

Abertura do módulo GA

SinacorWeb > Gestor de Ativos > Opção de Movimento (Lateral superior direito da tela) > Abertura

Função de Acesso:

1. Ao clicar no botão de "Abertura", é chamada a api [http://\[servidor\]:25149/Custodia/api/EstadoModulo/Abertura](http://[servidor]:25149/Custodia/api/EstadoModulo/Abertura) passando como parâmetro a próxima data útil e o login do usuários.
 - a. Logs: \SYSTEM\WebApi\Controllers\Custodia\EstadoModuloController\Abertura
2. A API realiza algumas validações: (Logs: \SYSTEM\WebApi\Controllers\Custodia\EstadoModuloController\Abertura) (classe ... \Sinacor\code\NET\CF\VC\Source\Common\BusinessLogic\Fila\FilaSolicitacaoViradaDia.cs método Solicitar)
 - a. Valida se o movimento está fechado TCFPARAMETRO.IND_STAT = 'F' ;
 - b. Verifica se foi feita uma solicitação de abertura por outro usuário TCFPARAMETRO.IND_STAT = 'E';
 - c. Se o usuário informado na API existe no cadastro de usuários do Sinacor (GAS);
 - d. Verifica se o módulo já está aberto para a data informada TCFPARAMETRO.IND_STAT = 'A' ;
 - e. Verifica se o calendário de movimento do GA está válido para a data informada TCFCAL (Dias para calendário é parametrizável no Informações de Apoio > Cadastros > Eventos > Sistema "Faturamento Bolsa" campo "Projeção Saldo Físico")
 - i. É possível remontar o calendário pelo SinacorWeb > Gestor de Ativos > Opção de Movimento (Lateral superior direito da tela) > Remonta Calendário
 - f. Verifica se data de abertura é um feriado regional TGEFERIADO (Informações de Apoio > Cadastros > Feriados > Sistema = CF opção Regional)
 - i. Se for um feriado Regional, o sistema continua o processo de abertura sem validar a importação do PAPT (pois não é gerado pelo CAP), para continuar com processos que são executados em feriado regional, como por exemplo, negociação e liquidação de Tesouro Direto;
 - g. Verifica se o arquivo de títulos PAPT foi importado para a data de referência (TCFCTRL_CUST_FIDU.NOME_MSG = 'PAPT') ;
 - i. Ver mais: [Importação PAPT](#)
3. Após validações acima, a API envia mensagem rabbitmq para a APE - BVMF Custodia Fiduciária - Processos, plugin automático Módulo;
4. Inicia o processo de abertura limpando as tabelas de erros TCFRETE_VIRADA_DIA e TCFERRO onde DESC_ORI = 'VIRADA DE DIA';
5. Chama procedure spCf_Abertura.spCf_Abertura executando os processos da abertura passando como parâmetro o código do usuário;
 - a. Qualquer erro no processo, é gravado na TCFERRO e indicado no próprio botão da solicitação;



6. Sistema valida se a data retornada na procedure anterior é igual a data indicada da API, caso for diferente, retorna erro "Processo calculou a abertura para a data de {dataMovimentoAtual:dd/MM/yyyy}. Data solicitada foi de {parametros.DataMovimento:dd/MM/yyyy}."
7. Após abertura realizada com sucesso, é enviada uma mensagem rabbitmq para o tratamento dos arquivos de conciliação de posição (movimentação e saldo dos respectivos mercados);

- **Integração de operações de BMF:**

- Valida se o módulo de Faturamento BMF está instalado no participante:

```
select count(*)
from tgesinst a, tgeempr b
where a.cd_sistema = 'BMF'
and b.cd_empresa = a.cd_empresa
and b.cd_usua_orac = user
and instr( a.vs_sist_inst, 'D' ) = 0;
```

- Valida se o parâmetro de integração com o BMF está ligado em SinacorWeb > Cadastros > Parametrização > Aba Integrações > "BM&F - Recebe Integração do Módulo de Faturamento BM&F": Tcfparametro.ind_rcbm_intr_bmf

The screenshot shows the 'PARAMETRIZAÇÕES' screen in SinacorWeb. Under the 'Integrações' tab, the 'BM&F' integration is enabled (toggle on). Below it, the 'Solicitações' section shows 'Integra Termo' also enabled. At the bottom, there are input fields for 'LA - D+1' with values 2 and 1.

- Consulta posição atual dos clientes no módulo Faturamento BM&F que estejam com os ativos validos para o pregão e as movimentações com resultado final do calculo financeiro dos clientes (Nota de corretagem) (Consulta posição de ouro disponível (OZ1) que não é mais controlado pelo RTC - Derivativos, e sim pela depositária diretamente no Gestor de Ativos):

```
select a.dt_datmov datpre, a.cd_agecom agecom, a.cd_commod commod,
       a.cd_serie serie, s.dt_venc datvct, a.cd_cliente codcli,
       a.cd_mercad mercad, s.pr_exerc preexe, a.qt_dia_atu qtdata,
       c.dtultop, a.vl_ajupos_acu ajuacu
from tmfposic a, tscclibmf c, tmfserie s
where a.dt_datmov = 'Data atual do sistema'
and a.cd_cliente = c.codcli
and a.cd_commod = s.cd_commod
and a.cd_mercad = s.cd_mercad
and a.cd_serie = s.cd_serie
and a.dt_datmov between s.DT_INI_SERIE and s.DT_FIM_SERIE
order by a.cd_cliente;
```

```
select a.datpre, a.agecom, a.codcli, a.commod, a.mercad,
       a.serie, a.datvct, a.numneg, a.natope, a.qtdneg,
       a.tipneg, a.preuni, a.exeopc, a.valneg, a.valdes,
       a.valaju
from mfimpost a,
       tscclibmf b
where a.datpre = :V_dtDataMvto
and a.codcli = b.codcli
and b.indbro = "N"
and b.indpld = "N"
order by a.agecom, a.codcli;
```

