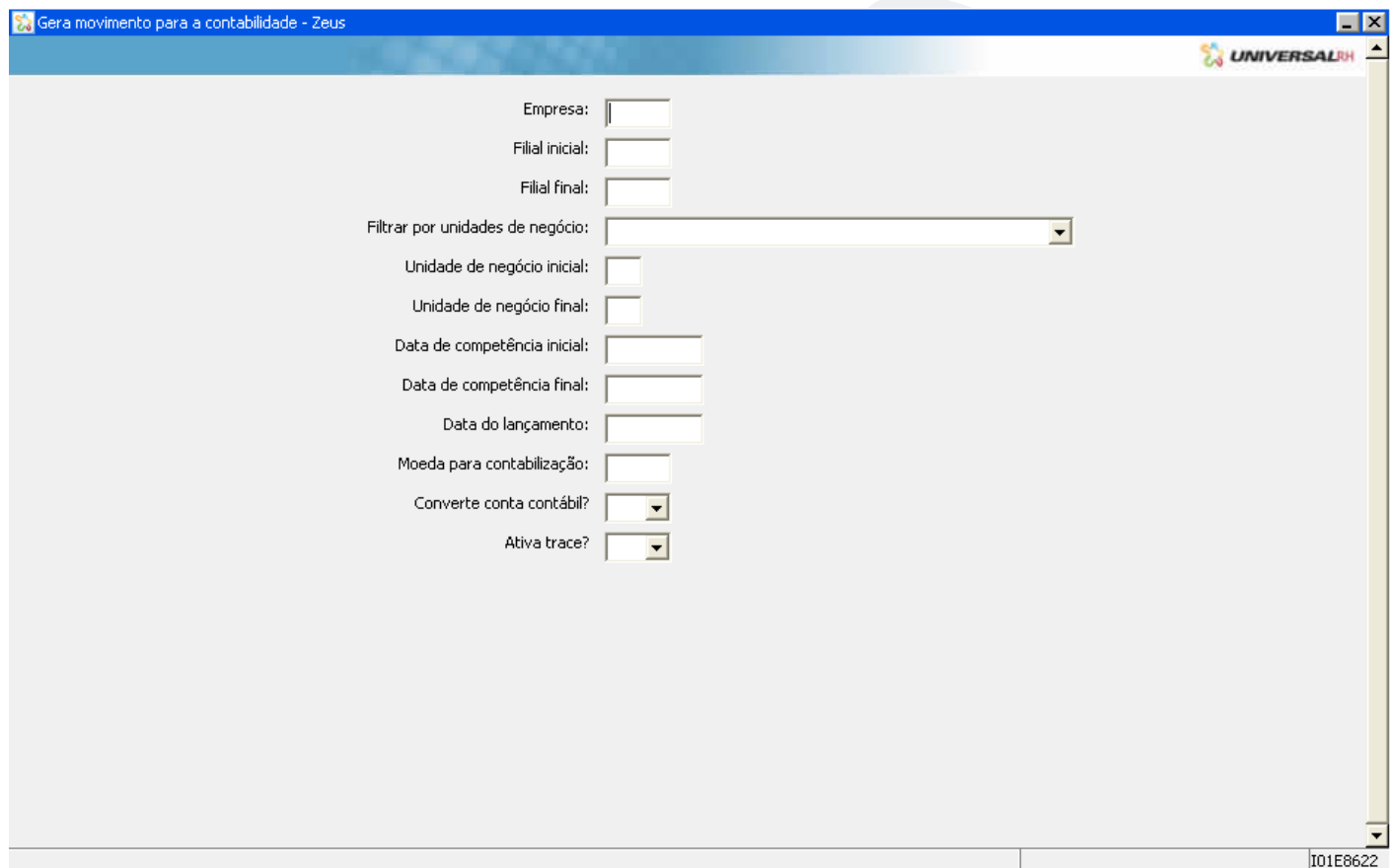
Empresa
Filial
Data de Lançamento
Centro de Custo
Unidade de Negócio
Código da Conta Contábil

I018622 – Formatador de arquivo para a contabilidade (SESI)

Código do Programa : I018622
Forms de Parâmetro : I01E8622
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Formatador de arquivo para a contabilidade
Específico : SESI

Objetivo do Programa:

O objetivo do programa é gerar um arquivo sequencial formatado para carga do movimento da folha de pagamento. Interface Contábil Zeus.



Descrição dos Campos de Parâmetros:

Competência inicial e final:

São campos de parâmetros destinados a receber os valores que determinam o intervalo de competências que deverão participar do processo. Serão selecionados os registros da tabela de movimentos calculados conforme intervalo das datas de competência.

Filtrar por unidades de negócio:

25 - Integração Contabilidade (I01)

Campo destinado a identificar a forma de seleção para unidade de negócio, tendo como opções:

- Incluídas no intervalo especificado
Seleciona os registros para integração para as unidades de negócio que satisfizerem o intervalo informado em **Unidade de negócio inicial:** e **Unidade de negócio final:**.
- Sem unidade de negócio associada
Seleciona os registros onde a unidade de negócio não esteja associada à unidade organizacional.
- Todas
Seleciona todos os registros independente de terem ou não uma unidade de negócio associada à unidade organizacional.

Data do lançamento:

Campo de parâmetro destinado a receber o valor da data de Lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas LANCAMENTOS_ABERTOS e LANCAMENTOS_GERADOS.

Converte conta contábil?

Campo de parâmetro destinado a receber o valor SIM ou NÃO para substituir a Conta Contábil pela conta Contábil do Projeto da tabela BOLETIM_ALOCACAO_COLABORADORES que deverá participar do processo.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

Este programa é executado logo após a execução do programa I01P8600.

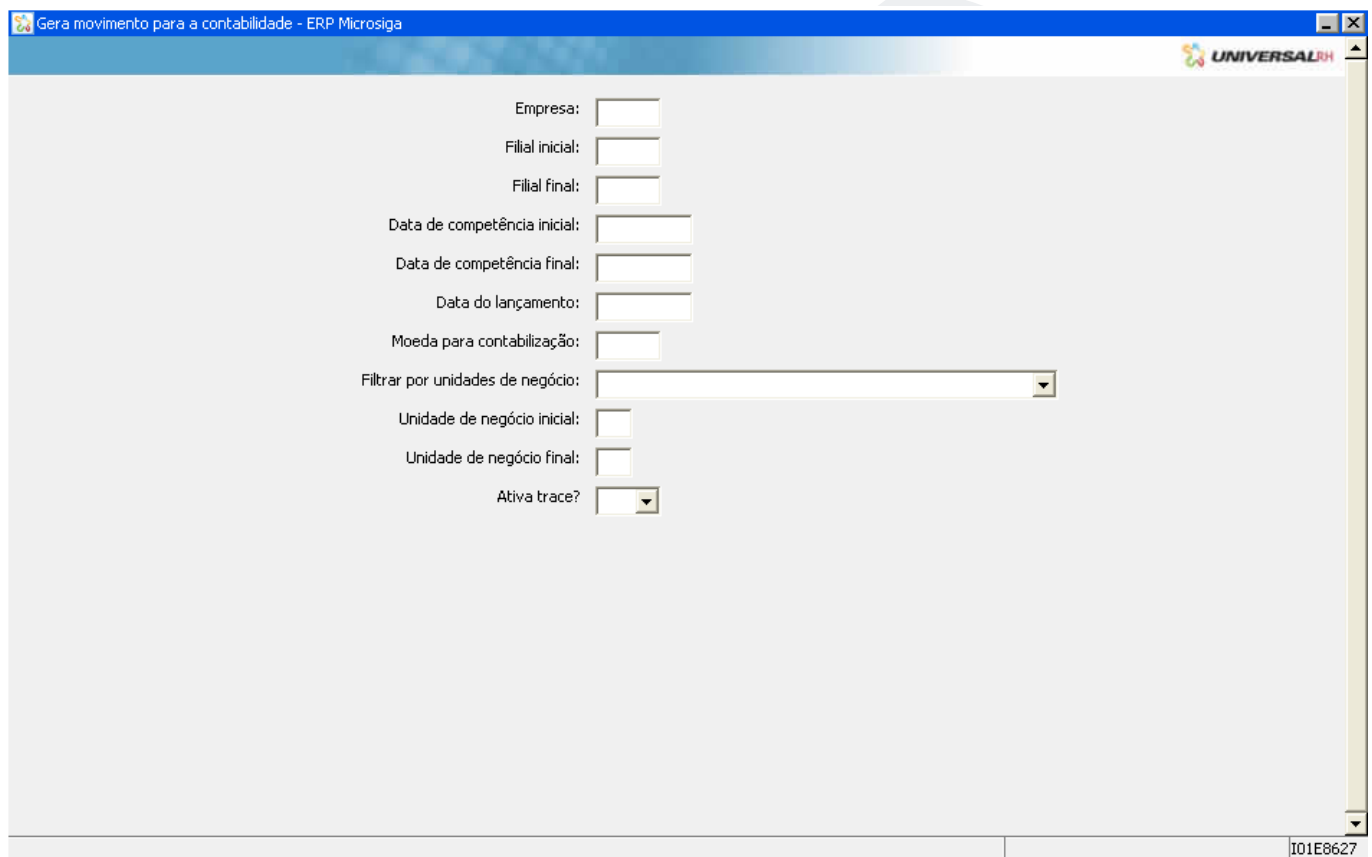
Os campos deste arquivo estão separados por tabulação (TAB).

I018627 – Formatador de arquivo para a contabilidade ERP – SAP (Marcopolo)

Código do Programa : I018627
Forms de Parâmetro : I01E8627
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Formatador de arquivo para a contabilidade – ERP SAP
Específico : Marcopolo

Objetivo do Programa:

O objetivo do programa é gerar um arquivo sequencial formatado para carga do movimento da folha de pagamento. Interface Contábil ERP - SAP.



Descrição dos Campos de Parâmetros:

Competência inicial e final:

São campos de parâmetros destinados a receber os valores que determinam o intervalo de competência que deverão participar do processo. Serão selecionados os registros da tabela de movimentos calculados conforme intervalo das datas de competência.

Data do lançamento:

27 – Integração Contabilidade (I01)

Campo de parâmetro destinado a receber o valor da data de lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas LANCAMENTOS_ABERTOS e LANCAMENTOS_GERADOS.

Filtrar por unidades de negócio:

Campo destinado a identificar a forma de seleção para unidade de negócio, tendo como opções:

- Incluídas no intervalo especificado
Seleciona os registros para integração para as unidades de negócio que satisfizerem o intervalo informado em **Unidade de negócio inicial:** e **Unidade de negócio final:**.
- Sem unidade de negócio associada
Seleciona os registros onde a unidade de negócio não esteja associada à unidade organizacional.
- Todas
Seleciona todos os registros independente de terem ou não uma unidade de negócio associada à unidade organizacional.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

Customizar os Códigos Internos:

Para a tabela TIPOS_DE_PARA:

- 3** - Identifica o código da empresa

Será utilizado o campo CHARACTER_PARA da tabela REGISTROS_DE_PARA para gerar a informação da empresa conforme cadastro do sistema de contabilidade SAP.

Critério de Ordenação, Quebras, outras particularidades:

Este programa é executado logo após a execução do programa I01P8600.

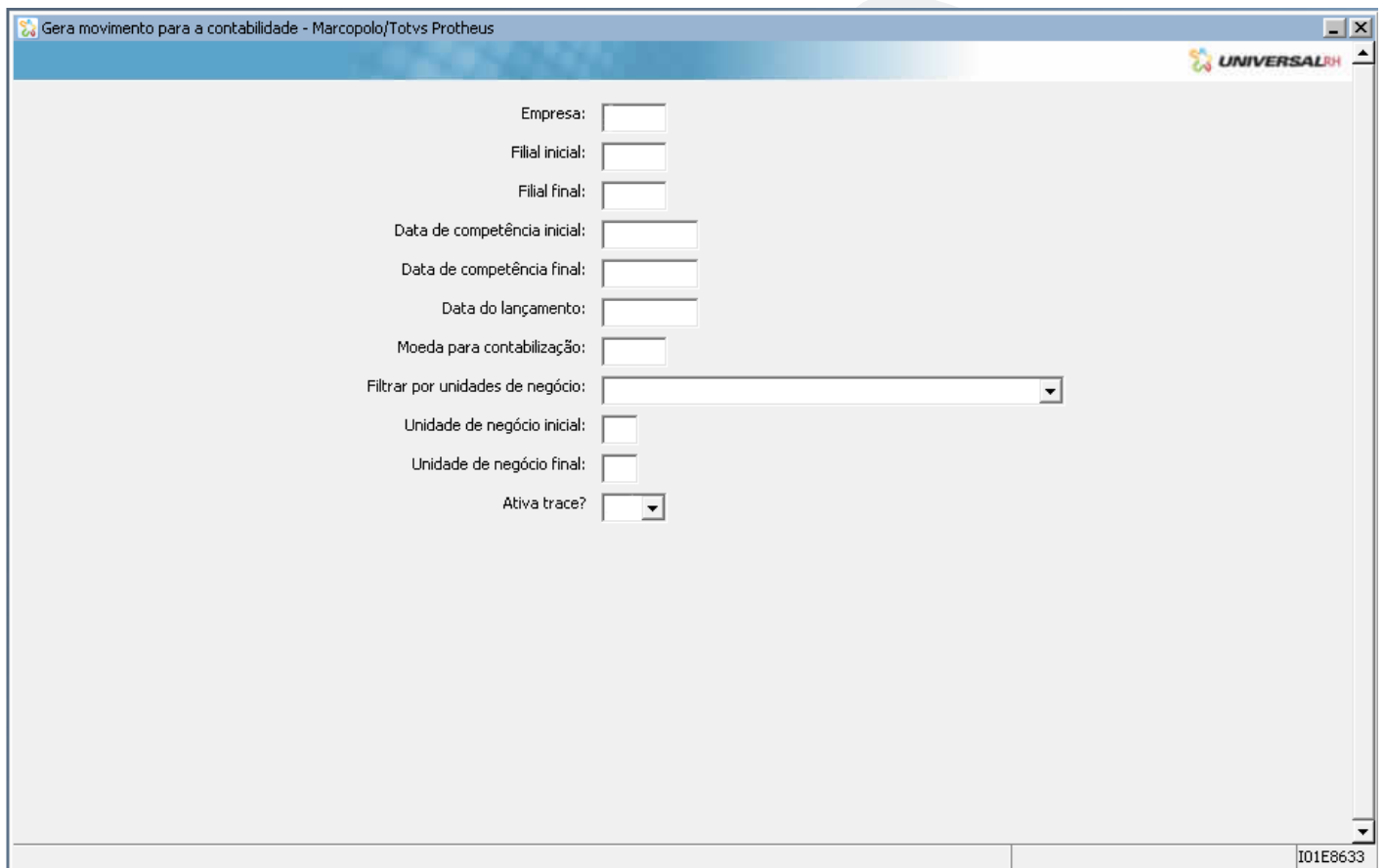
Os campos deste arquivo estão separados por tabulação ponto e vírgula “;”.

I018633 – Formatador de arquivo para a contabilidade (TOTVS Protheus 12.1.2310)

Código do Programa : I018633
Forms de Parâmetro : I01E8633
Package : I01P8600
Linguagem : PL/SQL
Título do Programa : Formatador de arquivo para a contabilidade
Específico : Marcopolo S/A

Objetivo do Programa:

O objetivo do programa é gerar um arquivo sequencial formatado para carga do movimento da folha de pagamento. Sistema de Contabilidade Microsiga.



Gera movimento para a contabilidade - Marcopolo/Totvs Protheus

UNIVERSALRH

Empresa:

Filial inicial:

Filial final:

Data de competência inicial:

Data de competência final:

Data do lançamento:

Moeda para contabilização:

Filtrar por unidades de negócio:

Unidade de negócio inicial:

Unidade de negócio final:

Ativa trace?

I01E8633

Descrição dos Campos de Parâmetros:

Competência inicial e final:

São campos de parâmetros destinados a receber os valores que determinam o intervalo de competências que deverão participar do processo. Serão selecionados os registros da tabela de movimentos calculados conforme intervalo das datas de competência.

Data do lançamento:

29 - Integração Contabilidade (I01)

Campo de parâmetro destinado a receber o valor da data de Lançamento que deverá participar do processo. Esta data será usada na gravação das tabelas LANCAMENTOS_ABERTOS e LANCAMENTOS_GERADOS.

Obs.: Para visualização das descrições dos Campos de Parâmetros Genéricos consultar o arquivo de documentação Generalidades.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

Este programa é executado logo após a execução do programa I01P8600.SQL.

Empresa

Filial

Data de Lançamento

Centro de Custo

Unidade de Negócio

Código da Conta Contábil

I01P8601 – Gera movimento para contabilidade (Albarus-Cooperativa)

Código do Programa : I01P8601
Forms de Parâmetro :
Package : I01P8601
Linguagem : PL/SQL
Título do Programa : Gera movimento para contabilidade (Albarus - Cooperativa)
Específico : Albarus S/A

Objetivo do Programa:

O objetivo do programa é gerar a contabilização da Folha de Pagamento para cooperativa.

Complementações Necessárias:(O usuário deverá implementar)

O usuário deverá implementar na folha, o cadastramento de:

- Descrição de Históricos;
- Descrição de Contas Contábeis;
- Lançamentos Contábeis (esta transação tem por objetivo criar as regras de contabilização para qualquer evento que se julgar importante contabilizar).

Obs.: Estas transações estão no menu Parâmetros para Contabilização.

Critério de Ordenação, Quebras, outras particularidades:

O critério de aglutinação será definido pelo usuário, usando principalmente, a transação Lançamento Contábil. Logo, este procedural apenas lê o que estiver cadastrado no Lançamento Contábil, para criar o movimento para contabilidade.

Após o término do programa de criação do movimento será executado um procedimento que criará um arquivo formatado para carga na contabilidade. Este programa, no geral, é customizado para cada cliente.

Quebras:

Empresa
Data de Lançamento
Conta
Centro de Custo
Tipo Lançamento
Unidade de Negócio
Histórico Contábil.

Guia do Usuário

Triggers

Contabilização Universal

Versão 12.0 Release 28.01.2025

32 - Integração Contabilidade (I01)

I01T0010 – Gerar identificador de PK

Código do Programa : I01T0010
Linguagem : PL/SQL
Título do Programa : Gerar identificador de PK
Tabela : LANCAMENTOS_GERADOS

Objetivo do Programa:

Gerar o número identificar único (GUID) para a PK

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

33 - Integração Contabilidade (I01)

I01T0020 – Gerar identificador de PK

Código do Programa : I01T0020
Linguagem : PL/SQL
Título do Programa : Gerar identificador de PK
Tabela : LANCAMENTOS_GERADOS_DEB_CRE

Objetivo do Programa:

Gerar o número identificar único (GUID) para a PK

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:



Guia do Usuário

Anexos

Contabilização Universal

Versão 12.0 Release 19.05.2015

1. Instalação do UNIVERSALRH

Antes de iniciar a instalação do produto, recomendamos a leitura deste manual, pois nele estão contidas todas as orientações para instalar e dar início do seu funcionamento. Entretanto, devido ao grande número de itens opcionais que impactam na forma de utilização do **UNIVERSALRH**, achamos aconselhável sua leitura bem como um pequeno planejamento. Isso poderá evitar que haja necessidade de reinstalação.

1.1. Base Tecnológica

Para o funcionamento do **UNIVERSALRH** são utilizados vários produtos de diferentes fabricantes, tais como Sistemas Operacionais, Gerenciadores de Rede, Gerenciadores de Banco de Dados, etc. Esses produtos formam a base tecnológica em que o produto está fundamentado, e como tal, possui uma interdependência com eles. Deste modo, a garantia de funcionamento do **UNIVERSALRH** está limitada a permanência de funcionamento da base tecnológica. O mau funcionamento do **UNIVERSALRH** causado por qualquer dos produtos que formam a base tecnológica, será de inteira responsabilidade do fabricante daquele produto. Salientamos que, o fato do serviço de suporte do **UNIVERSALRH** desconhecer qualquer dos problemas que por ventura sejam detectados pelo usuário durante a utilização, motivados pelo fato descrito anteriormente, não caracteriza omissão e nem gera qualquer compromisso de correção.