- Valida se os clientes têm relacionamento da Atividade Custódia com a Atividade BM&F no Cadastro de Clientes (Sinacor > Cadastro de Clientes > Cadastro Geral > Atividade BM&F > Aba Relacionamento):

```
select cd_cliente_dest
from tscrelacatvd
where cd_cliente_ori = 'código contas'
and cd_sistema_ori = 'BMF'
and cd_sistema_dest = 'CF';
```

- Valida se os clientes estão cadastros ou desativados na Atividade Custódia no Cadastro de Clientes (Sinacor > Cadastro de Clientes > Cadastro Geral > Atividade Custódia):

```
select in_situac from tscclicus where cd_cliente = 'código dos clientes';
```

Retorna 'X' caso não exista na tabela e 'D' como desativado;

- Valida se o mercado das posições estão cadastros no Gestor de Ativos:

```
select 0
from tcfmercado
where tipo_grup = 'BMF'
and tipo_merc = 'Mercados de derivativos: Termo, Exercio de Opções, Opções, Contratos Futuros, etc.';
```

- Erros na validação da integração são incluídos da TCFERRO.
- Atualiza posição no Gestor de Ativos: tcfposi_bmf
- Insere movimentação no Gestor de Ativos com os respectivos mercados / descrição (spCf_Integra.spCf_Intr_Bmf): tcfmovi_bmf

• Integração de operações de Bolsa:

- Valida se o módulo de Faturamento Bovespa está instalado no participante:

```
select count(*)
from tgesinst a, tgeempr b
where a.cd_sistema = 'BOL'
and b.cd_empresa = a.cd_empresa
and b.cd_usua_orac = user
and instr( a.vs_sist_inst, 'D' ) = 0;
```

- Consulta Integração das movimentações geradas pelo Faturamento Bovespa no pregão anterior (agrupadas em Local, Conta, Isin, Distrib, Carteira, Data Aquisição, Valor da Aquisição): tbointr_cust

- Valida se os Mercados e o códigos de negociação a serem integrados existem no Gestor de Ativos e se estão dentro do período de negociação (Importados pelo PAPT):

```
select tipo_merc, cod_isin, num_dist, to_char(data_ini_neg,'dd/mm/yyyy') data_ini_neg,
       to_char(data_exer_opc,'dd/mm/yyyy') data_exer_opc
from tcfpap_merc
where cod_neg = 'Códigos de Negociação Ex: PETR4, VALE5, etc.' ** para ETF (ultima letra do código de negociação
= 'G'), buscar o mercado com o código do a vista
and data_ini_neg <= 'Data do movimento'
and data_lim_neg >= 'Data do movimento';
```

- *** Avaliar: Se para ETF não for encontrado o próprio código ou como VIS, gera um cadastro de título a partir do ISIN: spCf_Pu_Pap.spCf_Cad_Cod_Neg ETF;
- Valida se a carteira de negociação está cadastrada no GA ([Cadastro de Carteiras](#));

```
select * from TCFCARTEIRA;
```

- Valida se a conta da operação, ou a conta direcionada para custódia interna (próprio participante), ignorando o mercado de Opções e Termo, existe no Cadastro de Clientes > Atividade Custódia ()
 - Como identificar se houve direcionamento interno de custódia:
 - tbointr_cust.ind_tipo_liqd = 'C' and tbointr_cust.cod_usua_cant = o código do próprio participante (tgelocal.cd_empr_bolsa)
 - Valida o cadastro do relacionamento da Atividade Bovespa com Atividade Custódia:

```
select cd_cliente_dest
from tscrelacatvd
where cd_cliente_ori = 'código contas'
and cd_sistema_ori = 'BOL'
and cd_sistema_dest = 'CF';
```

- Valida se os clientes estão cadastros ou desativados na Atividade Custódia no Cadastro de Clientes (Sinacor > Cadastro de Clientes > Cadastro Geral > Atividade Custódia):


```
select in_situac from tscclicus where cd_cliente = 'código contas';
```

 Retorna 'X' caso não exista na tabela e 'D' como desativado;
- Valida se o módulo de Faturamento Bovespa está aberto para o movimento corrente:

```
select dt_datmov, tp_movimento = 'A' from tboparam;
```