1.2. Requerimentos de Software

O **UNIVERSALRH** está certificado para ser executado nas seguintes versões dos *software* da Oracle:

1. Oracle *WebLogic Server* 11g R1 (**10.3.5** e **10.3.6**);
2. Oracle *Forms and Reports* 11g R2 (**11.1.2.2.0**);
3. Oracle *Application Server 10g Release 2 (10.1.2.3.0) Forms and Reports Services*, como servidor de aplicação. Recomendado aplicar os seguintes patches sobre a instalação *baseline* (10.1.2.0), para estabilização dos serviços oferecidos pelo servidor de aplicação da Oracle, a saber (recomendamos a leitura do arquivo *README.txt* de cada um dos *patches* relacionados):

Ordem de aplicação do patch	Número do patch
1	5983622 (Patchset – 10.1.2.3.0)
2	6640838
3	6880880
4	7123031
5	9593176
6	11710576
7	14825718

4. Uma das seguintes versões do banco de dados Oracle: 10g R1, 10g R2, 11g R1 e 11g R2 e 12c;
5. Sistema Operacional *Windows* XP, 2003, 2008, Vista, 7 e 8.1, para execução da camada de apresentação do **UNIVERSALRH**;
6. Sistema Operacional Mac OS X, para execução da camada de apresentação do **UNIVERSALRH**;
7. Sistema Operacional Linux distribuição *Red Hat* 6.2 ou superior. Para maiores informações sobre produtos Oracle certificados em plataforma Linux, consulte:
 - <http://www.redhat.com/about/news/archive/2011/8/red-hat-submits-oracle-11gr2-on-red-hat-enterprise-linux-6-certification-test-results-to-oracle> (acesso em agosto de 2015).
 - <http://www.oracle.com/us/technologies/linux/overview/index.html> (acesso em agosto de 2015).
8. *Java Runtime Environment* (JRE) 1.6 32-bit, ou superior, para qualquer um dos sistemas operacionais disponíveis no mercado, suportados pelo JRE.
9. Pré-compilador Oracle PRO*C/C++ versão 9.2 ou superior;
10. Compiladores C/C++: *MS-Visual C/C++* 5, *MS-Visual Studio* 2010 (32/64Bits), CC ou XLC (AIX,SOLARIS, HP-UX) e GCC (Linux).
11. Navegadores de *Internet*:
 - a. *Windows Internet Explorer* 8 e 11;
 - b. *Mozilla Fire Fox* 33 ou superior.

1.3. Requerimentos de Hardware para Estações de Trabalho

Para a execução do sistema **UNIVERSALRH**, são exigidas as seguintes configurações mínimas de hardware:

1. Processador 1.4 Ghz;
2. Sistema Operacional *Windows* XP/2003/Vista/7/8, Mac OS X, ou as distribuições Linux Desktop.

As configurações acima podem variar dependendo do sistema operacional usado na estação de trabalho. Recomendamos consultar os sites:

Para informações sobre o **Oracle Application Server 10g Release 2**, acesse:

- <http://otn.oracle.com> (acesso em agosto de 2015)
- <http://www.oracle.com/technetwork/middleware/ias/downloads/as-certification-r2-101202-095871.html#BABGCBHA> (acesso em agosto de 2015)

Para obter maiores informações sobre os requisitos mínimos de hardware exigido para execução de aplicações em **Oracle Forms** e **Oracle Reports**.

Para informações sobre o **Oracle Forms Services 11g Release 2**, acesse:

- <http://www.oracle.com/technetwork/developer-tools/forms/overview/index.html?ssSourceSiteId=ocombr> (acesso em agosto de 2015).

1.4. Requerimentos de Hardware para o Servidor de Aplicação

O dimensionamento do hardware do servidor de aplicação deve ser feito conforme documentos disponíveis nos seguintes endereços eletrônicos:

- <http://www.oracle.com/technetwork/middleware/ias/downloads/as-certification-r2-101202-095871.html> (site com informações variadas sobre o **Oracle Forms 10g Release 2** - acesso em agosto de 2015);
 - <http://www.oracle.com/technetwork/developer-tools/forms/documentation/techlisting10gr2-093389.html> (documentos com informações para dimensionar o hardware do servidor de aplicação, para executar o **Oracle Application Server Forms Services 10g** – acesso em agosto de 2015);
- http://docs.oracle.com/cd/E48391_01/index.htm (índice da documentação do **Oracle Fusion Middleware – Forms and Reports 11g** – acesso em agosto de 2015);
- <http://www.oracle.com/technetwork/developer-tools/forms/webutil-090641.html> (índice da biblioteca **Oracle WebUtil** – acesso em agosto de 2015), e;
- http://docs.oracle.com/cd/E27559_01/doc.1112/e28552/toc.htm (**Fusion Middleware Performance and Tuning Guide** – acesso em agosto de 2015).

2. Ordem de Instalação

Deverá ser obedecida a ordem abaixo descrita, para os produtos adquiridos, para a execução dos procedimentos de instalação:

Administração de Recursos Humanos	(ARH) ¹
Administração de Pessoal.....	(GFP)
Integração da Contabilidade UNIVERSALRH	(I01)
Integração Contas a Pagar - Oracle Applications R11 .	(I03)
Integração da Contabilidade IFS	(I07)
Procedimentos	(GPR)
Recrutamento e Seleção	(GRS)
Cargos e Salários.....	(GCS)
Treinamento	(GTR)
Convênio	(GCV)
Benefício	(GBE)
Segurança do Trabalho.....	(GST)
Medicina do Trabalho.....	(GMT)
Avaliação de Pessoal	(GAP)
Controle de Custos.....	(GCC)
Processos Trabalhistas	(GPT)
eSocial	(GES)
Específico SESI.....	(E01)
Multifunções	(E03)
Aerus.....	(E05)
Específico Wal-Mart	(E07)
Universal Business Interface	(UBI)
Orientador	(ORI)

Observação:

O produto de Administração de Recursos Humanos (Gerenciador de Componentes), deve ser instalado em qualquer circunstância, pois ele é o módulo responsável por registrar informações sobre a instalação do sistema.

¹ Gerenciador de Componentes

3. Servidor de Aplicação

3.1. Instalando *Oracle Application Server 10g Release 2 (10.1.2.3.0)* e *Oracle Fusion Middleware 11g Release 2 (11.1.2.0)*

Antes de instalar o **UNIVERSALRH**, o administrador de rede e/ou administrador de banco de dados deverá instalar e configurar o servidor de aplicação da Oracle. Esta instalação deve ser executada com base no guia de instalação e configuração, que está disponível na documentação do *software*. Além disso, informações adicionais de instalação e configuração do **Oracle Forms Services** estão disponíveis em:

- <http://www.oracle.com/technetwork/developer-tools/forms/documentation/techlisting10gr2-093389.html> (documentação sobre o *OAS 10g Forms and Reports Services* – acesso em abril de 2013);
- <http://www.oracle.com/technetwork/middleware/reports/documentation/index.html> (documentação do *Oracle Reports* – acesso em abril de 2013);
- http://docs.oracle.com/cd/E24269_01/index.htm (documentação de instalação do *Oracle Fusion Middleware 11g Release 2 (11.1.2.0) - Forms and Reports Services* – acesso em agosto de 2015), e;
- <http://www.oracle.com/technetwork/developer-tools/forms/overview/index.html> (documentação completa do *Oracle Application Server 10g Release 2* e *Oracle Forms Services 11g* - acesso em agosto de 2015).

Para instalação e configuração do *Oracle Fusion Middleware 11g Release 2 Forms and Reports Services*, recomendamos a leitura do manual de instalação, pois é necessário que sejam entendidos os conceitos que regem a instalação e configuração do *WebLogic* e do *Forms and Reports Services 11g Release 2*. Para tanto, recomendamos a leitura do seguinte manual:

- https://docs.oracle.com/cd/E29542_01/core.11111/e10103/terminology.htm#ASCON117 (manual com explicações de conceitos chave para instalação do *WebLogic* – acesso em agosto de 2015), e;
- http://docs.oracle.com/cd/E24269_01/doc.11120/e23960/toc.htm (guia de instalação do *Oracle Forms and Reports* - acesso em agosto de 2015);
- http://docs.oracle.com/cd/E48391_01/doc.11120/e24477/toc.htm (guia de publicação de aplicações *Oracle Forms* – acesso em agosto de 2015);
- http://docs.oracle.com/cd/E48391_01/doc.11120/e24477/appc.htm#FSDEP457 (documento que relaciona a localização e exemplos dos arquivos de configuração – acesso em agosto de 2015).

3.2. Configurando da biblioteca *Oracle Webutil* no *Oracle Application Server 10g Release 2* e *Oracle Forms and Reports Services 11g Release 2*

É necessário instalar a biblioteca *Oracle Webutil* no servidor de aplicação para que algumas funções do sistema funcionem corretamente. Para habilitar o *Oracle Webutil*, o administrador do servidor de aplicação deverá seguir as orientações para o respectivo servidor de aplicação em uso, conforme relacionado a seguir.

3.2.1. Oracle Application Server 10g R2

Para o OAS 10g R2, o documento “*How To Install Webutil on 10.1.2.x Oracle Application Server*” explica como configurar o servidor de aplicação para utilizar a biblioteca *WebUtil*. O documento está disponível no site do suporte Oracle, com identificação **566628.1**, e também está disponível no site de atualizações do **UNIVERSALRH**, em http://suporte.universalrh.com.br/atualizacoes/ORACLE/DOCUMENTOS/OAS/HOW_TO_INSTALL_WEBUTIL_566628.1.PDF.

Também está disponível outro documento da Oracle que descreve as funções do *Oracle Webutil*, além de como configurar a biblioteca no servidor de aplicação da Oracle. Esse documento está disponível em: http://suporte.universalrh.com.br/atualizacoes/ORACLE/DOCUMENTOS/OAS/WEB_UTIL_USERS_GUIDE.PDF.

Este documento pode ser usado de forma complementar ao documento anterior, uma vez que este documento apresenta informações detalhadas sobre a biblioteca *Oracle Webutil*.

Os arquivos de instalação do *Oracle Webutil*, para o OAS 10g R2, estão disponíveis para [download](http://suporte.universalrh.com.br/atualizacoes/ORACLE/DOCUMENTOS/OAS/WEB_UTIL.ZIP) em: http://suporte.universalrh.com.br/atualizacoes/ORACLE/DOCUMENTOS/OAS/WEB_UTIL.ZIP.

3.2.2. Oracle Forms and Reports Services 11g R2

Está disponível no site de atualizações do **UNIVERSALRH** um documento que explica, passo-a-passo, como configurar a biblioteca *WebUtil* no *Oracle Forms Services 11g R2*. O documento pode ser acessado em: http://suporte.universalrh.com.br/atualizacoes/ORACLE/DOCUMENTOS/WEBLOGIC/CONFIGURE_WEBUTIL_IN_FORMS_11G_RELEASE_2_ORACLE_COMMUNITY.PDF.

Ainda, o documento “*How to Configure Webutil in Forms 11g*” também está disponível no site do suporte Oracle, com identificação **1093985.1**. Ele também está disponível em http://suporte.universalrh.com.br/atualizacoes/ORACLE/DOCUMENTOS/WEBLOGIC/HOW_TO_CONFIGURE_WEBUTIL_IN_FORMS_11G.PDF

Os arquivos de instalação da biblioteca *JACOB*, exigida pela *WebUtil* do *Forms 11g*, pode ser obtida em <http://suporte.universalrh.com.br/atualizacoes/ORACLE/DOCUMENTOS/WEBLOGIC/JACOB-1.14.3.ZIP>

Por fim, o próximo passo é instalar o **UNIVERSALRH** e configurar o servidor de aplicação para executar o sistema.

3.3. Instalando os Programas do **UNIVERSALRH** no servidor de aplicação da Oracle

Os procedimentos da instalação do sistema dependem da versão do sistema operacional e do servidor de aplicação da Oracle que será utilizado. O **UNIVERSALRH** é originalmente liberado com executáveis compilados com *Oracle Forms 10g R2* (10.1.2.3.0) para o sistema operacional *Windows*. O cliente que utilizar o *Oracle Forms and Reports 11g R2*, ou sistemas operacionais *UNIX* ou *Linux*, deverá gerar os executáveis da aplicação com o compilador do *Oracle Forms*, disponível em sua instalação. A seção 3.3.8, explica como compilar o sistema com o compilador do *Oracle Forms*.

Cada módulo do **UNIVERSALRH** poderá ter um diretório para abrigar os seus componentes. (ver a documentação da transação [Sistemas](#) no tutorial do módulo de Componentes). Recomendamos essa separação em pastas para organizar a biblioteca de programas da aplicação, visando facilitar a sua manutenção.

Depois de copiados todos os arquivos, deverão ser executados os seguintes procedimentos:

- **Descompactar o arquivo ARHPLL.zip:** arquivo compactado com as bibliotecas compartilhadas, utilizadas pelo **UNIVERSALRH**.
- **Arquivo ARHJ8300.jar:** este arquivo deve ser copiado para diretório *home* de instalação do *Forms Service*:
 - **Oracle Application Server 10g R2:**
[ORACLE_HOME]/forms/java
 - **Oracle Forms and Reports Services 11g R2:**
[ORACLE_HOME]/forms/java
- **Arquivo ARHICONS.jar:** este arquivo deve ser copiado para diretório *home* de instalação do *Forms Service*:
 - **Oracle Application Server 10g R2:**
[ORACLE_HOME]/forms/java
 - **Oracle Forms and Reports Services 11g R2:**
[ORACLE_HOME]/forms/java

3.3.1. Configurando o servidor de aplicação da Oracle para executar o **UNIVERSALRH**

Nesta seção será apresentado como o administrador do servidor de aplicação deverá criar uma seção de configuração no *Oracle Application Server 10g R2* ou *Oracle Forms and Reports Services 11g R2*, permitindo que o servidor de aplicação execute corretamente o **UNIVERSALRH**. Esta configuração deve ser feita diretamente na máquina que roda o servidor de aplicação, onde será necessário customizar os seguintes arquivos:

- **Formsweb.cfg:** este arquivo armazena informações sobre a configuração do ambiente de runtime do serviço do Forms. Nele, o administrador do servidor de aplicação deve criar uma seção de configuração para que o **UNIVERSALRH** possa ser executado pelos usuários, através do navegador *Web*. Este arquivo está localizado no diretório:
 - **Oracle Application Server 10g R2:**

- *[ORACLE_HOME]/forms/server*
 - **Oracle Forms and Reports Services 11g R2:**
[DOMAIN_HOME]/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config
- **Default.env:** este arquivo armazena as variáveis de ambiente que devem ser inicializadas pelo servidor de aplicações, no momento em que o usuário fizer a requisição de execução do **UNIVERSALRH** através do navegador *Web*. Este arquivo deve definir as seguintes variáveis de ambiente:
 - **FORMS_PATH**
 - **PATH**
 - **CLASSPATH**

O arquivo **Default.env** é criado no momento da instalação do servidor de aplicação, sendo necessário configurar as variáveis **FORMS_PATH** e **PATH** com o diretório de instalação do **UNIVERSALRH**. As variáveis **FORMS_PATH** e **PATH** devem ter uma entrada para o diretório que contém as bibliotecas compartilhadas (disponíveis no arquivo **ARHPLL.zip**) do sistema, para que o sistema funcione corretamente. O arquivo **Default.env** deverá ser copiado para outro arquivo, a fim de ser customizado para uso exclusivo do sistema. Este arquivo está localizado no diretório:

 - **Oracle Application Server 10g R2:**
[ORACLE_HOME]/forms/server
 - **Oracle Forms and Reports Services 11g R2:**
[DOMAIN_HOME]/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config
- Copiar o arquivo **ARHUI.dat** (disponíveis no arquivo **ARHPLL.zip**) para a seguinte pasta do servidor de aplicação:
 - **Oracle Application Server 10g R2:**
[ORACLE_HOME]/forms/java/oracle/forms/registry
 - **Oracle Forms and Reports Services 11g R2:**
[DOMAIN_HOME]/config/fmwconfig/servers/WLS_FORMS/applications/formsapp_11.1.2/config/forms/registry/oracle/forms/registry
- Copiar o arquivo **ARHBACKGROUND.jpg** (disponíveis no arquivo **ARHPLL.zip**) para a seguinte pasta do servidor de aplicação:
 - **Oracle Application Server 10g R2:**
[ORACLE_HOME]/forms/java

3.3.2. Exemplo de configuração do Oracle Application Server 10g Release 2 e Oracle Forms and Reports Services 11g R2

Nesta seção apresentaremos um exemplo de configuração do servidor de aplicação para executar o **UNIVERSALRH**. Vale lembrar que os valores das variáveis de ambiente, bem como os diretórios de configuração são apenas para ilustrar a configuração. O administrador do servidor de aplicação deverá fazer os devidos ajustes, conforme o seu ambiente operacional, ou seja, os valores das variáveis de ambiente e os diretórios de instalação devem refletir a real situação da instalação realizada, tanto do servidor de aplicação quanto dos executáveis do **UNIVERSALRH**.

Para começar, vamos definir que a configuração do servidor de aplicação será totalmente baseada em uma seção de configuração exclusiva para o **UNIVERSALRH**. A partir deste ponto a seção será referenciada no texto como **URH**. Vale salientar que uma seção de configuração é definida sempre com o nome delimitado por abre e fecha colchetes, como por exemplo **[URH]**. Portanto faremos a customização do arquivo **FORMSWEB.CFG**, criando a referida seção de configuração. Exemplo:

```
#####
# Secao de configuracao do UNIVERSALRH
#####
[URH]
form=F:\UNIVERSAL\ARH\ARH4000.fmx
separateFrame=True

# PARAMETRO OPCIONAL. PODE SER REMOVIDO DA SECAO DE CONFIGURACAO
# PARA QUE O FORMS SOLICITE INFORMACOES DE LOGIN DO BANCO PARA
# O USUARIO.
userid=JOAO/SILVA@PRODUCAO

# O PARAMETRO TERM ESTA USANDO O ARQUIVO DE CONFIGURACAO
# DE TECLADO FRMPCWEB.RES, PARA MANTER AS MESMAS DEFINICOES
# DE TECLAS DE FUNCAO DO AMBIENTE CLIENTE/SERVIDOR.
#
# PARA O ORACLE APPLICATION SERVER 10G R2.
term=[ORACLE_HOME]\forms\fmrpcweb.res
# PARA O ORACLE FORMS AND REPORTS SERVICES 11G R2.
#term=[ORACLE_INSTANCE]\config\FormsComponent\forms\fmrpcweb.res

# O PARAMETRO workingDirectory DEFINE PARA O SECAO DE CONFIGURACAO
# ONDE SERAO GERADOS OS ARQUIVOS DE LOG DE APLICACAO POR PADRAO.
# (ver seção 2.3.3)
workingDirectory=C:\UNIVERSAL\LOGS

# CUSTOMIZACAO DO ARQUIVO QUE CONTEM A DEFINICAO DAS VARIAVEIS
# DE AMBIENTE UTILIZADAS PARA EXECUTAR O UNIVERSALRH
# O ARQUIVO universalrh.env E UMA COPIA DO ARQUIVO DEFAULT.ENV
envFile=universalrh.env

# REDEFINICAO DA FONTE TAHOMA COMO FONTE PADRAO DO ORACLE FORMS.
# O ADMINISTRADOR DO OAS DEVERA COPIAR O ARQUIVO ARHUI.DAT PARA
# O DIRETORIO:
# * ORACLE APPLICATION SERVER 10G R2
# [ORACLE_HOME]\FORMS\JAVA\ORACLE\FORMS\REGISTRY
# * ORACLE FORMS AND REPORTS SERVICES 11G R2
# [DOMAIN_HOME]/config/fmwconfig/servers/WLS_FORMS/
# applications/formsapp_11.1.2/config/forms/registry/
# oracle/forms/registry
# REPARE PARA O NOME DO ARQUIVO, POIS ELE NAO PODE TER EXTENSAO.
serverApp=arhui
```

```
# IMAGEM DE FUNDO DA MDI DO RUNTIME DO FORMS.
# * ORACLE APPLICATION SERVER 10G R2
# O ADMINISTRADOR DO SERVIDOR DE APLICACAO DEVERA COPIAR
# O ARQUIVO ARHBACKGROUND.jpg
# PARA O DIRETORIO DE INSTALACAO DO SERVIDOR DE APLICACAO:
# [ORACLE_HOME]\FORMS\JAVA
background=/forms/java/ARHBACKGROUND.jpg
# * ORACLE FORMS AND REPORTS SERVICES 11G R2
# NAO E NECESSARIO COPIAR, POIS NESSE SERVIDOR DE APLICACAO
# O ARQUIVO ARHBACKGROUND.jpg SERA LIDO A PARTIR DO ARQUIVO
# ARHICONS.jar
#background=ARHBACKGROUND.jpg

# CONFIGURACAO PARA APRESENTACAO DOS ICONES DA BARRA DE
# FERRAMENTAS DO SISTEMA.
# O ADMINISTRADOR DEVERA COPIAR O ARQUIVO ARHICONS.JAR
# PARA O DIRETORIO DE INSTALACAO DO SERVIDOR DE APLICACAO
# * ORACLE APPLICATION SERVER 10G R2
# [ORACLE_HOME]\FORMS\JAVA
# * ORACLE FORMS AND REPORTS SERVICES 11G R2
# [ORACLE_HOME]\FORMS\JAVA
imagebase=codebase
allowNewConnections=True

# EXEMPLO DE CONFIGURACAO PARA USO DO JRE 1.6 32-bit,
# OU SUPERIOR, PARA CAMADA DE APRESENTACAO DA APLICACAO.
# PARA CONFIGURAR SEM O WEBUTIL, REDEFINIR O PARAMETRO archive
# APENAS COM: frmall.jar, arhicons.jar
archive=frmall.jar, arhicons.jar, frmwebutil.jar, jacob.jar

# EXEMPLO DE CONFIGURACAO PARA USO DO JRE 1.6 32-bit,
# OU SUPERIOR, PARA CAMADA DE APRESENTACAO DA APLICACAO.
# PARA CONFIGURAR SEM O WEBUTIL, REDEFINIR AS VARIAVEIS
# baseHTMLjinitiator, baseHTML E baseHTMLjpi
# APENAS COM: basejpi.htm
baseHTML=basejpi.htm
baseHTMLjinitiator=basejpi.htm
baseHTMLjpi=basejpi.htm

# OS PARAMETROS ABAIXO ESTAO RELACIONADOS COM A CONFIGURACAO
# DA WEBUTIL.
# PARA CONFIGURAR SEM A WEBUTIL, REMOVE-LOS DA SECAO DE
# CONFIGURACAO.
WebUtilArchive=frmwebutil.jar, jacob.jar
WebUtilLogging=off
WebUtilLoggingDetail=normal
WebUtilErrorMode=Alert
WebUtilDispatchMonitorInterval=5
```

```
WebUtilTrustInternal=True
WebUtilMaxTransferSize=16384
```

```
# AJUSTAR OS PARAMETROS jpi_classid E jpi_mimetype
# COM A VERSAO DO JRE QUE SERA UTILIZADO.
# NESTE EXEMPLO E UTILIZADO O JRE 1.6 UPDATE 31
jpi_classid=clsid:CAFEEFAC-0018-0000-FFFF-ABCDEFFEDCBA
jpi_mimetype=application/x-java-applet;version=1.6

# AJUSTAR O PARAMETRO jpi_codebase COM A URL PARA DOWNLOAD
# AUTOMATICO DO PLUG-IN DO JRE DESEJADO.
jpi_codebase=http://java.sun.com/products/plugin/autodl/jinstall-
1_8_0-windows-i586.cab#Version=1,8,0,101
```

```
#####
# Fim da secao de configuracao do UNIVERSALRH
#####
```