- Erros na validação da integração são incluídos da TCFERRO.
- Identifica todas operações realizadas no Faturamento Bovespa do pregão anterior:

```
select w.data_nego
      , w.bol_mvto
      , w.cod_isin
      , w.cod_neg
      , w.cod_nat_ope nat_ope
      , w.num_nego
      , case when w.ind_faz_prop = 0 then w.qtde_espe
            else abs(w.qt_liquid_tot)
        end
      , qtde_espe
      , w.cod_crsp
      , w.val_nego
      , w.num_dias_venc
      , w.custo_opel cst_ope
      , w.fat_cot
      , w.cod_usua_conp
      , w.cod_agen_qlfd
      , w.cod_cli_qlfd
      , w.ind_mvto -- "DIA": movimentações integradas do dia e "EST": Movimentações
integradas que foram recalculadas pelo Fat B01
      , w.num_dias_liqd
      , w.num_seq_ordm
      , decode(nvl(w.cod_usua_cant, 0), 0, :V_nCodUsuaCant, w.cod_usua_cant)
      , decode(decode(nvl(w.cod_usua_cant, 0), 0, :V_nCodUsuaCant, w.cod_usua_cant), :
V_nCodUsuaCant, decode(nvl(w.cod_cli_cant, 0), 0, w.cod_cli, w.cod_cli_cant)
      , nvl(w.cod_cli_cant, 0))
      , nvl(w.ind_tipo_liqd, ' ')
      , decode(w.cod_cart_liqd, 0, 216, w.cod_cart_liqd) cod_cart_liqd
      , w.cod_cli
      , w.nr_ofemega
      , w.cod_tipo_merc
      , w.cod_tipo_liqc
      , 'N' ind_neg_part_dayt
```

```

, null cod_etrg_neg
, case when w.ind_faz_prop = 0 then w.vl_tot_brut
    else case when w.cnt = 2 then
        case when w.cod_nat_ope = 'V' then
            w.vl_tot_brut_prop_tot_vda * w.qt_liquid_tot / w.
qt_liquid
            else w.vl_tot_brut_prop_tot_cpa * w.qt_liquid_tot / w.
qt_liquid
        end
    else greatest( w.vl_tot_brut_prop_tot_cpa, abs( w.
vl_tot_brut_prop_tot_vda ) )
    end
end vl_tot_brut
, case when w.ind_faz_prop = 0 then w.vl_tot_liqd
    else case when w.cnt = 2 then
        case when w.cod_nat_ope = 'V' then
            w.vl_tot_liqd_prop_tot_vda * w.qt_liquid_tot / w.
qt_liquid
            else w.vl_tot_liqd_prop_tot_cpa * w.qt_liquid_tot / w.
qt_liquid
        end
    else greatest( w.vl_tot_liqd_prop_tot_cpa, abs( w.
vl_tot_liqd_prop_tot_vda ) )
    end
end vl_tot_liqd
, w.ind_rpss
from (select x.*
    , row_number() over (partition by x.data_nego
        , x.cod_cli
        , x.cod_isin
        , x.ind_mvto
        , nvl(x.cod_usua_cant, :V_nCodUsuaCant)
        , nvl(x.cod_cli_cant, x.cod_cli)
        , x.cod_crsp
        , x.num_dias_liqd
        , x.cod_tipo_liqc
        , x.rwid_grup
        order by abs(x.qt_liquid) desc)
    rn_tot
    , sum(case when x.cod_nat_ope = 'V' then -abs(x.qt_liquid) else abs(x.
qt_liquid) end)
    over (partition by x.data_nego
        , x.cod_cli
        , x.cod_isin
        , x.ind_mvto
        , nvl(x.cod_usua_cant, :V_nCodUsuaCant)
        , nvl(x.cod_cli_cant, x.cod_cli)
        , x.cod_crsp
        , x.num_dias_liqd
        , x.cod_tipo_liqc
        , x.rwid_grup)
    qt_liquid_tot
    , sum(x.vl_tot_brut_prop_cpa) over(partition by x.data_nego
        , x.cod_cli
        , x.cod_isin
        , x.ind_mvto
        , nvl(x.cod_usua_cant, :
V_nCodUsuaCant)
        , nvl(x.cod_cli_cant, x.cod_cli)
        , x.cod_crsp
        , x.num_dias_liqd
        , x.cod_tipo_liqc
        , x.rwid_grup)
    vl_tot_brut_prop_tot_cpa
    , sum(x.vl_tot_liqd_prop_cpa) over(partition by x.data_nego
        , x.cod_cli
        , x.cod_isin
        , x.ind_mvto
        , nvl(x.cod_usua_cant, :
V_nCodUsuaCant)
        , nvl(x.cod_cli_cant, x.cod_cli)

```

```

, x.cod_crsp
, x.num_dias_liqd
, x.cod_tipo_liqc
, x.rwid_grup)

vl_tot_liqd_prop_tot_cpa
, sum(x.vl_tot_brut_prop_vda) over(partition by x.data_nego
, x.cod_cli
, x.cod_isin
, x.ind_mvto
, nvl(x.cod_usua_cant, :
V_nCodUsuaCant)
, nvl(x.cod_cli_cant, x.
cod_cli)
, x.cod_crsp
, x.num_dias_liqd
, x.cod_tipo_liqc
, x.rwid_grup)

vl_tot_brut_prop_tot_vda
, sum(x.vl_tot_liqd_prop_vda) over(partition by x.data_nego
, x.cod_cli
, x.cod_isin
, x.ind_mvto
, nvl(x.cod_usua_cant, :
V_nCodUsuaCant)
, nvl(x.cod_cli_cant, x.
cod_cli)
, x.cod_crsp
, x.num_dias_liqd
, x.cod_tipo_liqc
, x.rwid_grup)

vl_tot_liqd_prop_tot_vda
from (select k.*
, count(distinct(k.nat_ope)) over(partition by k.data_nego
, k.cod_cli
, k.cod_isin
, k.ind_mvto
, nvl(k.cod_usua_cant,
:V_nCodUsuaCant)
, nvl(k.cod_cli_cant,
k.cod_cli)
, k.cod_crsp
, k.num_dias_liqd
, k.cod_tipo_liqc
, k.rwid_grup)

cnt
, case when k.cod_tipo_merc in ('TER', 'FUT') then k.csto_opel
else 0
end custo_opel
, case when k.ind_faz_prop = 1 then
case when k.qt_liquid > 0 then 'C'
else 'V'
end
else k.nat_ope
end cod_nat_ope
, spCf_Integra.fFra2Vis(k.cod_neg) cod_neg_vis
, case when k.ind_faz_prop = 1 then
case when k.qt_liquid > 0 then
decode(k.qt_brut_cpa, 0, 0, k.qt_liquid * k.
vl_tot_brut_liquid_cpa / k.qt_brut_cpa)
else 0
end
else k.vl_tot_brut_liquid_cpa
end vl_tot_brut_prop_cpa
, case when k.ind_faz_prop = 1 then
case when k.qt_liquid > 0 then
decode(k.qt_brut_cpa, 0, 0, k.qt_liquid * k.
vl_tot_liqd_liquid_cpa / k.qt_brut_cpa)
else 0
end
else k.vl_tot_liqd_liquid_cpa
end vl_tot_liqd_prop_cpa