Finalmente, o administrador do servidor de aplicação também deverá customizar o arquivos **UNIVERSALRH.ENV** (este arquivo deve ser uma cópia do arquivo **Default.env**).
Exemplo:

```
#
# Search path for Forms applications (.fmx files,
# PL/SQL libraries, identified by PLX extension)
# If you need to include more than one directory,
# they should be semi-colon
# separated (e.g. /private/dir1;/private/dir2)
#
FORMS_PATH=F:\UNIVERSAL\ARH;C:\ORACLE\ODS10G\FORMS

#
# The PATH setting is required in order to pick up the JVM (jvm.dll).
# The Forms runtime executable and dll's are assumed to be in
# C:\ORACLE\ODS10G\bin if they are not in the PATH.
PATH=F:\UNIVERSAL\ARH;C:\ORACLE\ODS10G\bin;C:\ORACLE\ODS10G\jdk\jre\bin\c
lient

###
# INICIO - VARIAVEIS PARA CONFIGURACAO DO WEBUTIL
###

# A VARIABEL WEBUTIL_CONFIG INDICA A LOCALIZACAO DO
# ARQUIVO DE CONFIGURACAO DA WEBUTIL, CUSTOMIZADA PARA A APLICACAO.
# NESSE EXEMPLO, O ARQUIVO WEBUTILURH.CFG E UMA COPIA
# DO ARQUIVO WEBUTIL.CFG.
# * ORACLE APPLICATION SERVER 10G R2
WEBUTIL_CONFIG=[ORACLE_HOME]\forms\server\webutilurh.cfg
# * ORACLE FORMS AND REPORTS SERVICES 11G R2
#WEBUTIL_CONFIG=[ORACLE_INSTANCE]\config\FormsComponent
#\forms\server\webutilurh.cfg
```

```

# ADICIONAR NA VARIÁVEL CLASSPATH AS SEGUINTE BIBLIOTECAS:
# * ORACLE APPLICATION SERVER 10G R2
# [ORACLE_HOME]\forms\java\frmall.jar;
# [ORACLE_HOME]\forms\java\frmwebutil.jar;
# [ORACLE_HOME]\forms\java\jacob.jar;
# [ORACLE_HOME]\jdk\jre\lib\rt.jar
# * ORACLE FORMS AND REPORTS SERVICES 11G R2
# [ORACLE_HOME]\forms\java\frmwebutil.jar;
# [ORACLE_HOME]\forms\java\jacob.jar;

###
# FIM - VARIÁVEIS PARA CONFIGURAÇÃO DO WEBUTIL
###

```

Neste exemplo, o servidor de aplicação é conhecido na rede corporativa como **ORACLE-SERVER**. É necessário ajustar o nome do servidor para que as estações possam executar corretamente o **UNIVERSALRH**. Para maiores informações sobre a identificação do servidor de aplicações na rede, recomendamos consultar o administrador da rede.

3.3.3. Configurando o Diretório de Trabalho no servidor de aplicação da Oracle

Deverá ser criado um diretório no servidor de aplicação para armazenamento de Logs, Traces e arquivos formatados dos procedurais (EX.: SEFIP, CAGED, etc...) e para integração com os arquivos de movimentação (Ex.: Controle de Refeitório). A utilização desse diretório pelo **UNIVERSALRH** é definido pelo parâmetro **workingDirectory**, conforme exemplo da seção 3.3.2.

Ainda, poderá ser criado no servidor de aplicação um diretório virtual para listar o conteúdo do diretório de trabalho através de Browser. Este mapeamento poderá ser realizado conforme o exemplo abaixo (**esta customização deve ser feita no arquivo HTTPD.conf**):

```
Alias /rhlogs/ "C:\UNIVERSAL\LOGS/"
```

```

<Directory C:\UNIVERSAL\LOGS>
    Options Indexes MultiViews
    AllowOverride None
    Order allow,deny
    Allow from all
</Directory>

```

Uma vez definido o diretório virtual, os usuários do sistema acessarão os arquivos de log, trace e de formatação através do seguinte endereço WEB:

<http://oracle-server/rhlogs/>

Por fim, como alternativa, os usuários do **UNIVERSALRH** poderão acessar os arquivos produzidos pelo sistema através do *Windows Explorer*, bastando, para isso, que as

estações de trabalho tenham mapeados o diretório de trabalho definido no servidor de aplicação (estamos levando em consideração um ambiente de rede local, onde a pasta do diretório de trabalho é compartilhada).

Para maiores informações recomendamos consultar a documentação do servidor de aplicação sobre a configuração do servidor HTTP.

3.3.4. Configuração de idioma para a geração de relatórios

A configuração da formatação de data e números para execução de relatórios é feita no serviço do *Reports Server* do *Oracle Application Server*. Essa configuração é feita através da variável `NLS_LANG`, onde é possível definir o território e o conjunto de caracteres que serão utilizados pelo *Reports Server* para executar os relatórios do sistema.

Para atender a formatação Português do Brasil, a variável `NLS_LANG` deverá ser definida com o seguinte valor:

`BRAZILIAN PORTUGUESE_BRAZIL.WE8ISO8859P1`

Para saber mais sobre a configuração da variável `NLS_LANG`, recomendamos consultar o seguinte endereço:

https://docs.oracle.com/html/B13804_02/gblsupp.htm

Também recomendamos a leitura do documento 579980.1, disponível no site do Suporte Oracle.

A seguir relacionamos as formas de configurar a variável `NLS_LANG`, dependendo do sistema operacional onde foi instalado o *Oracle Application Server*:

1. Para servidores Windows:

Customizar a variável `NLS_LANG` no registro do *Windows*, através do editor de registro (`regedit`), no seguinte caminho:

`HKEY_LOCAL_MACHINE\SOFTWARE\ORACLE\KEY_[OASHOMENAME]_HOME`, onde `[OASHOMENAME]` deve ser substituído pelo nome dado ao diretório `HOME` do OAS, estabelecido durante a instalação do servidor de aplicação.

2. Para servidores UNIX e Linux:

Customizar a variável `NLS_LANG`, definida no arquivo `$ORACLE_HOME\bin\reports.sh`

Após o ajuste da variável é necessário reiniciar o serviço do *Reports Server*, para que o novo valor da variável `NLS_LANG` tenha efeito. Ainda, para verificar se a variável está com o valor correto, acesse a página do *Reports Server* onde são relacionadas as variáveis do serviço e os seus respectivos valores, exemplo:

<http://oracle-server/reports/rwservlet/showenv>

A variável `NLS_LANG` é apresentada na seção “*OracleAS Reports Services - Server and Engine Environment Variables*”

3.3.5. Executando o **UNIVERSALRH**

Uma vez instalado e configurado o servidor de aplicações, a execução do sistema deve ser feita através de um navegador *Web*, usando para isso, a **URL** de acesso ao servidor de aplicação. Em nosso exemplo, o nome do servidor foi definido como **ORACLE-SERVER** e a seção de configuração do sistema como **URH**, logo, deveremos criar um atalho, em cada estação, para executar o **UNIVERSALRH**. Exemplo:

<http://oracle-server:7777/forms/frmservlet?config=URH>

Note que, na **URL**, foi especificada a seção de configuração que será usada para rodar a aplicação através do parâmetro **config=**. Além disso, também foi especificada a porta do *listener* do servidor de aplicação, no nosso caso a 7777. O número da porta pode mudar de instalação para instalação, pois o administrador do servidor de aplicação pode alterar para porta que melhor convier para a empresa. Estes detalhes são muito importantes e devem ser observados com cuidado para que a aplicação seja executada corretamente a partir da estação de trabalho. É muito importante que sejam anotados os valores de portas TCP que serão usadas pelos *listener* do servidor de aplicação.

3.3.6. Certificação do *UNIVERSALRH* com o Oracle Application Server 10g Release 2 – Forms and Reports Services

O *UNIVERSALRH* foi certificado com o **Oracle Application Server 10g (10.1.2) Forms and Reports Services**, utilizando as seguintes versões de banco de dados (32-bit e 64-bit):

- 10.1.0.3 (*Oracle Database 10g*);
- 10.2.0.3. (*Oracle Database 10g Release 2*);
- 11.1.0.1.0 (*Oracle Database 11g Release 1*) e;
- 11.2.0.1.0 (*Oracle Database 11g Release 2*).

3.3.7. Certificação do *UNIVERSALRH* com o Oracle Forms and Reports Services 11g Release 2

O *UNIVERSALRH* foi certificado com o **Oracle Fusion Middleware 11g Release 2 (11.1.2.0) - Oracle Forms and Reports**, utilizando as seguintes versões de banco de dados (32-bit e 64-bit):

- 11.2.0.1.0 (*Oracle Database 11g Release 2*), e;
- 12.1.0.2.0 (*Oracle Database 12c*).

3.3.8. Compilação do *UNIVERSALRH* com o compilador do Oracle Forms

A compilação dos formulários, bibliotecas e menus, arquivos com extensão FMB, PLL e MMB, respectivamente, deve ser feita com o compilador do *Oracle Forms*. O compilador está disponível nas instalações para desenvolvedores e para produção do servidor de aplicação. A compilação pode ser feita na linha de comando do sistema operacional. Para tanto, deverão ser configuradas as seguintes variáveis de ambiente antes da compilação:

- **PATH**: deve conter o diretório onde fica o executável do compilador do *Oracle Forms*, arquivo identificado como: *frmcmp.exe* (*Windows*), *frmcmp.sh* (*UNIX* e *Linux*);
- **FORMS_PATH**: deve conter os diretórios onde ficam as bibliotecas do *UNIVERSALRH* e as bibliotecas do *Oracle WebUtil*;
- **ORACLE_HOME**: deve apontar o diretório raiz da instalação compilador do *Oracle Forms*;
- **NLS_LANG**: deve ser definida com o mesmo valor que será utilizado para o ambiente de *runtime*, ou seja, o seu valor deve ser o mesmo da variável

NLS_LANG, que está definida no arquivo de ambiente utilizado na seção de configuração da aplicação.

Para compilar a aplicação em *UNIX* e *Linux*, será necessário criar alguns *links* simbólicos, conforme descrito na seção 29.

Uma vez definidas as variáveis de ambiente, a compilação da aplicação poderá ser feita, na seguinte ordem:

1. Compilar as bibliotecas do sistema (arquivos com extensão **PLL**):
Comando: `frmcmp module_type=LIBRARY module=arquivo.pll
batch=yes window_state=minimize userid=usuario/senha@banco`
2. Compilar os forms do sistema (arquivos com extensão **FMB**):
Comando: `frmcmp module_type=FORM module=arquivo.fmb
compile_all=yes batch=yes window_state=minimize
userid=usuario/senha@banco`
3. Compilar os menus do sistema (arquivos com extensão **MMB**):
Comando: `frmcmp module_type=MENU module=arquivo.mmb
compile_all=yes batch=yes window_state=minimize
userid=usuario/senha@banco`

O resultado da compilação será gravado em arquivo de mesmo nome do módulo compilado, mas com a extensão **ERR**.

Para informações detalhadas sobre a compilação de aplicações *Oracle Forms*, consultar a documentação da Oracle em:

<http://www.oracle.com/technetwork/developer-tools/forms/documentation/index.html>

acesso em agosto de 2015.

4. Servidor de Banco de Dados

4.1. Pré-Requisitos e Configuração do Oracle Database Server

Antes de iniciar os trabalhos e instalação do **UNIVERSALRH** o servidor de banco de dados deverá possuir instalados os seguintes packages fornecidos pela Oracle, que deverão ser dados grants de execute para os usuários criados.

- DBMSPIPE.SQL
- DBMSSQL.SQL
- DBMSLOCK.SQL
- DBMSSNAPSHOT.SQL

Além disso, para os bancos Oracle 11g R1/R2 e Oracle12c, recomendamos configurar o parâmetro de inicialização **SEC_CASE_SENSITIVE_LOGON** com o valor **FALSE**.

Criar um usuário de banco de dados para receber a instalação do modelo de dados do **UNIVERSALRH** e conceder privilégios (Grant) nas seguintes roles:

- CONNECT
- RESOURCE

Obs: Caso o cliente queira utilizar a criação de usuário do **UNIVERSALRH** o seguinte privilégio deverá ser dado explicitamente para o usuário através do usuário **SYS**:

- GRANT CONNECT, RESOURCE TO <USUARIO> WITH ADMIN OPTION;

Ainda, para clientes que utilizam o banco de dados Oracle 10g Release 1 ou Release 2, conceder o seguinte privilégio para a role CONNECT:

- GRANT ALTER SESSION TO CONNECT;

Os seguintes privilégios também deverão ser dados explicitamente para o usuário (**não por Role**), através do usuário **SYS**:

- GRANT ALTER ANY TABLE TO <USUARIO>;
- GRANT ALTER ANY PROCEDURE TO <USUARIO>;
- GRANT ALTER USER TO <USUARIO> WITH ADMIN OPTION;
- GRANT CREATE ANY VIEW TO <USUARIO>;
- GRANT CREATE ANY SYNONYM TO <USUARIO>;
- GRANT CREATE ANY PROCEDURE TO <USUARIO>;
- GRANT CREATE ANY INDEX TO <USUARIO>;
- GRANT CREATE ANY TRIGGER TO <USUARIO>;
- GRANT CREATE USER TO <USUARIO>;
- GRANT CREATE ANY MATERIALIZED VIEW TO <USUARIO>;

- GRANT CREATE ANY DIRECTORY TO <USUARIO>;
- GRANT CREATE EVALUATION CONTEXT TO <USUARIO>;
- GRANT CREATE JOB TO <USUARIO>;
- GRANT CREATE RULE SET TO <USUARIO>;
- GRANT CREATE TABLE TO <USUARIO> WITH ADMIN OPTION;
- GRANT CREATE ANY RULE TO <USUARIO>;
- GRANT DROP ANY TABLE TO <USUARIO>;
- GRANT DROP ANY TRIGGER TO <USUARIO>;
- GRANT DROP ANY MATERIALIZED VIEW TO <USUARIO>;
- GRANT DROP ANY SYNONYM TO <USUARIO>;
- GRANT DROP ANY VIEW TO <USUARIO>;
- GRANT EXECUTE ON DBMS_PIPE TO <USUARIO> WITH GRANT OPTION;
- GRANT EXECUTE ON DBMS_SQL TO <USUARIO> WITH GRANT OPTION;
- GRANT EXECUTE ON DBMS_LOCK TO <USUARIO> WITH GRANT OPTION;
- GRANT EXECUTE ON DBMS_SNAPSHOT TO <USUARIO> WITH GRANT OPTION;
- GRANT EXECUTE ON DBMS_OBFUSCATION_TOOLKIT TO <USUARIO> WITH GRANT OPTION;
- GRANT EXECUTE ON DBMS_RANDOM TO <USUARIO> WITH GRANT OPTION;
- GRANT SELECT ANY TABLE TO <USUARIO>;
- GRANT SELECT ON DBA_SYS_PRIVS TO <USUARIO>;
- GRANT SELECT ON V_\$SESSION TO <USUARIO>;
- GRANT SELECT ON DBA_TAB_COLUMNS TO <USUARIO> WITH GRANT OPTION;
- GRANT SELECT ON DBA_CONSTRAINTS TO <USUARIO> WITH GRANT OPTION;
- GRANT SELECT ON DBA_CONS_COLUMNS TO <USUARIO> WITH GRANT OPTION;
- GRANT SELECT ON DBA_OBJECTS TO <USUARIO> WITH GRANT OPTION;
- GRANT SELECT ON DBA_SCHEDULER_PROGRAMS TO <USUARIO> WITH GRANT OPTION;
- GRANT SELECT ON DBA_SCHEDULER_PROGRAM_ARGS TO <USUARIO>;
- GRANT SELECT ON DBA_SCHEDULER_SCHEDULES TO <USUARIO>;
- GRANT SELECT ON DBA_SCHEDULER_JOBS TO <USUARIO>;
- GRANT SELECT ON DBA_SCHEDULER_CHAINS TO <USUARIO>;
- GRANT SELECT ON DBA_SCHEDULER_CHAIN_STEPS TO <USUARIO>;
- GRANT SELECT ON DBA_SCHEDULER_CHAIN_RULES TO <USUARIO>;
- GRANT SELECT ON DBA_SCHEDULER_JOB_RUN_DETAILS TO <USUARIO>;
- GRANT SELECT ON DBA_SNAPSHOTS TO USUARIO WITH GRANT OPTION;
- GRANT SCHEDULER_ADMIN TO <USUARIO> WITH GRANT OPTION;

Para criar diretórios no servidor de banco de dados executar o seguinte comando:

```
CREAT OR REPLACE DIRECTORY <NOME> AS 'caminho';
```

- GRANT READ ON DIRECTORY <Nome do diretório importado> TO <USUARIO>;
- GRANT WRITE ON DIRECTORY <Nome do diretório visualizado> TO <USUARIO>;

Pedimos especial atenção para **ORACLE_HOME**, que deverá ser substituída pelo conteúdo da variável de ambiente **ORACLE_HOME** definida no servidor de banco de dados.

Estes privilégios devem ser concedidos explicitamente pelo usuário SYS, pois existem funções no **UNIVERSALRH** que utilizam o package DBMS_SQL para execução de comandos SQL e PL/SQL dinâmicos. Para saber mais sobre a execução de comandos SQL

através do package `DBMS_SQL` consulte o manual da Oracle: *Oracle Supplied PL/SQL Packages Reference*.

Para clientes que utilizam o banco de dados **Oracle9i**, o parâmetro de inicialização **`OPTIMIZER_DYNAMIC_SAMPLING`** deve ter o seu valor ajustado para 2. Dessa forma, o otimizador de consultas do Oracle fará uma amostragem das *Global Temporary Tables* (GTT) antes de determinar o plano de execução de uma consulta. Com isso, o banco de dados terá uma performance superior ao executar as consultas que utilizam GTT. Ainda em relação às GTTs, é recomendado que estas tabelas tenham as suas estatísticas removidas do dicionário do banco de dados, conforme recomendação da Oracle. Para isso, o DBA poderá usar as ferramentas **`DBMS_STATS.DELETE_TABLE_STATS`** e **`DBMS_STATS.LOCK_TABLE_STATS`** (para maiores informações, consulte o manual de administração do banco de dados, ou o site <http://metalink.oracle.com>).

Por fim, recomendamos que seja consultada a documentação do Oracle para maiores detalhes sobre a configuração do RDBMS e que este trabalho seja feito por um DBA, com o objetivo de fazer a configuração adequada do banco de dados para otimização da performance do RDBMS.

4.2. No *UNIX* e *Linux*

Exportar a variável de ambiente:

- `LIBPATH=$ORACLE_HOME/lib`

É necessário por conta das Bibliotecas compartilhadas do Java Runtime, em plataformas *UNIX* e *Linux*.

Criar uma variável de ambiente:

`@$URH_TOP/<pasta>/<nomeScript>.sql`

onde `URH_TOP` é uma variável de ambiente criada no Oracle com o caminho do diretório raiz dos programas fontes do ***UNIVERSALRH***.

4.3. No *Windows*

Criar uma variável de ambiente:

`@%URH_TOP%\<pasta>\<nomeScript>.sql`

onde `URH_TOP` é uma variável de ambiente criada no Oracle com o caminho do diretório raiz dos programas fontes do ***UNIVERSALRH***.

5. Conceitos Gerais de Instalação no Banco de Dados

Antes de iniciar a instalação do produto, recomendamos que seja feita uma cópia do Banco de Dados, para, somente após aplicar os scripts de instalação do produto sobre a estrutura de dados atual. Os próximos itens relacionam os passos necessários para realizar a instalação da estrutura de dados do Sistema. Estes passos envolvem a criação de Tabelas, Índices, Constraints, catalogação de Database Triggers e Stored Procedures..

5.1. Criação das Tabelas, Índices, Constraints, Views e Triggers

Para criar todas as especificações do modelo de dados do **UNIVERSALRH**, é necessário executar o script:

Ambiente Windows: **UNIBASE_WIN.sql**

Ambiente Linux: **UNIBASE_LNX.sql**

(estes arquivos estão localizados no diretório **[UNIVERSAL_HOME]\UNIBASE**) diretamente no SQL*Plus, conectando com o usuário que será o proprietário do sistema. Após a execução deste procedimento, todas as tabelas, Índices, Constraints, Views e Triggers serão criados no banco de dados.

5.2. Criação das Tabelas Temporárias

Para possibilitar a execução dos relatórios e programas procedurais nos Sistemas, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para criar as tabelas de trabalho no Banco de dados, executar os arquivos XXXP5ZZZ.sql, onde XXX é a sigla do sistema, P é a identificação de package e ZZZ é o código que identifica o relatório, conectando-se com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

5.3. Criação das Procedures de Banco de Dados

Para que sejam criadas as procedures de Banco de dados, executar os arquivos XXXPZZZZ.sql, onde XXX é a sigla do sistema e ZZZZ é o código que identifica a procedure, conectando-se com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**. Para criar as classes Java, é necessário submeter os arquivos XXXJZZZZ.java, onde XXX é a sigla do sistema e ZZZZ é o código que identifica a classe Java, ao utilitário Oracle LOADJAVA. A linha de comando é a seguinte:

```
loadjava -definer -oci8 -resolve -user usuarioDB/senhaDB@Banco -verbose -force XXXJZZZZ.Java. Onde usuarioDB, senhaDB e Banco são informações para conexão ao banco de dados com o usuário proprietário do UNIVERSALRH.
```

6. Administração de Pessoal (GFP)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, criação de tabelas temporárias, índices e constraints.

6.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GFPPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

6.2. Pré-requisitos para Instalação do Sistema de Administração de Pessoal

São pré-requisitos para instalação do Sistema de Administração de Pessoal estar disponível os seguintes produtos:

O Compilador da linguagem "C" compatível com o Sistema Operacional do equipamento que irá executar o programa GFP8000.pc

Pré Compilador Pro*C da Oracle, com versão compatível com a versão do Banco de Dados e do Compilador "C" do Sistema Operacional do equipamento, que irá executar o programa GFP8000.pc (Programa executor de Cálculos) e do ARH8260.pc (Programa "daemon" do Oracle).

6.2.1. Compilação e linkedição dos componentes escritos em Pro*C/C++ para ambientes *Windows*

Para executar em plataformas INTEL, o programa de cálculo é fornecido como executável pronto para rodar. Entretanto é fornecido o fonte GFP8000.pc, que permite obter o seu executável. A seguir relacionamos os passos necessários para a geração do executável do programa GFP8000, utilizando o compilador Microsoft Visual C/C++ 6.0, ou superior:

- Criar um novo projeto;
- Selecionar a opção “Console Application”;
- No menu Insert, selecionar “Files into Project”;
- Selecionar os arquivos”GFP80NN.c” e uma das seguintes bibliotecas:
 - **Para banco 9.X:** selecionar a biblioteca ORASQL9.lib. Esta biblioteca não está disponível na mídia do **UNIVERSALRH**. Esta biblioteca está disponível na mídia de instalação do banco de dados, e deve ser instalada através do *Oracle Universal Installer*, selecionando o produto *Precompilers*.
 - **Para banco 10.X:** selecionar a biblioteca ORASQL10.lib. Esta biblioteca não está disponível na mídia do **UNIVERSALRH**. Esta biblioteca está disponível na mídia de instalação do banco de dados, e deve ser instalada através do *Oracle Universal Installer*, selecionando o produto *Precompilers*.
 - **Para banco 11.X:** selecionar a biblioteca ORASQX11.lib. Esta biblioteca não está disponível na mídia do **UNIVERSALRH**. Esta biblioteca está disponível na mídia de instalação do banco de dados, e deve ser instalada através do *Oracle Universal Installer*, selecionando o produto *Precompilers*.
- Selecionar a opção *Win32Release*, na opção de seleção de versão;
- Gerar o projeto.
- Ainda, deve ser configurado o projeto para concluir o arquivo header *sqlcpr.h* . através do parâmetro "Additional include directories", Pop-list Category, opção Preprocessador do compilador, localizada na guia C/C++. O arquivo de cabeçalho *sqlcpr.h* define o protótipo da função *sqlclu()*.

Para esta plataforma já fornecemos o executável do programa de cálculo compilado para as versões mais recentes do banco de dados em 32bits e 64bits. Também realizamos testes de execução do cálculo em bancos com as seguintes versões: 10.1, 10.2 ,11.1, 11.2 e 12.1.

6.2.2. Compilação e linkedição dos componentes escritos em Pro*C/C++ para ambientes *Unix*

A seguir apresentamos três formas possíveis de compilação dos programas GFP80NN.pc e ARH8260.pc:

Compilação com arquivo *Makefile* fornecido pela Oracle

A compilação a seguir utiliza o arquivo “demo_proc.mk”, ou “demo_proc_ic.mk” (o nome do arquivo dependerá da versão de Oracle cliente utilizada) para produção de um executável. Este arquivo está disponível na mídia *Companion CD* (arquivo

“demo_proc.mk”) para o banco Oracle 10g. Para o banco 11g, o arquivo “demo_proc_ic.mk” pode ser encontrado na mídia “Oracle examples” ou então “Instant client package – Precompilers”.

Cabe lembrar que o make file pode ser nomeado como o sufixo 32 ou 64, dependendo do tipo de compilação que se deseja fazer, ou seja, compilação para um sistema operacional de 32Bits ou 64Bits. Consulte o administrador do seu ambiente computacional para determinar qual é a arquitetura da máquina, onde será compilado o programa executor de fórmulas.

Antes de utilizar o arquivo Makefile, abra o arquivo em um editor de texto e configure a variável “ICLIBHOME” com o valor “\$(ORACLE_HOME)/lib/” para que as bibliotecas utilizadas na compilação sejam encontradas corretamente.

Para compilar utilizando o Makefile, utilize o seguinte comando:

```
make -f <Makefile> build EXE=<Arquivo>
OBJS=          <Arquivo>.o          PROCFLAGS="USERID=<usuario>/<senha>@<alias>
SQLCHECK=semantics"
```

Onde:

<Makefile>: Arquivo que contém os comandos a serem interpretados pelo programa “make”. No nosso arquivo é o arquivo “demo_proc.mk” ou então o arquivo “demo_proc_ic.mk”

<Arquivo>: Nome do arquivo a ser compilado. Neste caso, utilize somente o nome do arquivo, sem o seu sufixo (por exemplo, informe “ARH8260” para compilar o arquivo “ARH8260.PC”).

<usuario> - Nome do usuário proprietário do sistema **UNIVERSALRH** no banco de dados.

<senha> - Senha do usuário proprietário do sistema **UNIVERSALRH**.

<alias> - Alias de conexão do banco de dados que contém o sistema **UNIVERSALRH**.

Compilação sem o arquivo Makefile da Oracle

A compilação sem o Makefile da Oracle deve ser feita em duas etapas. A primeira é gerar o arquivo fonte em linguagem “C” com o pré-compilador Pro*C/C++. A segunda é a compilação deste arquivo fonte obtido com o compilador C instalado na máquina que irá executar o programa.

Para gerar o arquivo fonte na linguagem “C”, execute o seguinte comando:

```
proc iname=<Arquivo PC> oname=<Arquivo c> parse=none
sqlcheck=semantics userid=<usuario>/<senha>@<alias> hold_cursor=yes
release_cursor=no maxopencursors=500
```

Onde:

<Arquivo PC> - Nome do arquivo de extensão “.PC” que deve ser lido pelo pré-compilador Pro*C/C++ para gerar o arquivo fonte em linguagem “C”. Neste caso, informe o nome completo do arquivo (por exemplo, para utilizar o arquivo “ARH8260.PC”, informe “ARH8260.PC”).

<Arquivo c> - Nome do arquivo fonte em linguagem “C” que deve ser gerado pelo Pro*C/C++.

<usuario> - Nome do usuário proprietário do sistema **UNIVERSALRH** no banco de dados.

<senha> - Senha do usuário proprietário do sistema **UNIVERSALRH**.

<alias> - Alias de conexão do banco de dados que contém o sistema **UNIVERSALRH**.

Se durante a execução do comando ocorrer o seguinte erro:

```
proc: error while loading shared libraries: libclntsh.so.11.1:  
cannot open shared object file: No such file or directory
```

Configure a variável de ambiente “LD_LIBRARY_PATH” com o valor “\$ORACLE_HOME/lib” através do seguinte comando:

```
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
```

Isto fará com que a biblioteca “libclntsh.so.11.1” ou “libclntsh.so.10.1” (no caso da versão 10g) seja localizada durante a execução do pré-compilador.

Antes de seguir para a segunda parte da compilação, certifique-se que o arquivo gerado pelo pré-compilador possui o sufixo “.c” em letras minúsculas. A tentativa de compilação de um arquivo com o sufixo “.C” (maiúsculo) fará com que o programa compilador interprete o arquivo fonte como um arquivo “C++” ao invés de “C”, resultando em erros de compilação.

A segunda parte da compilação irá depender do sistema operacional em que o programa será executado – Linux ou AIX.

Em ambientes Linux deve ser utilizado o compilador “gcc” com os seguintes parâmetros:

```
gcc <Arquivo c> -o <Programa> -lm $ORACLE_HOME/lib/libclntsh.so  
$ORACLE_HOME/lib/libsqlplus.a
```

Onde:

<Arquivo c> - Nome do arquivo de extensão “.c” gerado pelo pré-compilador Pro*C/C++ na primeira etapa deste item. Observe que este arquivo deve possuir o sufixo “.c” em letra minúscula.

<Programa> - Nome do programa gerado pelo compilador.

Como alternativa, é possível ainda informar ao compilador que as bibliotecas da Oracle devem ser obtidas no diretório \$ORACLE_HOME/lib informando o parâmetro “-L” neste comando, não sendo necessário informar o caminho completo das bibliotecas Oracle anexadas ao programa. Neste caso o comando seria:

```
gcc <Arquivo c> -o <Programa> -L $ORACLE_HOME/lib -lm -lclntsh -  
lsqplus
```

Uma terceira forma de executar este comando é omitindo o parâmetro “-L”. Para isto, é necessário que a variável “LD_LIBRARY_PATH” esteja configurada para procurar

bibliotecas do sistema no diretório “lib” da Oracle. Para isto, execute primeiramente o seguinte comando:

```
export LD_LIBRARY_PATH=$ORACLE_HOME/lib:$LD_LIBRARY_PATH
```

Em seguida, o programa pode ser gerado com o seguinte comando:

```
gcc <Arquivo c> -o <Programa> -lm -lclntsh -lsqplus
```

Para realizar a compilação do fonte gerado pelo pré-compilador Pro*C no AIX, utilize o programa “xlc” através do seguinte comando:

```
xlc <Arquivo c> -o <Programa> -I$ORACLE_HOME/rdbms/public -lm  
$ORACLE_HOME/lib/libclntsh.a -bnoquiet -q64 -qarch=pwr4 -qtune=pwr4  
-qcpluscmt
```

Assim como no Linux, também é possível informar ao compilador o diretório onde as bibliotecas da Oracle devem ser obtidas executando o comando da seguinte forma:

```
xlc <Arquivo c> -o <Programa> -I$ORACLE_HOME/rdbms/public -lm -  
L$ORACLE_HOME/lib -lclntsh -bnoquiet -q64 -qarch=pwr4 -qtune=pwr4  
-qcpluscmt
```

Caso ocorram problemas durante a pré-compilação, sugerimos consultar o suporte do fornecedor do Banco de Dados para revisar os parâmetros do arquivo. O mesmo ocorre com o comando de compilação, que são comandos normalmente usados nos ambientes Unix. Caso ocorram problemas durante a compilação, sugerimos consultar o suporte do fornecedor do sistema operacional.

Salientamos que os comandos apresentados neste guia foram testados com sucesso em nossos ambientes Linux e AIX. Caso algum erro ocorra durante as suas execuções, verifique as configurações do seu ambiente e se ele possui todos os pré-requisitos existentes para a utilização das ferramentas.

6.2.3. Pré-requisitos para escolha da modalidade de execução do programa de cálculo

Qualquer que seja a modalidade escolhida, o programa GFP8000 em modo executável deve estar disponível no equipamento que será feita a execução.(ver item **Instalação do Sistema de Administração de Pessoal** neste manual)

6.2.3.1. Execução Local

Disponibilizar o programa executável GFP8000 no equipamento local. Efetuar a atualização da escolha do modo execução do programa de cálculo na transação Modalidade de Execução do Cálculo ([GPR0590](#)) no Gerenciador de Procedimentos..

6.2.3.2. Execução Utilizando Remote Shell (RSH)

Para esta modalidade é requerida a instalação de Software que possibilite a execução de comandos remotos (Remote Shell) a partir do(s) equipamento(s) “clients”, que permitem comandos sobre protocolo TCP/IP.

Exemplo:

PC NFS, PC/TCP, Rexec “Windows NT”.

Salientamos que, para ambientes *Windows 95*, os softwares de rede oferecem o remote shell que não é customizável pela linha de comando do DOS, o que inviabiliza a execução do cálculo por comando remoto.

Entretanto, existe na Internet (**Erro! A referência de hiperlink não é válida.**) ou na biblioteca do **UNIVERSALRH** um software de característica “freeware”. O arquivo **arhrsh.exe**, que um self-extractor e que nele está contido o fonte e executável dos programas de Remote Shell.

Após a extração dos arquivos serão recuperados os seguintes programas:

RSH.exe;

RSHD.exe.

No ambiente *Windows*, deverá ser copiado para o equipamento “client” o arquivo RSH.exe para um diretório mapeado pela variável PATH, pois é o programa que envia o comando para execução remota no servidor.

No ambiente *Windows NT/2000/2003*, deverá ser copiado o arquivo RSHD.exe para um diretório mapeado pela variável PATH. Este é o programa que recebe o comando enviado pelo “client “ (Este programa habilita um daemon de remote shell no NT).

O comando para ativar o programa RSHD.exe possui a seguinte sintaxe:

```
C:\>rshd -r
```

Obs.:

Recomenda-se que este comando seja informado em um atalho na pasta Iniciar para possibilitar a sua execução automática na inicialização do sistema operacional.

Nos ambientes *UNIX* e *Linux*, este serviço já está disponibilizado pelo próprio sistema operacional, não necessitando de customização.

Caso o remote shell da sua instalação não esteja funcionando, recomendamos entrar em contato com o serviço de suporte do sistema operacional do seu fabricante de hardware.

Em caso de seu sistema operacional ser o IBM AIX, faça as seguintes verificações:

- Verifique se no seu diretório *HOME* existe o arquivo *.rhosts*. Este arquivo deve conter os nomes das máquinas que podem executar o *remote shell*.

Ex.: Máquina Usuário

SERVER01 Joao

SERVER02 Maria

SERVER03 Jose

- Trocar as permissões: `chmod 600 .rhosts`
- Verificar no diretório */etc* se o arquivo *host.equiv* tem definido os hosts de acesso.
- Verificar no diretório */etc* se o arquivo *services* tem as seguintes definições: `fsh-spx 222/tcp` e `fsh-spx 222/udp`.

- Verificar no diretório /etc se o arquivo services está sem comentários no seguinte comando: rshd
- Executar o comando: refresh -s inetd
- Executar o comando: startsrc -t shell
- Executar o comando: lssrc -t shell
- No diretório /usr/sbin trocar as permissões para:
 chmod 555 rshd;
 chmod u+s rshd.

Disponibilizar o programa executável GFP8000 no equipamento que irá executar o cálculo.

Efetuar a customização do comando em Runtimes no item [EXECALC](#) no Gerenciador de Componentes.

Efetuar a customização do diretório padrão para geração dos logs no Runtime [DSTLOGCALC](#).

Efetuar a atualização da escolha do modo execução do programa de cálculo na transação Modalidade de Execução do Cálculo ([GPR0590](#)) no Gerenciador de Componentes.

6.2.3.3. Execução utilizando o Servidor de Banco de Dados

Para executar o programa de Cálculo nesta modalidade, devemos executar os seguintes passos:

Catalogar a Stored Procedure ARHP8260 com o script ARHP8260.sql. Para executar esta Stored Procedure, o usuário do Banco de Dados deverá possuir privilégios de **execute procedure**.

Exemplo:

```
SQL>grant execute to <usuário>.
```

Compilar, no ambiente do servidor do banco de dados, o programa ARH8260.pc. (ver item 6.2.2). Este programa deverá ser ativado no servidor do banco de dados sempre após a inicialização do Banco de Dados. Com ele, é possível executar o programa de cálculo no servidor do banco de dados, pois ele monitora um canal de comunicação gerado pelo pacote de banco ARH8260.

Disponibilizar o programa executável GFP8000 no equipamento que irá executar o cálculo.

Efetuar a atualização da escolha do modo execução do programa de cálculo na transação Modalidade de Execução do Cálculo ([GPR0590](#)) no Gerenciador de Componentes.

7. Administração de Recursos Humanos (ARH) (Gerenciador de Componentes)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

7.1. Catalogar as procedure de banco de dados

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação ARHPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**. Para criar as classes Java, é necessário submeter os arquivos ARHJZZZZ.java, onde ZZZZ é o código que identifica a classe Java, ao utilitário Oracle LOADJAVA. A linha de comando é a seguinte:

```
loadjava -definer -oci8 -resolve -user usuarioDB/senhaDB@Banco -verbose -force ARHJZZZZ.Java. Onde usuarioDB, senhaDB e Banco são informações para conexão ao banco de dados com o usuário proprietário do UNIVERSALRH.
```

Nota:

Com este procedimento o usuário estará instalando também o sistema Gerenciador de Componentes.

7.2. Para acessar o **UNIVERSALRH**

Concluída as etapas anteriores da instalação, o acesso ao **UNIVERSALRH** estará bloqueado para qualquer usuário, pois o controle de acesso não possui nenhum usuário com permissão de acesso habilitado. Para criar o perfil de administração, é necessário executar o programa **ARH8300.SQL**. Este programa cria o perfil administrativo para o usuário proprietário das tabelas do **UNIVERSALRH**. Após a sua execução, o proprietário do sistema terá acesso ao menu principal do **UNIVERSALRH** e ao menu principal do sistema Gerenciador de Componentes, não sendo possível navegar em outros sistemas.

Deverá ser importado o arquivo **ARH.dmp**, executar “truncate” nas tabelas “CG_REF_CODES” e “CG_FORM_HELP” e submeter o arquivo enviado (ARH.dmp) ao utilitário “IMP”.

Exemplo:

```
IMP USUÁRIO/SENHA FILE = ARH.DMP GRANTS = NO IGNORE = YES  
FROMUSER = ARHDES TOUSER = USUÁRIO-OWNER
```

Obs:

USUÁRIO – USUÁRIO SYSTEM

FILE – colocar todo o path do arquivo

USUÁRIO OWNER – é o usuário dono do modelo (OWNER).

Acessando o sistema de Componentes, devemos efetuar as definições necessárias a localização das bibliotecas de executáveis de cada Sistema, pois sem a qual o **UNIVERSALRH** não poderá executar nenhum programa que não seja pertencente a sigla ARH.

Obs.: Para efetuar esta tarefa é necessário consultar a documentação Tutorial de Componentes, no item [Sistemas](#) do sistema Gerenciador de Componentes.

Efetuada esta tarefa o proprietário da aplicação terá acesso a todos os Sistemas e a todas as transações sem restrição, podendo proceder todas as outras customizações do produto.

Abaixo mostramos um exemplo de cadastramento dos “Path do Sistema” de instalação na transação **Sistemas** do Gerenciador de Componentes:

Sig	Nome do sistema	Path
ARH	Administracao de Recursos Humanos Vrs 3	\\RS11UNI\FONTES\UNIVERSAL\ARH\
GAP	Avaliacao de Pessoal	\\RS11UNI\FONTES\UNIVERSAL\GAP\
GBE	Beneficios	\\RS11UNI\FONTES\UNIVERSAL\GBE\
GCC	Controle de Custos	\\RS11UNI\FONTES\UNIVERSAL\GCC\
GCS	Cargos e Salarios	\\RS11UNI\FONTES\UNIVERSAL\GCS\
GCV	Convenios	\\RS11UNI\FONTES\UNIVERSAL\GCV\
GES	eSocial	\\RS11UNI\FONTES\UNIVERSAL\GES\
GFA	Frequencia e Acesso	\\RS11UNI\FONTES\UNIVERSAL\GFA\
GFP	Administracao de Pessoal	\\RS11UNI\FONTES\UNIVERSAL\GFP\
GMT	Medicina do Trabalho	\\RS11UNI\FONTES\UNIVERSAL\GMT\
GPR	Procedimentos	\\RS11UNI\FONTES\UNIVERSAL\GPR\
GPT	Processos Trabalhistas	\\RS11UNI\FONTES\UNIVERSAL\GPT\
GRS	Recrutamento & Selecao	\\RS11UNI\FONTES\UNIVERSAL\GRS\
GST	Seguranca do Trabalho	\\RS11UNI\FONTES\UNIVERSAL\GST\
GTR	Treinamento	\\RS11UNI\FONTES\UNIVERSAL\GTR\
I01	Contabilidade Universal	\\RS11UNI\FONTES\UNIVERSAL\I01\
I02	Contas a Pagar Universal	\\RS11UNI\FONTES\UNIVERSAL\I02\
I03	Contas a pagar Oracle Applications R.11	\\RS11UNI\FONTES\UNIVERSAL\I03\
I07	Contabilidade IFS	\\RS11UNI\FONTES\UNIVERSAL\I07\
I08	HomologNet	\\RS11UNI\FONTES\UNIVERSAL\I08\
I09	eSocial	\\RS11UNI\FONTES\UNIVERSAL\I09\

8. Integração da Contabilidade **UNIVERSALRH** (I01)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

8.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação I01PZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

8.2. Criar tabelas

IMPORTANTE: Antes de executar o script I03BASE.syn, deverá ser customizado o schema do usuário proprietário do R11, antes do nome de cada tabela, para criar corretamente os sinônimos no schema do usuário proprietário do **UNIVERSALRH**.