```

```

, case when k.ind_faz_prop = 1 then
    case when k.qt_liquid < 0 then
        decode(k.qt_brut_vda, 0, 0, k.qt_liquid * k.
vl_tot_brut_liquid_vda / k.qt_brut_vda)
    else 0
    end
    else k.vl_tot_brut_liquid_vda
end vl_tot_brut_prop_vda
, case when k.ind_faz_prop = 1 then
    case when k.qt_liquid < 0 then
        decode(k.qt_brut_vda, 0, 0, k.qt_liquid * k.
vl_tot_liqd_liquid_vda / k.qt_brut_vda)
    else 0
    end
    else k.vl_tot_liqd_liquid_vda
end vl_tot_liqd_prop_vda
from (select a.*
, d.num_tipo_con
, case when d.num_tipo_con != 3
    and a.cod_tipo_merc not in ('OPC', 'OPV',
'TER', 'IER', 'EOC', 'EOV'))
    then 1
    else 0
end ind_faz_prop
, case when d.num_tipo_con != 3
    and a.cod_tipo_merc not in ('OPC', 'OPV',
'TER', 'IER', 'EOC', 'EOV'))
    then to_char(decode(a.cod_cart_liqd, 0, 216, 236,
216, cod_cart_liqd))
    else rowidtochar(a.rowid)
end rwid_grup
, row_number() over( partition by a.data_nego
, a.cod_cli
, a.cod_isin
, a.cod_cart_liqd
, a.ind_mvto
, nvl(a.cod_usua_cant, :
V_nCodUsuaCant)
, nvl(a.cod_cli_cant, a.
cod_cli)
, a.cod_crsp
, a.num_dias_liqd
, a.cod_tipo_liqc
, case when decode(a.
cod_cart_liqd, 0, 216, cod_cart_liqd) in (216, 236) then 216
else cod_cart_liqd
end
, case when d.num_tipo_con !=
= 3 then '1' else rowidtochar(d.rowid) end
, case when a.cod_tipo_merc
not in ('OPC', 'OPV', 'TER', 'IER', 'EOC', 'EOV') then '1'
else rowidtochar(a.
rowid)
end
order by a.cod_neg )
rn
, sum(case when a.nat_ope = 'V' then -a.qtde_espe else a.
qtde_espe end)
over (partition by a.data_nego
, a.cod_cli
, a.cod_isin
, a.cod_cart_liqd
, a.ind_mvto
, nvl(a.cod_usua_cant, :V_nCodUsuaCant)
, nvl(a.cod_cli_cant, a.cod_cli)
, a.cod_crsp
, a.num_dias_liqd
, a.cod_tipo_liqc
, case when decode(a.cod_cart_liqd, 0,
216, cod_cart_liqd) in (216, 236) then 216
else cod_cart_liqd

```

```

end
, case when d.num_tipo_con != 3 then
'1' else rowidtochar(d.rowid) end
, case when a.cod_tipo_merc not in
('OPC', 'OPV', 'TER', 'IER', 'EOC', 'EOV') then '1'
else rowidtochar(a.rowid)
end)
qt_liquid
, sum(case when d.num_tipo_con != 3
and a.cod_tipo_merc not in ('OPC', 'OPV',
'TER', 'IER', 'EOC', 'EOV')
and a.nat_ope = 'V'
then 0
else a.qtde_espe
end)
over (partition by a.data_nego
, a.cod_cli
, a.cod_isin
, a.cod_cart_liqd
, a.ind_mvto
, nvl(a.cod_usua_cant, :V_nCodUsuaCant)
, nvl(a.cod_cli_cant, a.cod_cli)
, a.cod_crsp
, a.num_dias_liqd
, a.cod_tipo_liqc
, case when decode(a.cod_cart_liqd, 0,
216, cod_cart_liqd) in (216, 236) then 216
else cod_cart_liqd
end
, case when d.num_tipo_con != 3 then
'1' else rowidtochar(d.rowid) end
, case when a.cod_tipo_merc not in
('OPC', 'OPV', 'TER', 'IER', 'EOC', 'EOV') then '1'
else rowidtochar(a.rowid)
end)
qt_brut_cpa
, sum(case when d.num_tipo_con != 3
and a.cod_tipo_merc not in ('OPC', 'OPV',
'TER', 'IER', 'EOC', 'EOV')
and a.nat_ope = 'C'
then 0
else a.qtde_espe
end)
over (partition by a.data_nego
, a.cod_cli
, a.cod_isin
, a.cod_cart_liqd
, a.ind_mvto
, nvl(a.cod_usua_cant, :V_nCodUsuaCant)
, nvl(a.cod_cli_cant, a.cod_cli)
, a.cod_crsp
, a.num_dias_liqd
, a.cod_tipo_liqc
, case when decode(a.cod_cart_liqd, 0,
216, cod_cart_liqd) in (216, 236) then 216
else cod_cart_liqd
end
, case when d.num_tipo_con != 3 then
'1' else rowidtochar(d.rowid) end
, case when a.cod_tipo_merc not in
('OPC', 'OPV', 'TER', 'IER', 'EOC', 'EOV') then '1'
else rowidtochar(a.rowid)
end)
qt_brut_vda
, sum(case when d.num_tipo_con != 3
and a.cod_tipo_merc not in ('OPC', 'OPV',
'TER', 'IER', 'EOC', 'EOV')
and a.nat_ope = 'V'
then 0
else a.vl_tot_brut
end)
end)

```

```

over (partition by a.data_nego
      , a.cod_cli
      , a.cod_isin
      , a.cod_cart_liqd
      , a.ind_mvto
      , nvl(a.cod_usua_cant, :V_nCodUsuaCant)
      , nvl(a.cod_cli_cant, a.cod_cli)
      , a.cod_crsp
      , a.num_dias_liqd
      , a.cod_tipo_liqc
      , case when decode(a.cod_cart_liqd, 0,
216, cod_cart_liqd) in (216, 236) then 216
      else cod_cart_liqd
      end
      , case when d.num_tipo_con != 3 then
      , case when a.cod_tipo_merc not in
(''OPC'', ''OPV'', ''TER'', ''IER'', ''EOC'', ''EOV'') then ''1''
      else rowidtochar(a.rowid)
      end)
      vl_tot_brut_liquid_cpa
      , sum(case when d.num_tipo_con != 3
      and a.cod_tipo_merc not in (''OPC'', ''OPV'',
''TER'', ''IER'', ''EOC'', ''EOV'')
      and a.nat_ope = ''C''
      then 0
      else a.vl_tot_brut
      end)
over (partition by a.data_nego
      , a.cod_cli
      , a.cod_isin
      , a.cod_cart_liqd
      , a.ind_mvto
      , nvl(a.cod_usua_cant, :V_nCodUsuaCant)
      , nvl(a.cod_cli_cant, a.cod_cli)
      , a.cod_crsp
      , a.num_dias_liqd
      , a.cod_tipo_liqc
      , case when decode(a.cod_cart_liqd, 0,
216, cod_cart_liqd) in (216, 236) then 216
      else cod_cart_liqd
      end
      , case when d.num_tipo_con != 3 then
      , case when a.cod_tipo_merc not in
(''OPC'', ''OPV'', ''TER'', ''IER'', ''EOC'', ''EOV'') then ''1''
      else rowidtochar(a.rowid)
      end)
      vl_tot_brut_liquid_vda
      , sum(case when d.num_tipo_con != 3
      and a.cod_tipo_merc not in (''OPC'', ''OPV'',
''TER'', ''IER'', ''EOC'', ''EOV'')
      and a.nat_ope = ''V''
      then 0
      else a.vl_tot_liqd
      end)
over (partition by a.data_nego
      , a.cod_cli
      , a.cod_isin
      , a.cod_cart_liqd
      , a.ind_mvto
      , nvl(a.cod_usua_cant, :V_nCodUsuaCant)
      , nvl(a.cod_cli_cant, a.cod_cli)
      , a.cod_crsp
      , a.num_dias_liqd
      , a.cod_tipo_liqc
      , case when decode(a.cod_cart_liqd, 0,
216, cod_cart_liqd) in (216, 236) then 216
      else cod_cart_liqd
      end
      , case when d.num_tipo_con != 3 then

```



```

    '1' else rowidtochar(d.rowid) end
                                , case when a.cod_tipo_merc not in
('OPC', 'OPV', 'TER', 'IER', 'EOC', 'EOV') then '1'
                                else rowidtochar(a.rowid)
                                end)
                                vl_tot_liqd_liquid_cpa
                                , sum(case when d.num_tipo_con != 3
                                and a.cod_tipo_merc not in ('OPC', 'OPV',
'TER', 'IER', 'EOC', 'EOV')
                                and a.nat_ope = 'C'
                                then 0
                                else a.vl_tot_liqd
                                end)
                                over (partition by a.data_nego
                                , a.cod_cli
                                , a.cod_isin
                                , a.cod_cart_liqd
                                , a.ind_mvto
                                , nvl(a.cod_usua_cant, :V_nCodUsuaCant)
                                , nvl(a.cod_cli_cant, a.cod_cli)
                                , a.cod_crsp
                                , a.num_dias_liqd
                                , a.cod_tipo_liqc
                                , case when decode(a.cod_cart_liqd, 0,
216, cod_cart_liqd) in (216, 236) then 216
                                else cod_cart_liqd
                                end
                                , case when d.num_tipo_con != 3 then
                                , case when a.cod_tipo_merc not in
('OPC', 'OPV', 'TER', 'IER', 'EOC', 'EOV') then '1'
                                else rowidtochar(a.rowid)
                                end)
                                vl_tot_liqd_liquid_vda
                                from tbointr_cust a
                                , tscclibol d
                                where a.data_mvto = :V_dtDataMvto
                                and a.cod_cli = d.cd_cliente
                                ) k
                                where k.rn = 1 or k.ind_faz_prop = 0
                                ) x
                                ) w
                                where w.rn_tot = 1
                                and 0 != case when w.ind_faz_prop = 0 then w.qtde_espe else abs(w.
qt_liquid_tot) end
                                order by decode( w.cod_tipo_merc,
                                'OPC', 1,
                                'OPV', 1,
                                'TER', 1,
                                'FUT', 1,
                                0 ),
                                w.ind_mvto desc, w.cod_crsp, w.bol_mvto, w.data_nego, w.ind_tipo_liqd, w.
cod_usua_cant, w.cod_cli_cant, w.cod_cli, w.cod_isin, w.cod_tipo_liqc,
                                decode( w.cod_tipo_merc,
                                'EOC', 0,
                                'EOV', 1,
                                'VIS', 2,
                                'FRA', 3,
                                4 ),
                                w.cod_neg, w.num_nego, w.cod_cart_liqd

```

- Atualiza projeção de posição a Vista: tcfbloqueada.
 - Mercados que geram projeção do a Vista: tcfmercado
 - VIS: Vista
 - FRA: Fracionário
 - EOC: Exercício de opção de Compra
 - EOV: Exercício de opção de Venda
 - LEI: Leilão
 - IER: Integralização e Resgate
 - CET: Cotas de ETF
 - LNC: Leilão de não Cotados

- Atualiza posição de Opções (novas movimentações): tcfposi_opc
 - Gera movimentação de "Integração/Estorno de Bolsas Credito/Debito": tcfmovi_acao
- Tratamento de exercício de opções: A partir da query acima, identifica as operações de Exercício (EOV, EOC) e faz a baixa da posição de opção referente a série da opção (tcfposi_opc);
 - Gera movimentação de "Exercício de opções Venda/Compra": tcfmovi_acao
- Atualiza posição de Termo (novas movimentações): tcfposi_term
 - Gera movimentação de "Integração/Estorno de Bolsas Credito/Debito": tcfmovi_acao
- Deleta tabela de integração TBOINTR_CUST com data de movimento (data_mvto) < menor data do calendário do GA (select min (data_mvto) from tcfcale);
- **Transferência de movimento digitado no GA:** Descontinuado.
 - Tela manual para inclusão de posição: acflanc_acao
 - Gera a posição tcfposi_Vist/tcfposi_term/tcfposi_opc/tcfposi_doads...
 - Insere movimentação tcfmovi_acao/tcfmovi_btc
- **Liquidação de operações de Termo - Cobertura de Venda de Termo:** *** Rever processo ***
 - Venda a termo com posição existente: tcfposi_term com cod_cart = 26018;
 - Venda a termo sem posição existente: tcfposi_term com cod_cart = 21016;
 - Comprava a vista (D-3/D-2): gerava tcfbloqueada
 - Identificava posições de termo descobertas e consulta projeção de liquidação de compra do a vista e usava essa projeção/liquidação para cobrir o termo:
 - tcfposi_cobe_term
 - tcfposi_cobe_term_vist
- **Liquidação de operações (projeções) com direcionamento de Custodiante:** Descontinuado.
 - Baixa nas projeções da TCFBLOQUEADA.ind_tipo_liqd = 'C'
 - Gera movimentação na tcfmovi_acao como 'LIQUIDACAO CUSTODIANTE EXTERNO '
- **Gera pendências de liquidação:**
 - Na abertura do movimento do próximo dia útil das liquidações (D+3 da operação), o sistema identifica liquidações que não receberam a movimentação da B3 (depositária) pela mensageria CBL0550 ou pelos arquivos CMC1,CMVC2 e CMVD (e CSVs):
 - Tipo de Movimentação = LC04
 - Se não encontrado, gera tcfpendencia.