8.3. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação I03PZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

9. Integração Contabilidade IFS (I07)

Para a integração com a Contabilidade do IFS foram utilizadas tabelas as quais deverão ter os privilégios no usuário proprietário do **UNIVERSALRH**.

Criar sinônimos, no usuário dono do **UNIVERSALRH**, e dar privilégios de select e references para as seguintes tabelas do sistema IFS:

- ACCOUNT
- GEN_LED_VOUCHER

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

9.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação I03PZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

10. Procedimentos (GPR)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

10.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GPRPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

11. Recrutamento e Seleção (GRS)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

11.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GRSPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

12. Cargos e Salários (GCS)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

12.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GCSPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

13. Treinamento (GTR)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

13.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GTRPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

14. Convênio (GCV)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

14.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GCVPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

15. Benefício (GBE)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

15.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GBEPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

16. Segurança do Trabalho (GST)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

16.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GSTPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

17. Medicina do Trabalho (GMT)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

17.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GMTPTZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

18. Avaliação de Pessoal (GAP)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

18.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GAPPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

19. Controle de Custos (GCC)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

19.1. Catalogar as procedure de banco de dados

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GCCPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

20. Processos Trabalhistas (GPT)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

20.1. Catalogar as procedure de banco de dados

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GPTPZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

21. eSocial (GES)

A seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

21.1. Catalogar as procedure de banco de dados

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação GESPZZZZ.sql, GESMZZZZ.sql, GESRZZZZ.sql, GESAZZZZ.sql, GESGZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

22. Específico SESI (E01)

Em caso de contratação deste módulo, a seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

22.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação E01PZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

Se não houver contratação deste módulo, efetuar somente o passo 21.1 Criação das Tabelas, pois todos os módulos do Sistema **UNIVERSALRH** são compartilhados entre si.

23. Multifunções (E03)

Em caso de contratação deste módulo, a seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

23.1. Catalogar as procedure de banco de dados

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação E03PZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

Se não houver contratação deste módulo, efetuar somente o passo 23.1 Criação das Tabelas, pois todos os módulos do Sistema **UNIVERSALRH** são compartilhados entre si.

24. Aerus (E05)

Em caso de contratação deste módulo, a seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

24.1. Criar as Tabelas Temporárias

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para criar as tabelas de trabalho no Banco de dados, executar o arquivo E055ZZZ.sql, onde ZZZ é o código que identifica o programa, conectando-se com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

24.2. Catalogar as procedure de banco de dados

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação E05PZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

25. Wal-Mart (E07)

Em caso de contratação deste módulo, relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de database triggers, stored procedure, a criação de tabelas temporárias, índices e constraints.

25.1. Criar as Tabelas Temporárias

Para possibilitar a execução dos relatórios e programas procedurais no Sistema, deverão ser criadas as tabelas de trabalho. Este procedimento se deve ao fato de que os programas não removem e nem criam as tabelas de trabalho. Deste modo, estas tabelas deverão ser criadas para cada um dos usuários do Banco de Dados, permitindo que cada relatório possa ser executado simultaneamente, por mais de um usuário.

Para criar as tabelas de trabalho no Banco de dados, executar o arquivo E075ZZZ.sql, onde ZZZ é o código que identifica o programa, conectando-se com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

25.2. Catalogar as procedure de banco de dados

Para serem catalogadas todas as procedure de banco de dados, executar todos os arquivos com a estrutura de formação E07PZZZZ.sql, onde ZZZZ é o código da procedure, conectando-se ao banco de dados com o SQL*Plus como usuário proprietário das tabelas do **UNIVERSALRH**.

Nota: Caso você não consiga acessar os módulos aplicativos, verifique se estão corretos os dados na transação [Sistemas](#) do sistema Gerenciador de Componentes, tais como Sigla e Path para cada sistema. Esses dados incorretos fazem com que o **UNIVERSALRH** não localize os programas executáveis, que por consequência não permite a navegação no sistema.

Se não houver contratação deste módulo, efetuar somente o passo 24.1 Criação das Tabelas, pois todos os módulos do Sistema **UNIVERSALRH** são compartilhados entre si.

26. Universal Business Interface (UBI)

Em caso de contratação desde módulo, a seguir relacionamos os passos para efetuar a criação da estrutura de dados. Estes passos envolvem a catalogação de procedure de banco de dados, criação de tabelas, índices e constraints.

26.1. Catalogar os procedure de Banco de Dado

Para serem catalogadas todas as procedure de banco de dados, executar o arquivo UBIBASE.SQL conectando-se ao banco de dados com o SQL*PLUS como usuário proprietário das tabelas do **UNIVERSALRH**.

27. Orientador

O Orientador é um conjunto de páginas HTML que auxiliam o usuário na utilização das funções do **UNIVERSALRH**, baseado na visão de processos da área de recursos humanos. Para ativar o Orientador no sistema, deverá ser feito, em primeiro lugar, o *download* das suas caixas de atualização, disponíveis no site de suporte do **UNIVERSALRH**. As caixas devem ser descompactadas na pasta escolhida (obedecendo a sua ordem numérica) no momento da instalação do sistema, para armazenar os arquivos HTML e imagens do Orientador.

Para utilizar o Orientador, o administrador da instalação deverá criar um diretório virtual no servidor de aplicação. Para tanto, será necessário customizar o arquivo **HTTPD.conf**, e adicionar o diretório virtual, conforme exemplo abaixo:

```
Alias /orientadordocs/ "F:\UNIVERSALRH\ORIENTADOR/"
```

Após ter criado o diretório virtual, será necessário customizar os *Runtimes*:

- **INSTORI**: deve ser cadastrado para informa ao sistema que o Orientador está instalado;
- **DOCWEBORI**: cadastrar o diretório virtual criado */orientadordocs/*. Dessa forma, o sistema terá condições de abrir uma nova janela do navegador de *Internet* com o menu do Orientador.

O cadastro dos *Runtimes* é feito no módulo **Componentes**, menu *Transações* → *Runtimes*. Para obter mais informações sobre os *runtimes* aqui relacionados, consulte a documentação de transações (arquivo ARHD4054.pdf) do módulo de Componentes.

A documentação de instalação do **UNIVERSALRH** Orientador está disponível no site de suporte do sistema, em suas caixas de atualização. O nome do arquivo é **ORID4001.pdf**. Já as orientações para realizar a atualização dos componentes do Orientador, estão documentadas no arquivo **ORID4002.pdf**, também disponível em suas caixas de atualização.

28. Uso do **UNIVERSALRH** em modo **WEB**

A partir do **UNIVERSALRH** Vrs 10.0 a sua execução é feita somente através da **WEB**, bastando para isso que sua organização possua alguns pré requisitos que são:

- Um endereço de página na **WEB**, ou Intranet que permita ao usuário efetuar a conexão;
- Possuir um apresentador de páginas da web (browser) instalado no equipamento do usuário;
- O plug-in Java *Runtime Environment* 1.6 32-bit, ou superior, instalado no equipamento do usuário. Este plug-in pode ser encontrado no seguinte endereço da Internet <http://www.oracle.com/technetwork/java/javase/downloads/index.html> (acesso em agosto de 2015), e;
- Um dos seguintes servidores de aplicação da Oracle:
 - **Oracle Application Server 10g Forms and Reports Services (10.1.2);**
 - **Oracle Fusion Middleware Forms and Reports Services 11g (11.1.2.2)**

Os arquivos executáveis contidos na mídia do **UNIVERSALRH**, são compilados e gerados em plataforma *Windows*. Caso o ambiente de execução do **UNIVERSALRH** seja outro, por exemplo *UNIX* ou *Linux*, será necessário recompilar as bibliotecas compartilhadas (os arquivos com extensão PLL, disponíveis no arquivo ARHPLL.zip), os *forms* (arquivos com extensão fmb) e os *menus* (arquivos com extensão mmb).

A recompilação dos programas deve ser feita no servidor de aplicação, com o compilador do *forms* na plataforma usada, visto que todos os fontes do **UNIVERSALRH** estão na mídia. Para maiores detalhes sobre compilação de forms, consulte a seção 3.3.8.

29. Compilação do **UNIVERSALRH** em sistemas **UNIX** e **Linux**

A recompilação das PLL, Forms e Menus deverá ser feita observando o seguinte:

- Definir a variável de ambiente FORMS_PATH com uma entrada para o diretório onde estão gravados os arquivos PLL e PLX do **UNIVERSALRH**;
- Criar links simbólicos para os arquivos PLL, conforme lista a seguir ([link simbólico](#) → **arquivo alvo**):
 - `adrh.olb` --> `ADRH.olb`
 - `Adrh.olb` --> `ADRH.olb`
 - `ADRH.OLB` --> `ADRH.olb`
 - `Ofgmes.pll` --> `OFGMES.pll`
 - `Ofgmes.plx` --> `OFGMES.plx`
- Compilar todas as bibliotecas compartilhadas, arquivos com extensão **pll**, do **UNIVERSALRH**;
- Compilar todos os Forms, arquivos com extensão **fmb**, do **UNIVERSALRH**, e;
- Compilar todos os Menus, arquivos com extensão **mmb**, do **UNIVERSALRH**

30. Configurações Pós-Instalação

Deverão ser cadastrados os seguintes códigos na transação de runtimes no módulo de Componentes: **DOCWEB**, **DOCWEBORI**, **REPWEB**, **ARHLOGO**, **SQLPLUS**, **SQLLOAD**, **NOMEIASSVR**, para determinar a execução dos programas, acesso à documentação e acesso à documentação do Orientador.

Além de configurar estes runtimes, será necessário indicar para o sistema a localização dos seus programas. Este cadastramento também deve ser feito no módulo de **Componentes**: menu *Transações* → *Sistemas*. Para maiores informações sobre estes *Runtimes* e da transação *Sistemas* verificar o tutorial do módulo de Componentes.

30.1. Importação dos eventos e códigos internos

Importar o arquivo de Eventos e Códigos Internos do **UNIVERSALRH** (arquivo GFPEICl.dat), através do módulo de **Administração de Pessoal**: menu *Procedurais* → *Integração de arquivos externos* → *Integra eventos e códigos internos*.

Este arquivo localiza-se na pasta de instalação do módulo de Administração de Pessoal (GFP).

30.2. Importação das informações internas

Importar as informações internas do sistema para que a função de busca de componentes seja ativada no **UNIVERSALRH**. A importação é feita no módulo de **Componentes**: menu *Procedurais* → *Atualiza informações internas*.

30.3. Procedimentos adicionais para Cloud Pag

Na base de produção, referente aos dados de responsabilidade de consultores será necessário executar o processo ARH8220.sql, este processo disponibiliza para compartilhamento tabelas com dados já preenchidos para as bases de clientes do Cloud Pag.

Na base dos clientes do Cloud Pag, deverá ser executado o ARH8230.sql, este processo por meio de DataBase Link, acessa para leitura as tabelas preenchidas disponibilizadas.

ARH0010 – Sistemas

Transação destinada à manutenção de endereços de bibliotecas de todos os sistemas que estão sendo utilizados no site de um cliente.

Para cada sistema deverá ser customizado o caminho para a localização dos programas executáveis do **UNIVERSALRH**. Podem ser cadastrado “Path por Sistema Padrão” e “Path por Sistemas por Perfil”, deste modo o sistema poderá executar programas diferentes por perfil de acesso. Observe que os sistemas têm suas identificações estabelecidas por siglas, siglas essas que são pré-estabelecidas pelo desenvolvedor.

Para identificar um sistema, utilizamos três caracteres, que são usados como prefixo na identificação de um programa. (**PPP**)

Por exemplo:

No sistema Gerenciador de Benefícios temos como sigla “**GBE**”.

Para identificar um programa, a sigla deve ser acompanhada de um sufixo, que é o identificador do programa. Logo, a identificação de um executável é composta pela Sigla do Sistema com 3 (três) caracteres, seguida do Sufixo de até cinco caracteres, que identifica o programa. (**SSSSS**).

Por exemplo:

Se a sigla GBE for seguida do sufixo 4000 (**GBE4000**) identificará o Menu Principal do Sistema Gerenciador de Benefícios.

Como as siglas são pré-estabelecidas. Relacionamos a seguir os sistemas e suas siglas para cadastramento na transação Sistemas no Sistema Gerenciador de Componentes.

Administração de Recursos Humanos	ARH (Componentes)
Gerenciador de Recrutamento e Seleção	GRS
Gerenciador de Cargos e Salários	GCS
Gerenciador de Treinamentos	GTR
Gerenciador de Administração de Pessoal	GFP (Folha de Pagamento)
Contabilização UNIVERSALRH	I01
Integração Contas a Pagar – Oracle Application R11	I03
Gerenciador de Avaliação de Pessoal	GAP
Gerenciador de Convênios	GFA
Gerenciador de Frequência e Acesso	GCV
Gerenciador de Benefícios	GBE
Gerenciador de Segurança do Trabalho	GST
Gerenciador de Medicina do Trabalho	GMT
Gerenciador de Procedimentos	GPR
Gerenciador de Controle de Custos	GCC
Gerenciador de Processos Trabalhistas	GPT
Gerenciador do eSocial	GES

A lógica de localização dos diretórios segue a seguinte estrutura:

82 - Componentes (ARH)

Define que a integração contábil será acumulada pela estrutura organizacional.

ACUMUCCCT

Define que a integração contábil será acumulada pelo centro de custo contábil.

ACUMUNATC

Define que a integração contábil será acumulada pela natureza do custo.

ACUMUNEG

Define que a integração contábil será acumulada pela unidade de negócio.

AGENDABG

Define o comando para executar o agendamento em background. Deve-se cadastrar na coluna **Cmd** o comando que permite a execução de processos em background relativamente ao sistema operacional usado pelo Cliente. Para maiores informações, verifique o Runtime [EXECALCBG](#).

ANTECIPA_EXAMES

Permite selecionar com quantos dias de antecedência a pessoa física será chamada para a realização de exames.

ARHLOGO

Define o nome do arquivo de imagem que será utilizado pelo **UNIVERSALRH** para customizar a toolbar do sistema. O arquivo deve ser do tipo GIF.

ATITSEV

Quando este identificador esta definido **NÃO** lista as atitudes ([GAP0410](#)) definidas em severidades ([GAP0420](#)).

BENEPERC

Permite selecionar como será gerado o cálculo do campo Valor Efe da tabela LANÇAMENTOS BENEFÍCIOS. Quando este runtime estiver customizado será aplicado sobre o valor o percentual registrado no campo PERC da tabela REG_BENEFICIOS, caso contrário será gerado o próprio valor.

CARIMBO

Permite selecionar a impressão ou não do carimbo padronizado do CGC nos relatórios de **Rescisão de Contrato**. O padrão é não imprimir o carimbo. Caso seja informado este Runtime-ID o carimbo passará a ser impresso.

83 - Componentes (ARH)

CBO

Define a partir de que data passou a ser utilizada a nova classificação de CBO (Código Brasileiro de Ocupação). Este RUNTIME é definido por programa e na coluna CMD deve ser informada a data de início de utilização da classificação no formato DDMMYYYY.

Exemplo:

RUNTIME_ID = CBO.XXXZZZZ,

CMD = 01012003

Onde:

XXX - É a sigla do sistema (digitado em maiúsculo);

ZZZZ - É o número do programa.

Neste caso, execução com data referência (ou competência) maior que 01.01.2003 será utilizada a classificação de 2002 e para data menor que 01.01.2003 será utilizada a classificação de 1994.

CBOCARGOSEGATUA

Esse RUNTIME será utilizado em todos os programas que necessitam do Código Brasileiro de Ocupação.

Quando esse RUNTIME estiver cadastrado, significa que o CBO será buscado na tabela Segmento de Atuação quando houver registro em REG_SEGMENTOS_ATUACOES ou será buscado na tabela Cargos quando houver registro em REG_CARGOS.

CHAPASEQ

Quando este identificador está definido a transação de Registro de Emprego gera o número da matrícula automaticamente, caso este não tenha sido digitado.

Informar no campo CMD o valor um (1) para geração do sequencial do registro de emprego somando um (1) na última matrícula.

Informar no campo CMD o valor dois (2) para gerar a matrícula do funcionário sendo o número da pessoa física concatenado com um sequencial de três dígitos iniciando em um (1). Esse sequencial é por pessoa física, independente da empresa.

CHECKUP

Define se a seleção para a chamada de exames e espera de visitas médicas será realizada através de checkup.

CIDSEQ

Quando este identificador está definido a transação de CIDS passará a gerar o código automaticamente, caso não tenha sido digitado.

CONFIGMASC

Indica se a substituição de máscaras de formatação está habilitado.

CONFIGTERMO

Indica se o ajuste dos prompts de forms/reports está habilitado.

CONTROLELIQBANCO

Define que haverá controle da geração de arquivos para o banco, não permitindo que seja realizada a geração do banco mais de uma vez para a mesma empresa, filial, banco, data de pagamento e tipo de pagamento.

CTRLDOCEXTERNO

Define que o controle do armazenamento dos documentos organizacionais será feito através de um programa externo, informar no campo RUNT_CMD qual o comando que deverá ser executado para busca do arquivo. Esse programa receberá como parâmetro o nome do documento e o número do mesmo. Após a busca, será realizada a visualização usando a url informada em URLDOCVIEW.

DAEMON_DBLINK

Indica o nome do dblink padrão para execução do daemon remoto (ARH8260).

DAEMON_PADRAO

Indica em qual pipe o programa de cálculo será executado, quando não informado será executado em 'ARH8260'.

DEFGTT

Define que as tabelas temporárias de relatórios deverão ser criadas como 'GLOBAL TEMPORARY TABLE'.

DEFTIMER

Define o tempo de espera para cancelar a execução de uma solicitação de cálculo pelo programa executor de cálculos. Caso este RUNTIME não seja especificado, o tempo padrão de espera é de dez (10) minutos. Para alterar o tempo padrão basta cadastrar este RUNTIME e na coluna CMD informar o valor em minutos desejado. O valor mínimo informado deve ser um (1) minuto. Veja o exemplo de cadastramento abaixo:

Runtime: DEFTIMER

Cmd: 1

Neste caso foi definido que o tempo de espera para cancelar a solicitação de cálculo em execução será de um (1) minuto.

DIAAFAST

Quando este identificador estiver definido a transação de Acidentes não irá considerar o dia de acidente como dias perdidos.

DIASESTABIL

Define o limite de dias perdidos por afastamento decorrente de acidente de trabalho. Caso o número de dias perdidos seja superior ao informado neste RUNTIME (coluna CMD), será gerado automaticamente o registro de estabilidade.

No caso de não ser informado será considerado 15 dias para o limite de dias perdidos.

DIASFERIAS

Define qual a função a ser usada para cálculo de avos de férias.

O usuário deverá criar uma função que receba como parâmetros a data de Referência do registro de férias, Início e fim de período aquisitivo, e retornar um valor numérico com o número de avos de férias.

No campo comando cadastrar corretamente a chamada da função.

DIASPERPROP

Este Runtime permite aplicar o critério de proporcionalidade ao Número de Dias Perdidos de Férias Devido a Faltas (NDPF) no cálculo do Saldo de Férias Proporcional (SFProp) relativo ao número de avos, os quais são determinados pela data de referência no Registro de Período Aquisitivo.

Para melhor compreensão dessa função demonstramos os passos aplicados a esse cálculo.

Determina-se o número de avos ao qual o colaborador tem direito calculando o número de meses entre a Data de Início do Período Aquisitivo e a data de Referência do registro.

Determina-se o Número de Dias Perdidos de Férias Proporcional efetuando o rateio do Número de Dias Perdidos de Férias por Faltas (NDPF) pelo número de avos.

O valor determinado pelo cálculo anterior é subtraído do Saldo de Férias Proporcionais (SFProp).

Sem a aplicação do runtime o sistema usa o critério de desconto total de Dias Perdidos de Férias (DPF) no cálculo do Saldo de Férias Proporcional (SFProp).

DIASRFIR

Controla o resultado do número de dias obtido pela fórmula **DATA ATUAL DO SISTEMA** (*SYSDATE*) **menos** a Data de Indicação de Retroativos (GPR0370). Se o resultado da fórmula for maior do que o valor indicado por este runtime, então será apresentada uma mensagem de erro para o usuário, impossibilitando a gravação do registro na tabela referenciada pela transação Monitor Retroativos (GPR0310). Se este runtime não estiver definido, será adotada a quantidade de 30 dias como valor padrão.

DIGVERIF

Controla o dígito verificador da matrícula.

Quando este identificador está definido o último dígito da matrícula é considerado dígito verificador.

Deve ser cadastrado na coluna CMD que módulo de verificação deve ser utilizado:

86 - Componentes (ARH)

- MÓDULO_11 utiliza a rotina CALCULA_MODULO_11 para validar o dígito informado.
- MÓDULO_10 utiliza a rotina CALCULA_MODULO_10 para validar o dígito informado.