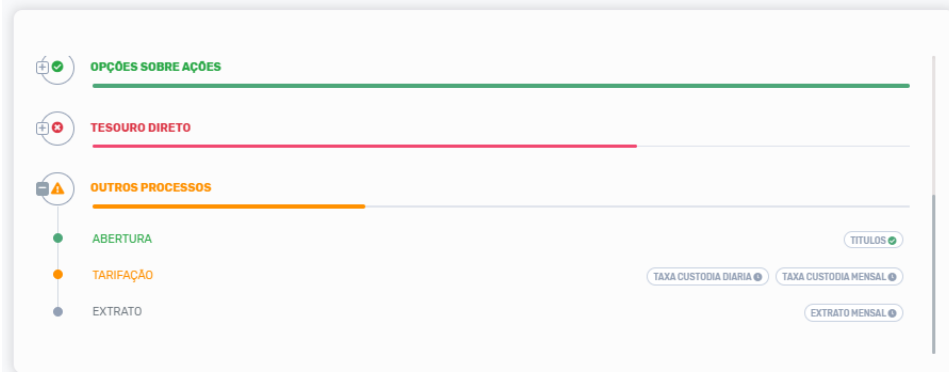
	13/fev	14/fev	15/fev	16/fev	17/fev
Compra		1200			
Venda					
Tomado	-1000				
Pendenc				0	200
Disp	0	0	0	0	0
Alq.	0	0	0	200	0
D+1	0	0	200	0	0
D+2	0	200	0	0	0
D+3	0	0	0	0	0
Total	0	200	200	200	200
				LC04	200

- **Tratamento de Opções que viraram pó:**
 - spCf_Posicao.spCf_Trat_Opc_Vecd_Bol
 - Consulta posição de opção que não foi exercida no seu vencimento (tcfposi_opc.data_exerc_opc <= Data do Movimento ainda fechado)
 - Insere movimentação (tcfmovi_acao) com o sinal contrário da posição para baixar a quantidade na tcfposi_opc
 - Esse processo é realizado para não aguardar a conciliação de opção e demonstrar diversas divergências.
- **Atualiza contratos de Termo que venceram e não foram liquidados via arquivo:**
 - spCf_Posicao.spCf_Liqd_Term_Vec
 - Consulta posição e cobertura de termo (tcfposi_term, tcfposi_cobe_term, tcfposi_cobe_term_vist) vencido sem ter recebido os arquivos de liquidação (bvb0.062 e bvb0.063) (tcfposi_term.data_venc <= Data do Movimento ainda fechado)
 - Delete posições encontradas.
 - *** Verificar possibilidade de descontinuar essa funcionalidade.
- **Atualiza posições de BMF vencidas:**
 - spCf_Posicao.spCf_Liqd_Bmf_Vecd
 - Consulta contratos de BMF vencidos e não liquidados. (tcfposi_bmf.data_venc <= Data do Movimento ainda fechado)
 - Gera movimentação com sinal invertido para zerar posição.
- **Atualiza Direitos de Subscrição vencidos:**
 - spCf_Pos_Prov.spCf_Liqd_Dir_Sub_Vecd
 - Consulta direitos de subscrição vencidos e não baixados pelo CMVD (tcfdir_subs.data_lim_subs <= Data do Movimento ainda fechado)
 - Deleta direitos encontrados.
 - *** Verificar possibilidade de descontinuar essa funcionalidade.
- **Atualiza Proventos vencidos:**

- spCf_Posicao.spCf_Liqd_Prov_Vecd
 - Consulta proventos provisionados que já foram pagos pela LC09 do CMVD (tcfposi_prov_pago), e que não foram debitados até seu vencimento.
 - Deleta proventos já pagos (existentes na tcfposi_prov_pago)
- **Atualiza Valor de Contrato de Termo atrelados ao Dólar:** Descontinuado.
 - spCf_Posicao.spCf_Atlz_Term_Valo_Cotr
 - Atualiza o valor do contrato de termo diariamente com o preço de exercício oriundo do PAPT (Termos atrelados ao Dolar).
(tcfposi_term.val_cotr = tcfposi_term.val_cotr * tcfpap_merc.prec_exerc)
 - *** Verificar possibilidade de descontinuar essa funcionalidade.
- **Cobra Taxa de Custódia para posições zeradas:**
 - spCf_Taxa.spCf_Trat_Cobr_Exec
 - As taxas de custódia são calculadas mensalmente e anualmente e gravadas na tcfcoobr_Taxa
 - Taxas calculadas como 0,00, ou já integradas para o financeiro, são inativadas para serem movidas para histórico;
- **Cópia Histórica de Movimentações:**
 - spCf_Tabela.spCf_Copi_Hist
 - Copia todas as posições que foram alteradas no pregão anterior da abertura para as respectivas tabelas de histórico para acompanhamento das posições.
 - tcfposi_vist => hcfposi_vist
 - tcfposi_opc => hcfposi_opc
 - tcfposi_term => hcfposi_term
 - tcfposi_fut => hcfposi_fut
 - tcfposi_fut => hcfposi_fut
 - tcfposi_btc_doad => hcfposi_btc_doad
 - tcfposi_btc_toma => hcfposi_btc_toma
- **Elimina depósitos para abertura de Ordem vencidas:** Descontinuado.
 - spCf_Aber_Ordm.spCf_Limp_Ordm_Vecd
 - Limpa ordens abertas direto no GA que já venceram tcfmovi_aber_ordm.data_vald_ordm < 'Data movimento anterior a abertura'
 - *** Verificar possibilidade de descontinuar essa funcionalidade.
- **Calcula nova data do sistema:**
Identifica feriados Nacionais e Regionais para calcular nova data pela proc pge_feriado.fge_Calc_prox_dia_util (XXX,XXX,XXX,XXXX) que são cadastrados na Informações de Apoio (IAPO) > Cadastros > Feriados (TGEFERIA)
- **Liquidação de operações de Day-Trade (Movimentação Contábil):** Descontinuado?
 - spCf_Liqd.spCf_Trat_Dayt
 - Identifica projeções (tcfbloqueada) que são consideradas operações de day-trade, e realiza a liquidação automática das posições (baixa de projeção da tcfbloqueada, gera movimentação tcfmovi_acao, baixa posição diária tcfposi_total_dia)
 - *** Verificar possibilidade de descontinuar essa funcionalidade.
- **Gera Extrato Mensal dos Clientes:**
 - spCf_Extrato.spCf_Mont_Extra
 - Gerar o Relatório de Extrato Mensal dos clientes (resumo de toda posição e movimentação dos clientes), que podem ser extraídos em SinacorWeb > Gestor de Ativos > Relatórios > Extrato.
 - Sempre gerado no primeiro de útil do mês para montar todo o histórico do mês passado (não pode ser um feriado Regional > IAPO > Cadastros > Feriado)
 - Os dados básicos do relatório como Nome, endereço, etc., são retirados do cadastro de clientes (tscclicus, tscclicc, tsccclimb, tsccclibol)
 - Dados de posição são extraídos das tabelas de posição do GA:
 - Vista: tcfposi_vist
 - Opção: tcfposi_opc
 - Termo: tcfposi_term
 - BMF: tcfposi_bmf
 - Tesouro Direto: tcfposi_tedi
 - Os dados de posição são gravados nas tabelas de extrato:
 - Vista/Opção/Termo: hcfposi_extr_acao
 - BMF: hcfposi_extr_bmf
 - TD: hcfposi_extr_tedi
 - Após gravar as posições, o sistema atualiza o número do extrato gerado naquele mês para todas as movimentações do período anterior:
 - Vista/Opção/Termo: tcfmovi_acao.num_extr
 - BMF: tcfmovi_bm.num_extr
 - TD: tcfmovi_tedi.num_extr
 - Se o cliente tiver um conta investimento, altera o ultimo número do extrato no cadastro de cliente (tscclicus)
 - O processo de abertura de movimento durante o período de virada de mês do módulo Gestor de Ativos apresentava lentidão em alguns participantes por conta de sua volumetria.
 - Após a versão 23.2, será permitido retirar o processo de extrato mensal da abertura do Gestor de Ativos:
 - Necessário ligar a chave EXEC_SALD_EXTESALD_ABER na tcfcnfg_cust.valor = 'S'
 - No processo de abertura na virada do mês, apenas serão copiados para um novo repositório os dados necessários para geração do extrato.
 - As tabelas abaixo são apagadas com os dados do mês anterior para nova remontagem.
 - spCf_Extrato_Par.spCf_Copia_Posicao(to_char(V_dtUltDataUtil, 'dd/mm/yyyy'));
 - Insere os dados das posições em tabelas auxiliares:
 - Posição a vista: acfposi_vist_extr
 - Posição de opção: acfposi_opc_extr
 - Posição de termo: acfposi_term_extr
 - Posição de bmf: acfposi_bmf_extr
 - Posição de TD: acfposi_tedi_extr
 - Ao termino da abertura do movimento, o sistema envia uma mensagem rabbitmq para o novo plugin de geração de Extrato Mensal (APE - Custódia - Processos)
 - Fila "SINACOR.DIR.D.PRIV.V1.CF.CF.Extrato.ExtratoSolicitado".

- É possível verificar o status de processamento do extrato no card de avisos do Gestor de Ativos visualizando a aba "Outros Processos" e a tarefa "Extrato Mensal", ou através da tabela TCFMONC_EXEC_DIA filtrando a coluna COD_TARE pelo valor 40.

Processos



- A tarefa manterá a cor cinza todos os dias do mês, exceto no primeiro dia útil. Quando o processamento ocorrer a cor azul informando "Em Processamento" será exibida e posteriormente verde em caso de "Sucesso", ou vermelho em caso de "Erro".