DIRCURIMP

Define o diretório (no servidor de banco de dados) que contém os arquivos de currículo para importação.

DIRDOCIMP

Runtime com o nome do diretório onde o arquivo estará armazenado para salvar no banco de dados.

Obs: Este diretório deve ser criado no servidor de banco de dados, verificar na Guia de Instalação ([ARHD4001](#)) os grants necessários para a criação do diretório. Esse diretório deverá estar mapeado no Oracle Directory.

DMLREP

Permitir a escolha do comando TRUNCATE ou DELETE.

Durante a execução de relatórios e programas procedurais, podem ser usadas as chamadas tabelas temporárias. Estas tabelas são esvaziadas (removidos os registros) a cada início de execução. Para efetuar esta tarefa, podem ser usados os comandos de “Truncate” ou “Delete”, devendo ser escolhido o comando mais adequado à instalação do Cliente. Para alterar o comando de “Runtime”, basta cadastrar na coluna CMD o comando DELETE. A partir desta customização os relatórios e procedurais utilizarão o comando DELETE para efetuar a limpeza das tabelas temporárias.

Runtime = DMLREP, Cmd = (DELETE)

Default - Truncate.

DOCWEB

Define a URL que contém os arquivos de Help do **UNIVERSALRH** em ambiente Web. A URL deverá ser definida na coluna CMD.

Runtime – DOCWEB, Cmd = (http://www.site/diretório_virtual/)

DOCWEBORI

Define o diretório virtual que contém os arquivos do Orientador do **UNIVERSALRH**.

O diretório virtual deverá ser definido na coluna CMD.

Runtime – DOCWEBORI, Cmd = (/diretório_virtual/).

DSTLOGCALC

Define o diretório de geração do arquivo de LOG do executor de cálculo (GFP8000). Caso esse runtime não seja definido o GFP8000 criará o LOG no diretório de inicialização do **UNIVERSALRH**.

Runtime = DSTLOGCALC, Cmd = (diretório de gravação do arquivo de LOG).

Default - sem default

87 - Componentes (ARH)

Exemplo:

RUNTIME_ID = DSTLOGCALC,

CMD = C:\TEMP\

Onde:

C:\TEMP\ - É o diretório onde será gravado o arquivo de LOG do programa GFP8000.

Para plataformas UNIX a sintaxe do diretório é, por exemplo, /TEMP/.

DTAVPREV

Imprime a data de rescisão no lugar da data de aviso prévio quando esta data for nula.

DTCOMPFER

Restringe pela data de competência o Movimento Variável, quando o contracheque for do tipo férias.

EFCSURL

Define o endereço da intranet(URL) para acionamento do web service a ser para integração do UNIVERSALRH com EFCS da SAP.

EFCSNSSOAPENV

Define o preenchimento do parâmetro constante no XML referente ao 'soapenv', onde deve contar o name space para o soapenv do eFCS.

EFCSNSWEB

Define o preenchimento do parâmetro constante no XML referente ao 'web', onde deve contar o name space para o web do eFCS.

EMPCAB

Permite selecionar dados da empresa ou da filial, na geração de relatórios externos. Informando-se EMPCAB no campo Runtime_Id da tabela runtimes, o procedural executado irá buscar dados da empresa para preencher os campos do cabeçalho. Se o runtime_id estiver NULL, o cabeçalho será impresso como default, ou seja, com dados da filial.

ENDESTAB

Na montagem do Registro tipo B do CAGED, busca os dados de DDD, fone e e-mail da tabela estabelecimentos

ES_ABATE_VLR

Define se na apresentação de verbas de remuneração ao eSocial, o valor a ser apresentado corresponde ao valor em movimentos calculados abatido da soma do mesmo evento em movimentos anteriores.

88 - Componentes (ARH)

No processo de apresentação das rubricas de remuneração ao eSocial, será utilizado a rotina 'GESP8001.abate_soma_mov_anterior', rotina responsável em retornar o valor resultante da operação do valor em movimentos calculados abatido da soma do mesmo evento em movimentos anteriores (remunPerApur).

EXECALC

Permitir determinar o comando que será executado quando o programa de cálculo estiver rodando na modalidade "Remote Shell".

UNIVERSALRH oferece varias opções na forma de execução do programa de cálculo. Uma delas é através do comando remoto.

Quando utilizarmos "remote shell", devemos cadastrar na coluna **Cmd** o comando correspondente:

Runtime = EXECALC, Cmd = (linha de comando de remote shell a ser executado).

Default - sem default

Exemplo:

RUNTIME = EXECALC,

CMD = C:\WINDOWS\RSH -L USER SERVER01 [ORACLE_HOME]

Onde:

C:\WINDOWS\RSH = Endereço do programa **RSH** que é o programa de comunicação para comandos de "remote shell". Este programa é de domínio publico, e encontra-se disponível na internet e na biblioteca do **UNIVERSALRH**.

-L = Parâmetro de execução do RSH. Uso obrigatório.

USER = Nome do usuário (login) no sistema operacional.

SERVER01 = Nome do servidor de aplicação. Mudando este parâmetro, o administrador do site poderá escolher qualquer servidor para executar o programa de cálculo.

ORACLE_HOME = Endereço do diretório onde estão localizadas as bibliotecas do ORACLE. Esta variável deve ser informada quando o servidor do banco de dados utiliza o sistema operacional Unix. Se for o caso, a customização desta variável deverá seguir o seguinte padrão:

```
export ORACLE_HOME=/oracle;
```

Onde:

export = Comando para o sistema operacional que tem por objetivo definir uma variável de ambiente na sessão onde irá rodar o processo.

ORACLE_HOME = Nome da variável que aponta para o diretório onde estão localizadas as bibliotecas do ORACLE.

/oracle = Caminho que define a localização do diretório onde estão localizadas as bibliotecas do ORACLE.

EXECALCBG

Permitir determinar o comando que será executado quando o programa de cálculo estiver rodando na modalidade "background".

O **UNIVERSALRH** oferece varias opções na forma de execução do programa de cálculo. Uma delas é a execução em "background", ou seja, sem provocar o travamento da estação de trabalho do usuário. Nesta forma de execução, devemos cadastrar no campo **Cmd** o

89 - Componentes (ARH)

comando que permite a execução de processos em background relativamente ao sistema operacional usado pelo Cliente.

Exemplo:

RUNTIME = EXECALCBG,

CMD = (linha de comando com o seguinte formato: “*comando complemento*”).

Onde:

comando = Comando do sistema operacional que permite desabilitar o sinal de “hang-up” (não permite que o usuário do client cancele o processo com o comando CTRL+D). Para sistemas operacionais Unix e Linux, o comando é “**nohup**”. Já para o sistema operacional Windows, o comando é “**start**”.

complemento = o literal “<CALCULO> > <LOG> 2>&1 &”

- Comando a ser cadastrado no campo **Cmd** do runtime para sistemas operacionais Unix e Linux :

“**nohup <CALCULO> > <LOG> 2>&1 &**”.

Onde:

<CALCULO> = Será substituído pelo comando que seria executado por “Remote Shell” ou DAEMON.

<LOG> = Identificador do arquivo de log que será gerado pela variável “**comando**”.

2>&1 = Indica ao sistema operacional que deverá ser redirecionada a saída padrão e de erro do sistema operacional para o arquivo indicado em **<LOG>**. Enfatizamos que este arquivo não é o arquivo de log gerado pelo programa de cálculo e sim por “**nohup**”. Este arquivo conterà as mensagens geradas pelo sistema operacional durante a execução do cálculo. O nome deste arquivo é formado da seguinte forma:

GFPNNNN.bg

Onde:

GFP = constante.

NNNN = número da solicitação do cálculo.

.bg = constante.

& = parâmetro para o “**nohup**” que define a execução do processo em modo de background para os sistemas operacionais Unix e Linux.

- Comando a ser cadastrado no campo **Cmd** do runtime para sistema operacional Windows:

“**start /min /B <CALCULO>**”.

Para outros sistemas operacionais, consultar a documentação sobre comandos de execução de processos em background.

FERIASCALC

Permite buscar os valores da tabela MOVIMENTOS_VARIÁVEIS na geração dos arquivos para banco para o **Tipo de Pagamento Férias**.

90 - Componentes (ARH)

GERAINSCRIND

Permite que seja atualizada a tabela INSCRICOES_INDIVIDUAIS.

GFP_ALLOC_TIME

Utilizado pelo programa de cálculo, define o tempo de espera para a obtenção de um código de bloqueio de processo dentro do banco de dados utilizando a função do Oracle DBMS_LOCK.ALLOCATE_UNIQUE. Caso não seja possível obter um código de bloqueio, a execução do programa é abortada. O programa de cálculo monta o código de bloqueio utilizando a sigla GFP + o número da pessoa física.

Este runtime é utilizado quando o **Modo de Cálculo** for "Pessoas Físicas por Seleção", "Registro de Emprego por Seleção" ou "Elo Contrato por Seleção".

Informar no campo CMD o tempo em segundos para este runtime.

GFP_REQUEST_TIME

Utilizado pelo programa de cálculo, define o tempo de espera por uma solicitação de bloqueio do código de bloqueio retornado pela função DBMS_LOCK.ALLOCATE_UNIQUE. O tempo de espera é definido utilizando a função do Oracle DBMS_LOCK.REQUEST. Caso o programa não possa acessar este código de bloqueio, ele permanece aguardando até que este seja liberado.

Este runtime é utilizado quando o **Modo de Cálculo** for "Pessoas Físicas por Seleção", "Registro de Emprego por Seleção" ou "Elo Contrato por Seleção".

Informar no campo CMD o tempo em segundos para este runtime.

GFP_SLEEP_TIME

Utilizado pelo programa de cálculo, define o tempo de espera entre as requisições de bloqueio do código de bloqueio utilizando a função do Oracle DBMS_LOCK.SLEEP. Caso o programa faça a requisição de bloqueio e este estiver em uso por outro processo, o programa irá aguardar o tempo definido neste runtime antes de realizar uma nova requisição.

Este runtime é utilizado quando o **Modo de Cálculo** for "Pessoas Físicas por Seleção", "Registro de Emprego por Seleção" ou "Elo Contrato por Seleção".

Informar no campo CMD o tempo em segundos para este runtime.

INCTPLOGR

Define se o tipo de logradouro será concatenado ao endereço no momento da geração dos arquivos gerados.

INSTORI

A existência deste runtime define que o produto Orientador já está instalado.

KITEVENTO

Este runtime define se os eventos do kit evento já foram inseridos na base do cliente.

91 - Componentes (ARH)

Caso ele não exista, ao usar a tela de inclusão de kit de evento os eventos na base serão todos excluídos e inseridos novamente.

Quando existir os eventos não serão inseridos.

LETRARESTRICAO

Este runtime define uma letra para composição das visões de restrição para ser utilizado na conciliação de usuário. Informar no campo CMD apenas uma letra.

Exemplo:

A

Quando houver a conciliação do usuário serão criadas no usuário proprietário do **UNIVERSALRH** as visões com a seguinte estrutura: A_nomedatabela

Exemplo:

A_REG_EMPREGOS

Com a possibilidade de determinar qual será a nomenclatura das tabelas de restrição, toda a vez que houver modificação na letra que gera a nomenclatura, sugerimos que sejam deletadas as visões criadas no usuário proprietário bem como seus sinônimos criados nos usuários criados pelo **UNIVERSALRH**. Caso o Cliente deseje manter estes objetos no banco não implicará erro na conciliação do usuário.

LOGR_NOME

Este runtime define a forma em que o nome do logradouro será montado para o preenchimento do campo 'NOME' na tabela 'CEP_LOGRADOUROS'.

As informações serão coletadas dos arquivos migrados dos Correios.

Informar no campo CMD conforme uma das opções abaixo:

NOME_COMPLETO - O nome do logradouro será composto pelo tipo e título por extenso, a preposição e o nome oficial do logradouro.

NOME_ABREV - O nome do logradouro será atribuído com o nome abreviado definido pelo DNE.

NOME_OFICIAL - O nome do logradouro será atribuído com o nome oficial definido pelo DNE.

(Este nome não contém tipo, nem título e nem preposição)

TIP_TIT_ABREV - O nome do logradouro será composto com o tipo e título abreviados, a preposição e o nome oficial do logradouro.

TIT_ABREV - O nome do logradouro será composto com título abreviado, a preposição e o nome oficial do logradouro.

(Nesta opção o tipo de logradouro não fará parte do nome do logradouro)

Caso o RUNTIME não esteja informado será adotada a mesma composição definida como TIT_ABREV, onde o tipo do logradouro não compõe o nome.

LOTSEQ

Quando este identificador está definido a transação de Mestre Lotes gera o número do lote automaticamente, caso este não tenha sido digitado.

MOVCALC

Permite selecionar a busca do valor **Maior Remuneração** para os relatórios de rescisão. Se não for cadastrado o Runtime-ID MOVCALC, esta informação virá da tabela REG_SALÁRIOS, se for informada, o report irá buscar o valor na tabela MOVIMENTOS_CALCULADOS, para o evento 1840.

NCLRREGDES

Permitir que o usuário escolha a funcionalidade durante a manutenção do Registro de Reg Valores Designação.

Ao alterar o valor designado, é necessário informar os dados da tabela salarial. Caso o valor informado em seguida não seja exatamente o valor da tabela, as informações referentes às tabelas salariais poderão ser removidas ou não, automaticamente. Ao alterar o registro o usuário estará optando por “não remover automaticamente”.

Runtime = NCLRREGDES

Default - Remover automaticamente.

NCLRREGSAL

Permitir que o usuário escolha a funcionalidade durante a manutenção do Registro de Salário.

Ao alterar o salário de um funcionário, é necessário informar os dados da tabela salarial. Caso o valor informado em seguida não seja exatamente o valor da tabela, as informações referentes às tabelas salariais poderão ser removidas ou não, automaticamente. Ao alterar o registro o usuário estará optando por “não remover automaticamente”.

Runtime = NCLRREGSAL

Default - Remover automaticamente.

NCLRREGVEE

Permitir que o usuário escolha a funcionalidade durante a manutenção do registro de Reg Valores Estabilidade Econômica.

Ao alterar o valor da estabilidade econômica, é necessário informar os dados da tabela salarial. Caso o valor informado em seguida não seja exatamente o valor da tabela, as informações referentes às tabelas salariais poderão ser removidas ou não, automaticamente. Ao alterar o registro o usuário estará optando por “não remover automaticamente”.

Runtime = NCLRREGVEE

Default – Remover automaticamente.

NCTLORC

Quando este identificador está definido não será realizado o controle do orçamento no módulo de Recrutamento e Seleção e de Cargos e Salários.

NFUNC

Permite ao usuário escolher qual a função que será utilizada para definir a regra das atitudes indicadas na lista de valores do campo atitude na transação de Registro de Ocorrências (GAP0410). O nome da função deverá ser colocado no campo CMD para esta variável, se nenhuma for definida o sistema assume a função ATITUDE_INDICADA_1 (GAPP8040).

NOMEIASSVR

Define o nome do computador que executa o servidor de aplicação Oracle OAS para execução de Relatórios

Esse runtime é utilizado pelo package que cria Trigger de Logon de Usuários (ARHP8130).

ORAEXP

Define qual o caminho se encontra o arquivo executável do Export da Oracle 'EXP.EXE'

ORAIMP

Define qual o caminho se encontra o arquivo executável do import da Oracle 'IMP.EXE'

ORDIFRATEIO

Define como será distribuído o valor restante do rateio quando o mesmo não atingiu 100%, realizando umas das seguintes customizações no campo CMD:

- 1 - Coloca o restante do valor, no centro de custo que foi definido com maior proporção;
- 2 - Coloca o restante do valor, no centro de custo que foi definido com menor proporção;
- 3 - Coloca o restante do valor, no centro de custo que foi definido com maior código de centro de custo;
- 4 - Coloca o restante do valor, no centro de custo que foi definido com menor código de centro de custo;
- 5 - Coloca o valor restante em R\$0,01 para todos os maiores códigos de centro de custo envolvidos no rateio até que o valor seja zerado.

PATHARXML

Define o local do sistema de arquivos para a gravação dos arquivos xml.

94 - Componentes (ARH)

Este runtime armazena o nome do diretório criado no banco de dados Oracle. Cabe salientar que o caminho do sistema de arquivos do servidor de aplicação deverá ser definido no serviço do Apache, ou seja, será necessário criar um diretório virtual no OAS HTTP Server que aponte para o diretório definido neste runtime.

PATHCURVIEW

Define o local do sistema de arquivo para gravação do arquivo temporário de currículos. Este runtime armazena o nome do diretório criado no banco de dados Oracle. Cabe salientar que o caminho do sistema de arquivos do servidor de aplicação deverá ser definido no serviço do Apache, ou seja, será necessário criar um diretório virtual no OAS HTTP Server que aponte para o diretório definido neste runtime.

PATHDOCVIEW

Runtime com o nome do diretório temporário aonde o arquivo será extraído do banco de dados e salvo para que seja visualizado.

Obs: Este diretório deve ser criado no servidor de banco de dados. Verifique no Guia de instalação ([ARH4001](#)) os grants necessários para a criação do diretório. Esse diretório deverá estar mapeado no Oracle Directory.

PERAQUISDIAS

Define o número de dias que auxilia na inicialização do novo período aquisitivo.

Quando este runtime estiver definido, o procedural Inicializa Períodos Aquisitivos, soma o número de dias definido no runtime com a data do sistema (sysdate) e compara com a data fim do período aquisitivo, se for maior, será inserido um novo período aquisitivo. O número de dias deve ser preenchido no campo CMD do runtimes.

PERIODO_EXAMES

Define qual o período em meses para a realização de exames.

PFISSEQ

Quando este identificador está definido as transações de Pessoas Físicas (GFP0385), Candidatos (GRS0080), Dependentes (GFP0395) e Entrevistas (GRS0180) passarão a gerar o número automaticamente, caso não tenha sido digitado.

PIPEXML

Define o PIPE de comunicação com o programa GES8160.
Por exemplo: GRAVARXMLESOCIAL.

PRESERVA_LOGS

Define o número de dias que os logs da tabela UNIVERSALRH_LOGS, serão preservados, estes LOGS são gerados na geração dos procedurais e tem sua visualização via o ARH0420

95 - Componentes (ARH)

O número de dias deve ser preenchido no campo CMD do runtimes, caso não informado os LOGS serão preservados por 10 dias.

PRGVERSAOCALC

Informa a nome e a localização do executável do programa de cálculo.

REAJSALCAT

Quando esse runtime estiver cadastrado indica que somente serão reajustadas as categorias salariais vinculadas a cada nível salarial.

REPCLI

Permite definir o caminho onde se encontra o Runtime RWCLI60, que é responsável por transferir por linha de comando os parâmetros passados através da transação (ARH0240) para o Servidor de Relatórios. Para que o usuário obtenha êxito no Agendamento de Relatórios este Runtime deve ser definido antes de utilizar a transação de agendamento (ARH0240).

Exemplo Cliente/Servidor:

Defini-se o caminho onde se encontra o Runtime RWCLI60 na máquina cliente.

CMD = C:\orawin95\BIN\RWCLI60.EXE

Exemplo Web:

Define-se o caminho onde se encontra o Runtime RWCLI60 na máquina definida como Servidor de Relatórios.

CMD = E:\oracle\fsrserver\RWCLI60.EXE

REPCTXDI

Define como será gerado o diretório de gravação de relatórios e/ou procedurais utilizando o conteúdo da variável de contexto.

O usuário deverá cadastrar na coluna CMD quais das seis variáveis de contexto (os dados atualizados do “usuário identificado” encontrados no bloco Variáveis de Contexto na transação Visualizar Contexto (ARH0170) formarão o nome ou diretório de gravação. As variáveis de contexto serão identificadas pelos nomes:

CTX_EMPRESA: Identifica a empresa do funcionário.

CTX_FILIAL: Identifica a filial atual do funcionário.

CTX_CCUSTO: Identifica o centro de custo atual do funcionário.

CTX_PFIS: Identifica o número da pessoa física.

CTX_CHAPA: Identifica a matrícula do funcionário.

CTX_CONTRATO: Identifica o contrato de uma pessoa física.

Exemplo (sintaxe de caminho em formato Windows):

Definindo o diretório de gravação com o conteúdo da variável de contexto

CTX_EMPRESA

CMD = CTX_EMPRESA

96 - Componentes (ARH)

Resultado: C:\TEMP\<>VALOR DA CTX_EMPRESA<><NOME DO RELATORIO> ou
C:\TEMP\<>VALOR DA CTX_EMPRESA<><NOME DO PROCEDURAL>.lis

REPCTXFN

Define como será gerado o nome do relatório ou procedural utilizando o conteúdo da variável de contexto.

O usuário deverá cadastrar na coluna CMD quais das seis variáveis de contexto (os dados atualizados do “usuário identificado” encontrados no bloco Variáveis de Contexto na transação Visualizar Contexto (ARH0170) formarão o nome ou diretório de gravação. As variáveis de contexto serão identificadas pelos nomes:

CTX_EMPRESA: Identifica a empresa do funcionário.

CTX_FILIAL: Identifica a filial atual do funcionário.

CTX_CCUSTO: Identifica o centro de custo atual do funcionário.

CTX_PFIS: Identifica o número da pessoa física.