- **Consiste posições que incidem CPMF:**
 - spCf_30092004.spCf_Chck_Pos
 - Validar se existem posições como compra anterior a 30/09/2004 tcfposi_vist_30092004 na principal da tcfposi_vist para fazer baixa.
 - *** Verificar possibilidade de descontinuar essa funcionalidade.
- **Liquida operações de Terceiros vencidas:**
 - spCf_Posicao.spCf_Zera_Terc
 - Necessário estar com a chave IND_ZERO_POSI_TERC ligada na tcfcnfg_cust.valor = 'S':
 - Liquida posições de externas de terceiro que venceram:
 - tcfposi_terc.data_atu = Data do Movimento na abertura
 - Gera liquidação / movimentação inversa da quantidade para zerar a posição:
 - tcfmovi_terc = 'RETIRADA DE POSICAO DE TERCEIRO'
- **Limpa tabelas de controle de proventos do sistema:**
 - spCf_Req_Sald_EvtCus.spCf_Limpa_Req_Venc
 - Limpeza das tabelas de controle de envio de solicitação de cálculo para o radar (cbl0510)
 - *** Verificar possibilidade de descontinuar essa funcionalidade (novo processo de eventos corporativos).
- **Montagem da tabela de posições gerais do GA - TCFPOSI_TOTAL_DIA:**
 - spCf_Pos_Total.spCf_Monta_dia
 - No início do processo, a tabela tcfposi_total_dia é totalmente excluída.
 - Processo de montagem da tabela de posição geral diária dos clientes no Gestor de Ativos. Todas as posições e projeções dos mercados atualmente atendimentos pelo GA, são agrupados nessa única tabela diariamente para facilitar as consultas de posição pelos usuários.
 - OBS: Qualquer alteração na estrutura dessa tabela, é necessário avisar o mercado previamente, pois é largamente utilizada nos sistemas de HomeBroker das corretoras.
 - Consulta de posição disponível: monta tcfposi_total_dia.qtde_disp
 - A vista: tcfposi_vist
 - Termo: tcfposi_term
 - Opção: tcfposi_opc
 - A vista anterior 2004: tcfposi_vist_30092004
 - Consulta projeções: monta tcfposi_total_dia.qtde_alqd / qtde_da1 / qtde_da2 / qtde_da3
 - tcfbloqueada conforme cadastro de perfil de saldo negociação (SinacorWeb > Gestor de Ativos > Cadastros > Perfil de Apresentação de Saldo Negociável)
 - Consulta posição pendente: tcfposi_total_dia.qtde_pend
 - Projeções que não foram liquidadas (não recebimento de LC04 pelos arquivos de movimentação da depositária CMVD, CMVCs)
 - tcfpendencia
 - Consulta posição bloqueada: tcfposi_total_dia.qtde_blqd
 - Projeções com data de liquidação maior que D+3 (ex: leilão)
 - tcfbloqueada
 - Consulta posições de termo para cobertura do a vista: tcfposi_total_dia.qtde_blqd
 - tcfposi_total_dia como mercado "Termo"
 - Encontra posição de a vista para o mesmo isin/distrib e bloqueia a quantidade.
 - Identifica papéis que estão sofrendo eventos corporativos e inicia o cálculo para o radar: [Eventos Corporativos](#)
- **Atualiza valores dos contratos de Empréstimo de Ativos:**
 - spCf_Pos_Total.spCf_Calc_Btc
 - Atualiza diariamente contratos de empréstimo de ativos (doador e tomador) a partir do novo preço médio do ativo objeto do pregão anterior.
 - Atualiza volume total e liquida, atualizando também os valores de comissão do Doador e corretagem carrying e executora.
 - tcfposi_btc_doad
 - tcfposi_btc_toma
- **Liquidação de Termo por Data de Rolagem:**
 - spCf_Liqd.spCf_Liqd_Term_Rola
 - Liquida contratos a termo automaticamente 2 dias úteis antes do vencimento do termo.

- Identifica se parâmetro de rolagem ligado (SinacorWeb > GA > Cadastros > Parametrização > Gera Mov. de Liq. de Termo na data de Rolagem
 - Identifica tcfposi_term.data_venc D+2
 - Gera movimentação dando baixa na posição do termo
 - tcfmovi_acao
 - Gera integração dos valores do termo para os módulos Financeiros
 - tcfmovi_liqd_term
- Atualiza tabelas de controles dos arquivos e mensageria:**
 - Libera conciliação de posição a vista pelo Centura:
 - TCFCTRL_MSG_PROC set ind_reco_cust = 'N'
 - Libera importação dos arquivos diários:
 - Sendo Feriado Regional
 - update tcfarquivo
set ind_stat = 'S'
where nome_arq in ('PAPD');
 - update tcfarquivo
set num_arq = num_arq + 1,
ind_stat = decode(ind_stat, 'X', 'S', 'N'),
cod_prot_arq_prin = null
where nome_arq in ('MDTD','CSTD');
 - Não Sendo Feriado Regional
 - update tcfarquivo
set num_arq = num_arq + 1,
ind_stat = decode(ind_stat, 'X', 'S', 'N'),
cod_prot_arq_prin = null
where nome_arq not in ('CMVC1','CMVC2','CMVC3','CFTE.CBLC','CCBD');
 - update tcfarquivo
set num_arq = num_arq + 2,
ind_stat = 'N',
cod_prot_arq_prin = null
where nome_arq in ('CMVC1','CMVC2');
 - Libera mensageria SMP da depositária limpando as janelas de liquidação:
 - delete tcfseq_hist
 - tcfctrl_prce
 - Libera controle de últimos arquivos do CRTD
 - delete tcfctrl_arq_gerd where NOME_ARQ = 'CRTD';
 - Se chave de Virada estiver ligada, apaga tabela de controle de reinvestimento do TD
 - Chave NovoProcReinvestTesouroDireto (cod_mod=18 e cd_chave = 7)
 - delete tcfctrl_arq_reinv
 - Altera controle de arquivo de taxa de custódia caso virada de mês
 - update tcfarquivo
set num_arq = num_arq + 1,
ind_stat = 'I'
where nome_arq = 'CCBD';
 - Limpa controle de projeção de termo e emprestimo (BVBG.063)
 - update tcfctrl_atlz_proj a
set a.ind_atlz_proj_term = 'N'
, a.ind_atlz_proj_empr = 'N';
 - *** Rever tabelas de controle com os novos processos do GA
- Atualiza Indicador de ultima posição / movimentação no Cadastro de Clientes:**
 - Atualiza atividade custódia do Cadastro de Clientes indicando se o cliente está posicionado em algum ativo:
 - tscclicus.in_posicao = S
 - Atualiza data da ultima movimentação de liquidação de posição (LC04) dos clientes
 - tscclicus.data_ult_mvto
 - Com a chave virada, atualiza tabela de documentos do Cadastro de Clientes com a ultima data de movimentação
 - Chave DesativaAtualzChaveViradaLote (cod_mod=18 e cd_chave = 13)
- Requisita calculo de Saldo ao RADAR**
 - [Eventos Corporativos](#)
- Grava log com usuário, data início e fim na tabela geral do sistema**
 - TGETEMPO