CTX_CHAPA: Identifica a matrícula do funcionário.

CTX_CONTRATO: Identifica o contrato de uma pessoa física.

CTX_USER: identifica o usuário do banco de dados.

Exemplo (sintaxe de caminho em formato Windows):

Definindo o nome do arquivo de gravação com o conteúdo da variável de contexto
CTX_EMPRESA

CMD = CTX_EMPRESA

Resultado: C:\TEMP\<>VALOR DA CTX_EMPRESA<><NOME DO RELATORIO> ou
C:\TEMP\<>VALOR DA CTX_EMPRESA<><NOME DO PROCEDURAL>.lis

REPDIRETORIO

Define o diretório base de geração do relatório.

Exemplo (sintaxe de caminho em formato Windows):

Definindo o diretório base de gravação do relatório na pasta C:\TEMP do servidor de aplicação

CMD = C:\TEMP

REFMT

Permite a escolha do formato de saída gerado pelo relatório em ambiente web. Deverá ser cadastrado na coluna cmd os seguintes formatos:

- PDF – indica ao Report Server que o arquivo gerado será no formato Adobe Acrobat Reader.
- HTML – indica ao Report Server que o arquivo gerado será no formato HTML.

Este runtime só será utilizado caso o ambiente de produção esteja baseado em Web, ou seja, as transações e os relatórios são disponibilizados aos usuários através de um navegador Web.

Para maiores detalhes sobre configuração do Report Server, consulte o manual do Oracle Developer (Deploying Applications on the Web).

REPSVR

Permite que o usuário defina um Servidor de Relatório padrão, que será utilizado pelo **UNIVERSALRH** no momento em que o usuário agendar um relatório. Para que o usuário obtenha êxito no agendamento este Runtime deve ser definido antes de utilizar a transação de agendamento (ARH0240).

Exemplo:

CMD: Nome do Servidor de Relatórios;

Onde:

Nome do Servidor de Relatórios = REP60SVR

Passos para configuração do Servidor de Relatórios Cliente/Servidor

1. Alterar o arquivo TNSNames.ORA da máquina cliente para apontar para o Servidor de relatórios.

Exemplo:

```
REP60SVR = (ADDRESS=(PROTOCOL=tcp)(HOST=10.0.0.131)(PORT=1949))
```

Onde:

HOST: Endereço IP do Servidor de Relatórios;

PORT: Porta padrão destinada à transmissão dos dados.

2. Definir o Runtime REPSVR na transação Runtimes.

Passos para configuração do Servidor de Relatórios Web

1. Alterar o arquivo TNSNames.ORA da máquina destinada a ser o Servidor de Relatório.

Exemplo:

```
REP60SVR = (ADDRESS=(PROTOCOL=tcp)(HOST=10.0.0.131)(PORT=1949))
```

Onde:

HOST: Endereço IP do Servidor de Relatórios;

PORT: Porta padrão destinada à transmissão dos dados.

2. Definir o Runtime REPSVR na transação Runtimes.

Para maiores informações sobre o Oracle Reports Server, por favor, consulte o endereço <http://www.oracle.com/technology/products/reports/index.html>

REPUSUARIO

Quando cadastrado, o **UNIVERSALRH** usará o nome do usuário de banco de dados logado no sistema, para complementar o diretório base (REPDIRETORIO).

REPWEB

Permite definir a configuração da execução de relatórios em ambiente web. No campo cmd deverá ser cadastrado a URL que identifica o diretório virtual do servidor de aplicação. Poderá ser utilizada a configuração para execução via cartucho ou CGI.

Exemplo da configuração:

- Configuração para Cartucho – Oracle 9iAS Release 1:
`http://servidor_de_aplicação/r60ows?server=nome_do_repserver+paramform=html`
- Configuração para CGI - Oracle 9iAS Release 1:

98 - Componentes (ARH)

`http://servidor_de_aplicação/cgi-bin/rwcgi60.exe?server=nome_do_repserver+paramform=html`

- Configuração para servlet - Oracle 9iAS Release 1:
`http://servidor_de_aplicação/reports/rwervlet?mode=default+paramform=html`
- Configuração para servlet - Oracle 9iAS Release 2 ou OAS 10g:
`http://servidor_de_aplicação/reports/rwervlet?mode=default+paramform=yes`

Caso o cliente esteja usando o Servidor de Aplicação Oracle 9iAS Release 1, o parâmetro **paramform** deverá ter o ser valor definido como **HTML**. Caso o servidor seja o Oracle 9iAS Release 2 ou OAS 10g, o parâmetro **paramform** deverá ter o valor definido como **YES**.

Este runtime só será utilizado caso o ambiente de produção esteja baseado em Web, ou seja, as transações e os relatórios são disponibilizados aos usuários através de um navegador Web.

Para maiores detalhes sobre configuração do Report Server, consulte o manual do Oracle Developer (Deploying Applications on the Web).

REQSEQ

Quando este identificador está definido a transação de Requisições (GRS0050) passará a gerar o código da requisição automaticamente, caso não tenha sido digitado.

Este Runtime é necessário para utilização da configuração por termo em relatórios.

RWCONV

Indica a localização do utilitário `rwconverter.exe`, deverá ser informado no campo **Cmd** qual o diretório, caminho completo, do aplicativo. Ajustar a variável de ambiente `REPORTS_PATH`, no servidor do OAS, para ter uma entrada para o diretório de trabalho da seção de configuração definida para rodar o sistema.

SALDOFERIAS

Permite definir qual o critério de proporcionalidade para cálculo do saldo de férias.

O sistema originalmente usa critérios e tabelas diferentes para o cálculo de Saldo de Férias, dependendo da Categoria Salarial. Caso a empresa possua funcionários mensalistas e horistas, porém trate todos com o mesmo critério, este runtime informará ao sistema o critério de calculo de saldo de férias que deve ser adotado para todos os funcionários, independentemente da Categoria Salarial, podendo ser indicado se mensalista ou horista.

Cadastrar na coluna `cmd` os seguintes formatos:

- `MENSALISTA` – indica que o critério padrão de proporcionalidade será a mensalista.
- `HORISTA` – indica que o critério padrão de proporcionalidade será a horista.

SEFIPDTCOMP

Define a geração do registro tipo 32 de Movimentação do Trabalhador considerando a data de competência do `REG_RESCISOES` para os códigos de Movimentação de Rescisão.

Default – considera a data de início de vigência da tabela de `REG_AFASTAMENTOS`.

SELECAO_CHECKUP

99 - Componentes (ARH)

Permite selecionar qual a modalidade de Visita Médica do Tipo Checkup será executada na geração do procedural que faz a Seleção para Solicitações de Exames e Esperas de Visitas Médicas. Deverá ser informado no campo CMD:

- Para a modalidade Checkup por Tipo de Visita Médica: **1**
- Para a modalidade Checkup por Emissão de ASO: **2**

SELECAOPFIS

Quando a forma do processo de uma solicitação de cálculo for “R. Emprego seleção <p. física>”, “R. Emprego seleção <matrícula>”, “P. Física seleção <número>”, “E. Contrato seleção <p. física>” ou “E. Contrato seleção <contrato>”, a solicitação de cálculo utilizará o procedural cadastrado neste runtime para realizar a seleção de pessoas físicas que devem ter seus eventos calculados através de uma solicitação de cálculo.

O usuário poderá desenvolver uma procedure que recebe como parâmetro o número da solicitação de cálculo e poderá montar um modo de seleção de pessoas físicas para cada uma das formas de processo citadas acima. O usuário poderá utilizar como base o programa Seleção de pessoas físicas (GPRP8110).

Runtime = SELECAOPFIS, Username = (opcional), Cmd = (informar o nome da procedure), Password = (opcional)

SISTEMA_MENSAGERIA

Define a sigla do sistema de mensageria que está sendo usada UBI ou UBS.

SISTOPERSEPARA

Define qual o separador deverá ser utilizado na definição de diretórios para a geração de relatórios ou procedurais conforme sistema operacional do Cliente. Informar no campo CMD barra (/) ou contrabarra (\). Por default, será utilizado contrabarra (\) formato Windows.

SQLLOAD

Permitir a escolha do nome do executável do **SQL*Loader**.

Em virtude de estarem disponíveis várias versões do Oracle, os Clientes podem determinar que versão do Sql*Loader deseja executar, pois o nome deste programa tem sido alterado pela Oracle em versões diferentes do Banco de Dados. Outra função é a de trocar o usuário em tempo de execução, dando autoridade diferente para o **UNIVERSALRH** ao executar um comando que o usuário solicitante não possua.

Runtime = SQLLOAD, Username = (opcional), Cmd = (nome do executável), Password = (opcional)

Default – SQLLDR80.EXE com o usuário solicitante.

SQLPLUS

Permitir a escolha do nome do executável do **SQL*Plus**.

Em virtude de estarem disponíveis várias versões do Oracle, os Clientes podem determinar que versão do Sql*Plus deseja executar, pois o nome deste programa tem sido alterado

100 - Componentes (ARH)

pela Oracle em versão diferentes do Banco de Dados. Outra função é a de trocar o usuário em tempo de execução, dando autoridade diferente para o **UNIVERSALRH** ao executar um comando que o usuário solicitante não possua.

Runtime = SQLPLUS, Username = (opcional), Cmd = (nome do executável), Password = (opcional)

Default – PLUS80W.EXE com o usuário solicitante.

TABSPCDEF

Serve para indicar a Tablespace Default.

TABSPCTMP

Serve para indicar a Tablespace Temporária.

TESTACARGO

Este runtime quando cadastrado, indica que o trigger que validar a inserção de um funcionário em um cargo (GFPT0280) deve ser disparado.

TIPO_AUTENTICACAO

Este runtime define o tipo de autenticação a ser utilizado quando preenchido com 'AVANÇADO' utiliza a autenticação Avançada, quando preenchimento com 'INTERMEDIARIO' usa autenticação intermediária.

TRANSFEXT

Este runtime permite cadastrar as extensões desejadas para a transferência de arquivos (GPRE8470).

Quando não existir, somente serão permitidas as transferências dos arquivos GPR84*.XML, GPR84*.DAT e GPR84*.LOG.

Se cadastrado deverá ser preenchido usando a expressão regular '.XML\$|.LIS\$|.LOG\$'.

UBS_JDBCDRIVERNAME

Informar o driver para conexão com a base de dados. Para oracle informar oracle.jdbc.driver.OracleDriver

UNLOCKTEMP

Permite a escolha de lock de tabela temporária.

Este runtime permite desativar a opção de lock das tabelas temporárias geradas pelos relatórios, ou dos procedurais que também utilizam este recurso para processamento. Por default todos os relatórios executam lock de tabela temporária (Ex.: GFP000T1) para garantir exclusão mútua destas tabelas em tempo de execução. Com isso o **UNIVERSALRH** garante que dois usuários diferentes não executem o mesmo relatório em

101 - Componentes (ARH)

estações diferentes. Para desabilitar esta funcionalidade basta cadastrar este identificador em Runtimes.

Runtime = UNLOCKTEMP.

UORG_FISICA

Define que o endereço válido para as unidades organizacionais é o endereço do seu próprio cadastro. Quando não informado, o endereço das unidades organizacionais não será considerado.

URLCURVIEW

Define a URL de localização do documento de currículo no servidor de aplicação.

URLDOCVIEW

Runtime com o nome do diretório temporário aonde o arquivo será salvo para que seja visualizado.

Informar a url contendo o endereço do servidor de aplicação seguido do alias definido em HTTPD.config. Deve apontar para o mesmo diretório informado em PATHDOCVIEW.

Obs: Após configurar o HTTPD.config deverá ser feito restart no servidor de aplicação.

VALIDA_DT_REFERENCIA

Este runtime é responsável pela validação da data de indicação de retroação. Caso a data de indicação tenha um "envelhecimento" superior ao indicado pelo runtime, será levantada uma exceção no trigger de monitoramento, evitando que ocorra perda de registros que devem ser monitorados.

VLRREMUNERACAO

Permite selecionar no Registro de Salário o valor da remuneração.

WALLETDIR

Informa a localização do arquivo wallet, no sistema de arquivos do servidor de banco de dados Oracle. O usuário deverá cadastrar, na coluna CMD, com a seguinte sintaxe:

file:[diretório de localização do wallet].

Exemplo para banco de dados rodando em UNIX/Linux:

file:/app/oracle/secure

WALLETPWD

Informa a senha para abertura do arquivo wallet. Essa senha é definida pelo administrador de banco de dados, no momento da criação do arquivo wallet. A senha definida para o arquivo wallet deve ter no máximo 10 caracteres, combinando letras e números somente.

Não é permitido o uso de caracteres especiais, tais como @, &, \$, #, !, *, entre outros.

102 - Componentes (ARH)

WSINTEGRAMENUS

Identificar o nome do webservice de integração do URH com a mensageria Universal Bussiness Interface.

WSSAIDA

Indica o nome do recurso para identificação do WS ou Table API para envio.

No campo cmd informar o nome do recurso conforme cadastro na tabela ESOCIAL_RECURSOS_INTEGRACAO.

ZERADIASPROPNEG

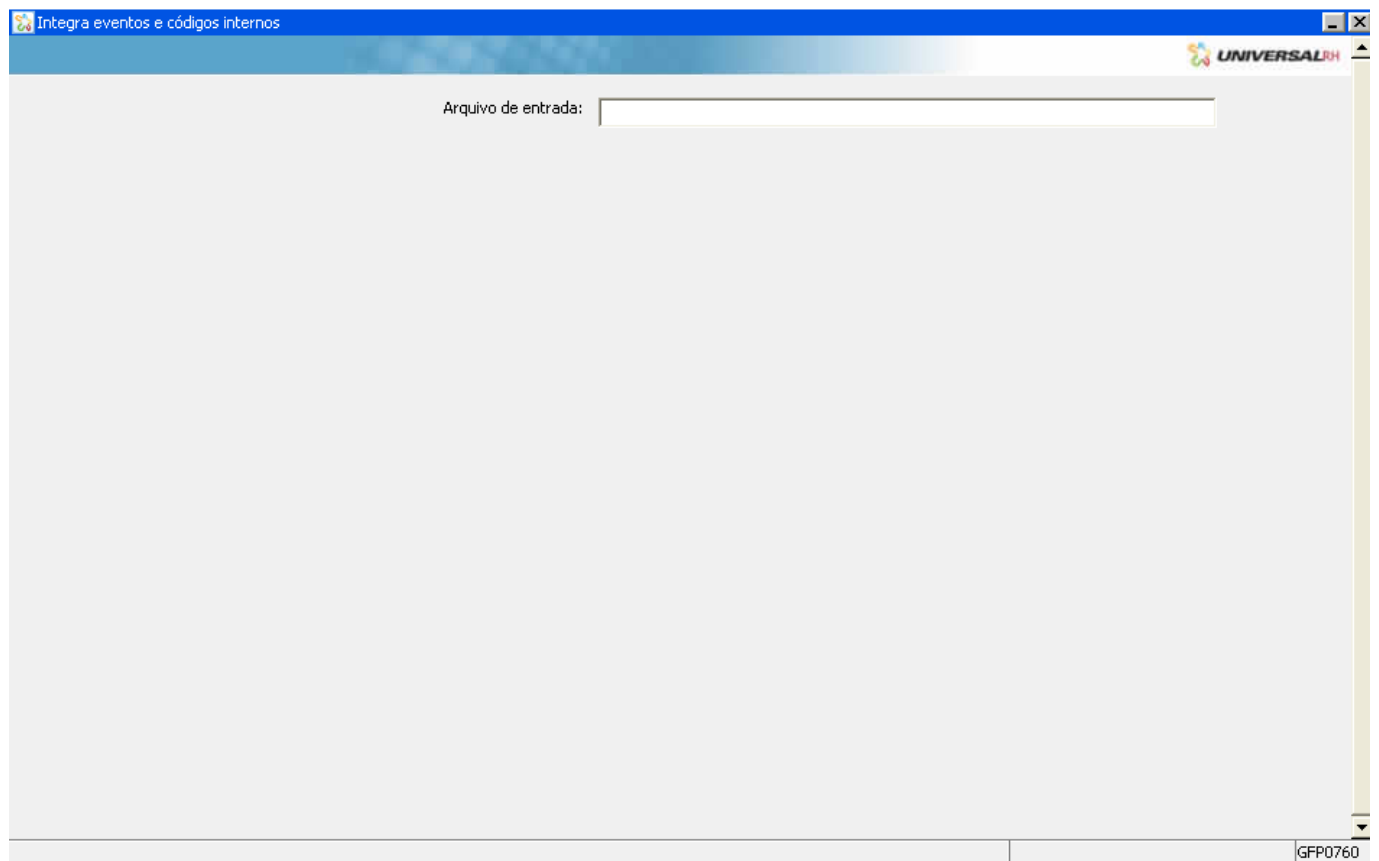
Zera o campo saldo de férias proporcional (SDO_FER_PROV) quando o campo número de dias de férias por afastamento (DIAS_PE_AFAST) for maior que zero e isso resultar num saldo de férias proporcional negativo.

108 - Administração de Pessoal (GFP)

GFP0760 – Integra eventos e códigos internos

O objetivo do programa é ler o arquivo GFPEICI.DAT e atualizar as tabelas de eventos e códigos internos. Ao ser lido pela transação, serão incluídos todos os códigos inexistentes nas tabelas do cliente. Os demais códigos já existentes terão a sua inclusão rejeitada automaticamente.

Integração eventos e códigos internos



GPR0590 – Execução de cálculo

Vimos no capítulo Solicitação de Cálculo, que antes de executarmos qualquer procedimento era necessário efetuarmos a solicitação, onde as informações eram essenciais para a execução. Porém ao terminarmos a solicitação o cálculo não é executado. Isto porque para executá-lo é necessário pedir a execução.

Quando o usuário solicitar a execução de cálculo, serão efetuados tantos processos quantas solicitações estiverem gravadas no Banco de Dados, e ainda não tenham sido processadas. Em outras palavras, é possível solicitar vários cálculos e dispará-los com um único pedido.

Os cálculos serão executados na mesma ordem que foram solicitados.

Não será possível disparar o cálculo enquanto existir uma montagem de procedimentos em execução.

A execução de cálculo permite o disparo de vários processos, os quais podem ser executados em paralelo se assim o usuário desejar, porém esta execução somente é realizada quando o modo de cálculo for por Seleção. O programa de cálculo a partir da versão 8042, realiza a execução de cálculos em paralelo analisando a tabela `CONTROLE_CALCULOS`. Por exemplo, a mesma pessoa física sendo executada por mais de um processo ao mesmo tempo. O programa faz a inserção da pessoa física na tabela `CONTROLE_CALCULOS` e começa a execução do processo. Se outro processo precisar executar a mesma pessoa física este será colocado em ESPERA até que finalize a execução do processo anterior.

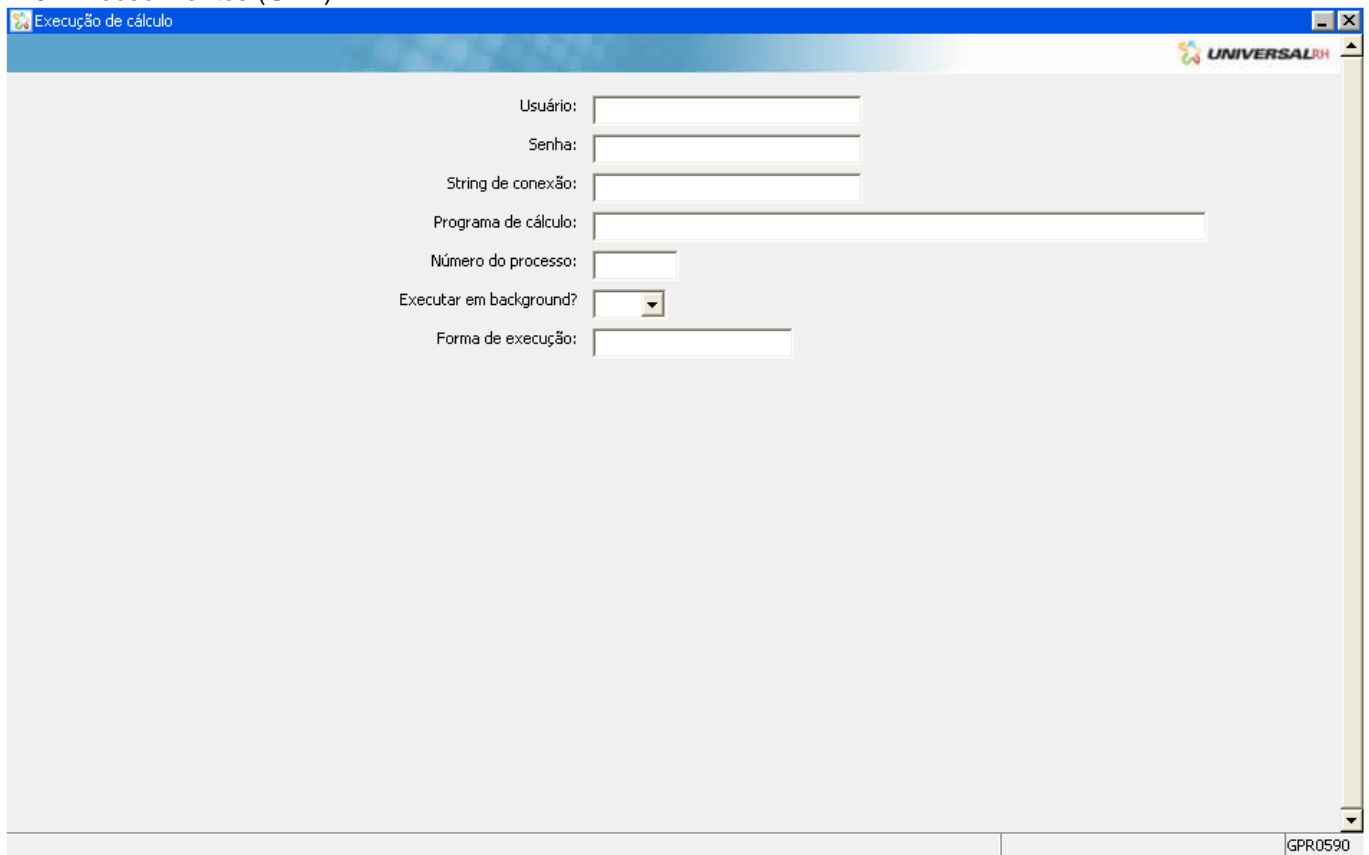
São utilizados como default os seguintes tempos para execução do cálculo em paralelo:

- requisição de um código de bloqueio de processo – 1 segundo ou utilização do runtime [GFP_ALLOC_TIME](#);
- espera por uma solicitação de bloqueio do código de processo – 5 segundos ou utilização do runtime [GFP_REQUEST_TIME](#);
- espera entre intervalos de requisições de bloqueio de processos – 300 segundos ou utilização do runtime [GFP_SLEEP_TIME](#).

Para definição o diretório de geração dos logs, customizar o Runtime [DSTLOGCALC](#).

Execução de Cálculo

110 - Procedimentos (GPR)



Número do processo:

Campo destinado a informar o número do processo caso deseje executar uma solicitação específica. Se nada for informado, irão ser executadas todas as Solicitações de Cálculo que possuírem “S” no campo PROCESSA GENÉRICO.

Executar em background?

Para ambientes cujo servidor de Banco de Dados possui um Sistema Operacional Multiusuário, o processo pode ser disparado no modo “background”, ou seja, liberando o terminal ou a estação de trabalho.

Obs.: Serão sempre gravados arquivos que registram todas as ocorrências durante a execução de cálculo. Os arquivos de LOG são identificados com o Número do Processo que está contido na Solicitação de Cálculo. Caso o arquivo de LOG não seja gerado por problemas de permissões de gravação, o programa de cálculo continuará o processo de execução normalmente e no Status de Execução será atualizado com a mensagem de término, no final da execução, com a palavra **NoLog**.

Exemplo: GFP_151.log, onde o número 151 identifica a solicitação de cálculo.

Para maiores informações sobre como compilar o programa de cálculo e customizações para sua execução consulte o [Guia de Instalação](#).

GPRP8400 – Monitor retroativos

Código do Programa : GPRP8400
Forms de Parâmetro :
Package : GPRP8400
Linguagem : PL/SQL
Título do Programa : Cria monitor retroativos
Específico :

Objetivo do Programa:

O objetivo do programa é criar triggers de banco de dados para realizar o monitoramento de tabelas que podem gerar a necessidade de cálculos retroativos. O usuário deverá usar a transação Monitor Retroativos ([GPR0310](#)) para informar ao **UNIVERSALRH** quais tabelas deverão ser monitoradas pelo sistema. Recomendamos a leitura da documentação de Runtimes, Runtime [DIASRFIR](#), e da transação Data Indicação Retroativos ([GPR0370](#)), a fim obter informações de como configurar o **UNIVERSALRH** no controle do monitoramento de retroativos.

Complementações Necessárias:(O usuário deverá implementar)

Critério de Ordenação, Quebras, outras particularidades:

GPR0310 – Monitor retroativo

Transação destinada a conter as tabelas e colunas que devem ser monitoradas para processo retroativo. O usuário deverá definir quais as tabelas do **UNIVERSALRH** que deverão ser monitoradas para sinalizar a geração de ocorrências que são passíveis de retroação.

Obs.:

Somente objetos do proprietário do **UNIVERSALRH** podem ser monitorados.

Ao gravar o Monitor Retroativo, será gerado um *trigger* de banco de dados que fará o monitoramento da inserção de registros na tabela indicada nesta transação. Além disso, o usuário deverá observar os seguintes critérios adotados para a realização do monitoramento:

- A informação da Data de Indicação de Retroativos ([GPR0370](#)) deve estar válida em relação ao número de dias indicado pelo runtime [DIASRFIR](#), e;
- Serão registrados em Registros Retroativos ([GPR0320](#)), todos os registros cujas colunas datas, indicadas nesta transação, tenham o seu valor menor ou igual à Data de Indicação de Retroativos ([GPR0370](#)).

Se o número da quantidade de dias, obtido pela fórmula **DATA ATUAL DO SISTEMA** (*SYSDATE*) **menos** a Data de Indicação de Retroativos ([GPR0370](#)), for maior que o número de dias indicado pelo runtime [DIASRFIR](#), então será apresentada uma mensagem de erro para o usuário, informando que o monitor de retroativos não permitirá a gravação do registro na tabela monitorada, até que seja ajustada a Data de Indicação de Retroativos ([GPR0370](#)). Desta forma, o **UNIVERSALRH** garante que não sejam perdidas informações sobre movimentos que são passíveis de retroação, em função da defasagem da Data de Indicação de Retroativos ([GPR0370](#)).

Monitor retroativos

Bloco destinado à manutenção de monitoramento retroativo.

O campo **Nome da tabela** apresenta uma lista de valores com os objetos do tipo tabela, visão e sinônimo do usuário proprietário do **UNIVERSALRH** cadastrado na transação Proprietários Sistemas ([ARH0340](#)).

Coluna monitor retroativos

Bloco destinado à manutenção das colunas que compõem o monitoramento de uma tabela.

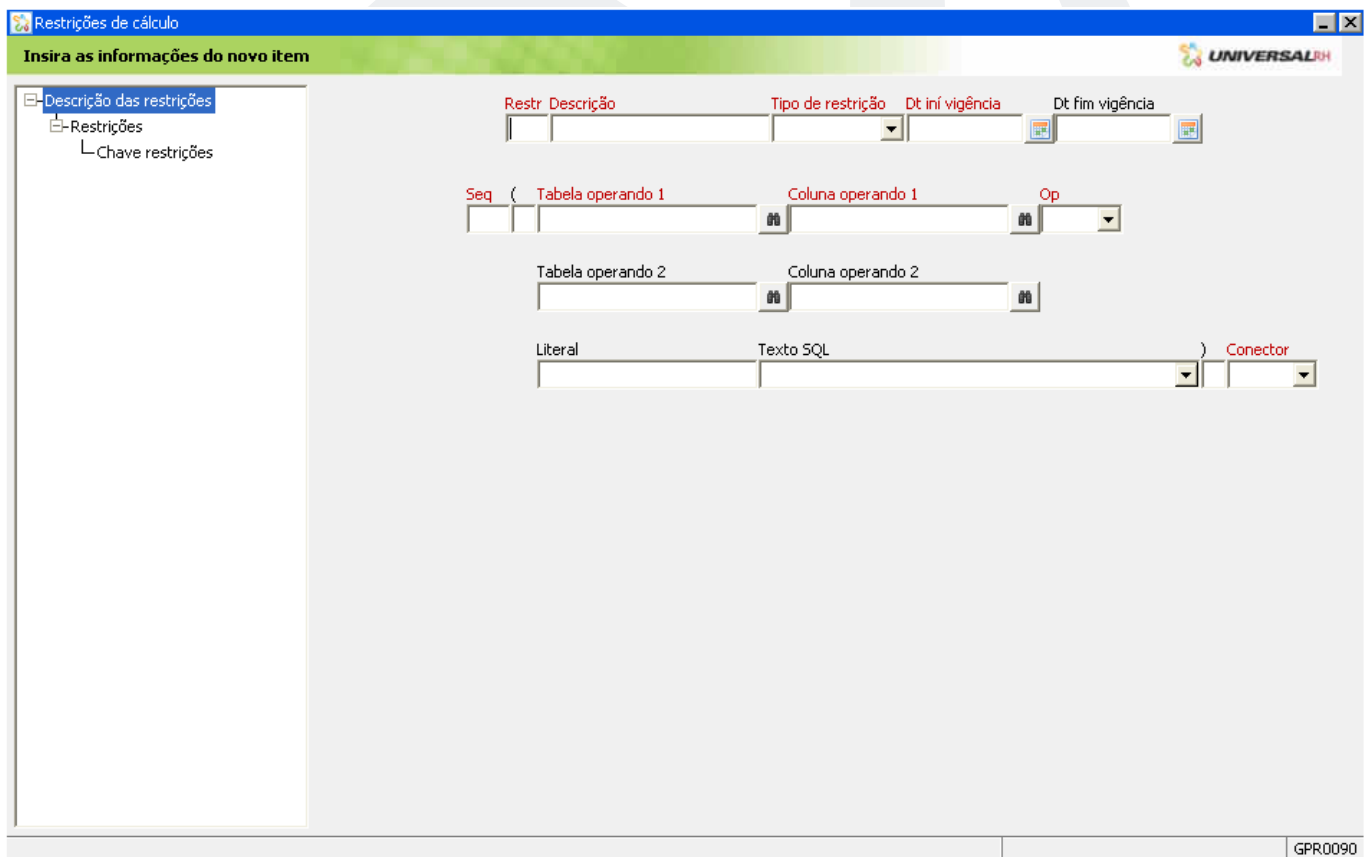
GPR0090 – Restrições de cálculo

Qualquer definição de cálculo pode receber uma Restrição permanente ou temporária. A restrição é na realidade uma super condição, ou seja uma condição que é colocada acima de todas as condições existentes em um cálculo. A construção de uma restrição pode ser baseada em qualquer dado do Banco de Dados que possa ser relacionado aos dados do funcionário. Além disso, uma Restrição pode ser definida para ser utilizada pelo processo de contabilização, sendo usado, deste modo, como uma ferramenta poderosa para construir regras de Lançamentos Contábeis ([1010510](#)).

Os campos que selecionam tabelas apresentam uma lista de valores com os objetos do tipo tabela, visão e sinônimo dos usuários cadastrados na transação Proprietários Sistemas ([ARH0340](#)), onde dentre eles deve estar cadastrado o usuário proprietário do **UNIVERSALRH**.

Descrição das restrições

Bloco destinado ao registro das descrições das restrições.



Observações:

117 - Procedimentos (GPR)

Quando uma restrição (Restr) for do tipo Contábil, o texto do campo Amb, no bloco Chave Restrições, será modificado para *Campo Contábil* e a lista de valores desse campo será modificada para apresentar a variáveis contábeis.

Uma restrição de cálculo somente será executada se sua **Data de início de vigência** for válida, ou seja, estiver ativa quando solicitada a sua execução.

Uma restrição de cálculo será executado se sua **Data de fim de vigência** for válida, ou seja, estiver ativa quando solicitada a sua execução. Em caso de não preenchimento, indica a validade constante a partir da data de início de validade.

Restrições

No campo sequência (**Seq**) recomenda-se numerar inicialmente de 5 em 5. Uma restrição pode ser construída por várias condições. Cada uma delas deve ser numerada em ordem crescente, inversamente a sua prioridade. A sequência menor será a primeira condição.

O campo abre parêntese "(" é usado para definir prioridade de execução. Assim como nas expressões algébricas, as restrições podem possuir prioridade na execução do comando de restrição.

Para construir uma restrição é necessário identificar a tabela com a qual será feita a restrição.

No campo **Op** (operador relacional) informar qual o operador relacional deve ser usado para que seja feita a relação entre o "Campo 1" e ou "Campo 2" ou o "Literal".

Para construir uma restrição pode ser necessário identificar uma segunda tabela a qual pode estabelecer a condição.

Ao construir uma Restrição, o valor contido na Coluna Operando 1, pode ser comparado com o campo Literal, que é um valor absoluto determinado pelo usuário. Observe que deverá ser informado o Literal somente se não houver sido usado nem a Tabela Operando 2 nem a Coluna Operando 2 nesta sequência (neste passo).

Quando o campo **Texto SQL** for preenchido, ele fará com que as informações cadastradas nas outras colunas, dessa transação, sejam desconsideradas pelo montador de Restrição de Cálculo. Ou seja, o montador fará a transformação apenas deste campo.

A construção de uma restrição através deste campo deve obedecer a sintaxe de um comando SQL e o uso de campos da AMB ou variáveis contábeis devem ser precedidas de ":" (caractere dois pontos).

Por exemplo:

```
SELECT 0 FROM MOVIMENTOS_CALCULADOS WHERE REMP_EMP_COD = :EMPRESA AND  
REMP_CHAPA = :CHAPA AND EVEN_COD = :EVENTO
```

O campo **Conector** efetua a conexão entre uma sequência e a seguinte. Seus valores podem ser "AND" ou "OR". (E ou OU). Para preenchimento automático o valor é "AND".

119 - Procedimentos (GPR)

O texto do **Campo amb** pode ser chaveado para *Campo Amb* ou *Campo Contábil* e a lista de valores também serão modificados para apresentar as variáveis da Amb ou as variáveis Contábeis, dependendo do valor do campo.

No campo **Valor** não poderá ser informado quando for preenchida a coluna “Campo Amb”.

O campo conector (**Cnt**) efetua a conexão entre uma sequência e a seguinte. Seus valores podem ser “AND” ou “OR”. (E ou OU). Para preenchimento automático o valor é “AND”.

Variáveis Contábeis

As variáveis contábeis são um conjunto de variáveis disponíveis para construção de Restrições de Cálculo do tipo Contábil. De forma semelhante a Amb, as variáveis contábeis são atualizadas com os valores dos parâmetros de execução da Contabilização e, também, das informações recuperadas do banco de dados em tempo de execução. Estas informações estão relacionadas com centro de custo, empresa, matrícula, filial entre outras, do funcionário que está sendo processado pelo procedural Gera Movimento para Contabilidade ([I01P8600](#)). A seguir, apresentamos o conjunto de variáveis disponíveis para criar Restrições de Cálculo para Contabilização:

Variáveis provenientes dos parâmetros de execução da Contabilização

- CT_PA_EMPRESA - CODIGO DA EMPRESA
- CT_PA_FILIAL_INI - CODIGO DA FILIAL INICIAL
- CT_PA_FILIAL_FIM - CODIGO DA FILIAL FINAL
- CT_PA_DT_REF_INI - DATA DE REFERENCIA INICIAL
- CT_PA_DT_REF_FIM - DATA DE REFERENCIA FINAL
- CT_PA_DT_LANC - DATA DE LANCAMENTO
- CT_PA_MOEDA - CODIGO DA MOEDA

Variáveis provenientes do processamento da Contabilização

- CT_RE_MATRICULA - CODIGO DA MATRICULA FUNCIONAL
- CT_RE_UNI_ORG - CODIGO DA UNIDADE ORGANIZACIONAL DO FUNCIONARIO
- CT_RE_FILIAL - CODIGO DA FILIAL DO FUNCIONARIO
- CT_RE_EST_ORG - CODIGO DA ESTRUTURA ORGANIZACIONAL DO FUNCIONARIO
- CT_RE_MDO - CODIGO DA MAO-DE-OBRA DO FUNCIONARIO
- CT_MC_EVENTO - CODIGO DO EVENTO
- CT_MC_DT_COMP - DATA DE COMPETENCIA
- CT_MC_DT_PAG - DATA DE PAGAMENTO

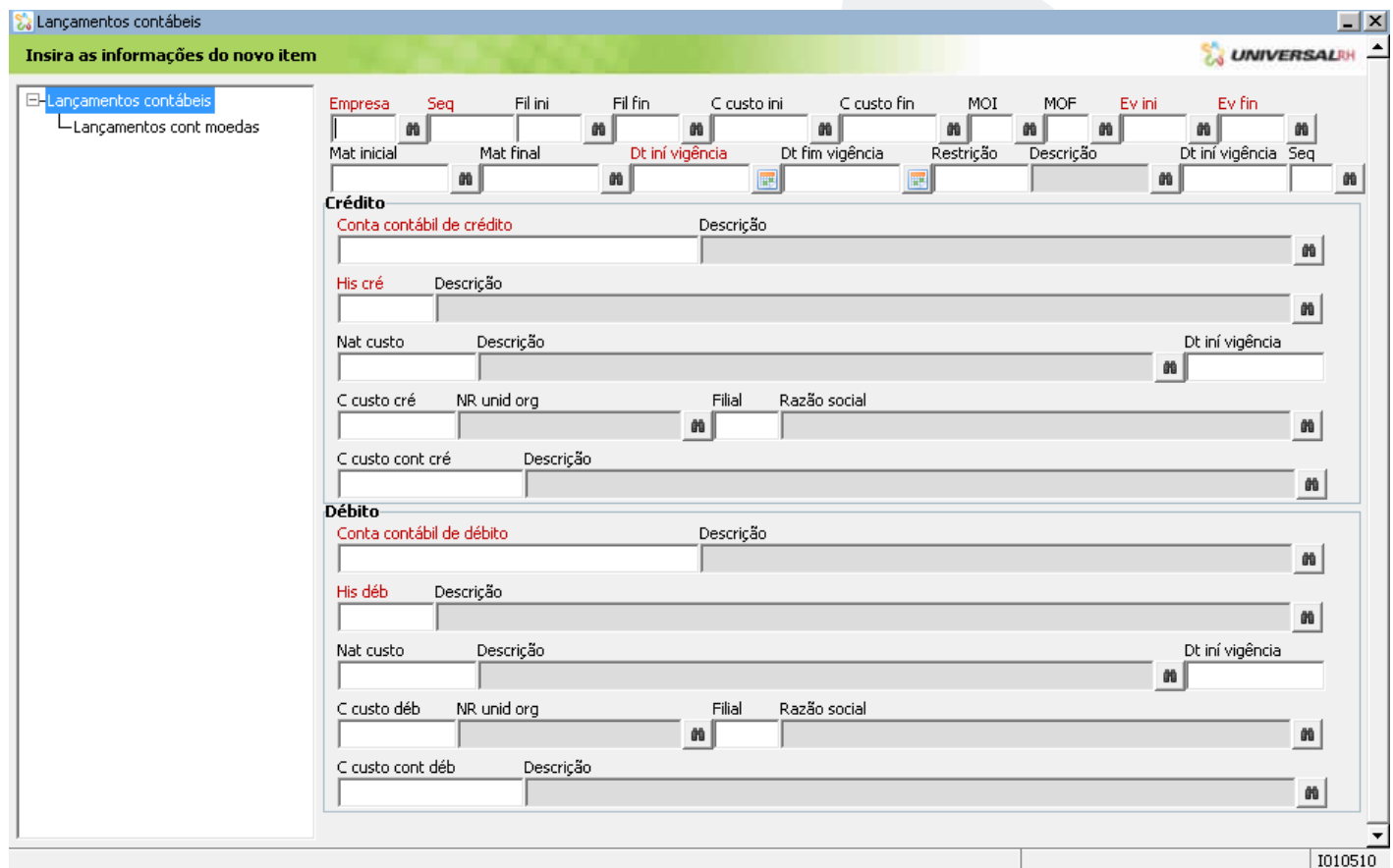
I010510 – Lançamentos contábeis

Transação destinada à manutenção dos lançamentos contábeis, associando para cada evento as suas contas contábeis. O campo código da condição restritiva indica se o lançamento é ou não para ser enquadrado, na montagem da condição o usuário deverá utilizar para a chave as variáveis destinadas à contabilização para fazer associação com os campos da chave do movimento calculado. As definições destas variáveis estão disponíveis no tutorial do sistema de Procedimentos, na transação Restrições de Cálculo ([GPR0090](#)).

Lançamentos contábeis

Transação destinada à manutenção dos lançamentos contábeis.

A lista de valores do campo **Restrição** mostra somente as Restrições de Cálculo definidas como tipo Contábil.



Lançamentos contábeis
 Insira as informações do novo item

Lançamentos contábeis
 Lançamentos cont moedas

Empresa Seq Fil ini Fil fin C custo ini C custo fin MOI MOF Ev ini Ev fin
 Mat inicial Mat final Dt iní vigência Dt fim vigência Restrição Descrição Dt iní vigência Seq

Crédito
 Conta contábil de crédito Descrição
 His cré Descrição
 Nat custo Descrição Dt iní vigência
 C custo cré NR unid org Filial Razão social
 C custo cont cré Descrição


Débito
 Conta contábil de débito Descrição
 His déb Descrição
 Nat custo Descrição Dt iní vigência
 C custo déb NR unid org Filial Razão social
 C custo cont déb Descrição

I010510

121 - Integração Contabilidade (I01)

Lançamentos cont moedas

Bloco destinado ao registro dos lançamentos contábeis em outras moedas.

 Lançamentos contábeis

Insira as informações do novo item

Empresa Seq Ev ini Dt ini vigência

Moeda Sigla da moeda Nome ND NM Posição Ajuste

Crédito

Conta contábil de crédito Descrição

His cr Descrição

Nat custo Descrição Dt ini vigência

C custo cré NR unid org Filial Razão social

C custo cont cré Descrição

Débito

Conta contábil de débito Descrição

His dé Descrição

Nat custo Descrição Dt ini vigência

C custo déb NR unid org Filial Razão social

C custo cont déb Descrição

I010510