

CPAPI001 - Reporte Produção por Ordem, Operação e Ponto de Controle

 Tempo aproximado para leitura: superior a 15 minutos

Documento de API

Produto:	Datasul
Ocorrência:	Documentação de API
Nome Físico:	cpp/cpapi001.p
Nome das Includes com os Parâmetros:	cpp/cpapi001.i / cpp/cpapi001.i1

Obs.: É preciso declarar as includes {cdp/cdcfgman.i} e {cdp/cdcfgmat.i} no programa específico antes das includes de definição de temp-tables, pois elas contém os pré-processadores necessários para o correto funcionamento da API.

Objetivo

Efetuar reportes da produção, sendo responsável pela entrada do produto acabado no estoque, dar baixa na matéria-prima utilizada para a produção destes itens e ainda efetuar transações específicas ao reporte tais como a geração de refugo e fichas de controle de qualidade.

Funcionamento

A API pode ser chamada de duas formas: A padrão (batch) e a persistente.

Na forma padrão, realiza-se a chamada através de um comando RUN normal, passando-se as temp-tables carregadas com as informações necessárias ao processamento de um ou mais reportes. A sintaxe para a chamada é:

```
run cpp/cpapi001.p (input-output table tt-rep-prod,  
input table tt-refugo,  
input table tt-res-neg,  
input table tt-apont-mob,  
input-output table tt-erro,  
input l-deleta-erros).
```

A execução na forma persistente, permite apenas o “carregamento” na memória do código da API, sem a execução de nenhum reporte, devendo as temp-tables serem passadas sem nenhum registro. Uma vez carregada, a API permite a execução de diversos “métodos”, que poderão ser chamados de forma interativa, englobando todo o processo de reporte. Esses “métodos” são procedures que são acessadas através de um “handle” que é retornado quando o programa é executado de forma persistente. A sintaxe para executar a API em modo persistente é;

```
run cpp/cpapi001.p persistent set h-handle (input-output table tt-rep-prod,  
input table tt-refugo,  
input table tt-res-neg,  
input table tt-apont-mob,  
input-output table tt-erro,  
input l-deleta-erros).
```

A variável denominada h-handle irá receber o “handle” do programa, permitindo o acesso aos “métodos” internos da API.

Parâmetros

Os parâmetros utilizados pela API são:

- tt-rep-prod – Temp-table contendo as informações principais do reporte tais como as quantidades reportadas e refugadas e depósito e localização de saída dos componentes.
- tt-refugo – Temp-table contendo as quantidades e motivos do refugo.

- tt-res-neg – Contém as informações relativas às reservas da ordem que gerarão as requisições.
- tt-apont-mob – Contém as informações relativas ao apontamento de mão-de-obra direta e GGF quando a ordem possuir apontamento pelo “real”.
- tt-erro – Temp-table que retorna os erros e “avisos” retornados pela API durante o processamento dos reportes.
- l-deleta-erros – Informa à API que o conteúdo anterior, eventualmente existente na temp-table de erros será eliminado logo após o início do processamento da API. Normalmente utiliza-se YES, sendo parametrizado como NO apenas se a chamada para a API fizer parte de uma transação maior, que eventualmente já tenha gerado registros na temp-table de erros e que devam permanecer após a chamada da API.

Além das temp-tables acima, se a API for executada em modo persistente, são utilizadas outras temp-tables a fim de passar os parâmetros para os “métodos” da API. Essas temp-tables são:

- tt-aloca – Contém as informações relativas à alocação das reservas da ordem de produção.
- tt-reservas – Contém as informações relativas às reservas da ordem, as quais irão gerar a movimentação de requisição para a ordem.

Retorno

A API irá setar o return-value de forma a informar ao programa “chamador” se a execução foi realizada com sucesso ou não. Os valores possíveis para o return-value são: OK (execução correta) e NOK (execução com problemas). Os motivos para a rejeição do movimento de reporte são especificados na temp-table tt-erro.

Principais Campos

Os principais campos, utilizados pela API são:

- TT-REP-PROD:

Tipo – Determina o tipo do reporte que será realizado, segundo a tabela abaixo:

- 1 – Reporte por Ordem
- 2 – Reporte por Operação
- 3 – Reporte por Ponto de Controle

! Importante

1. O campo Tipo tem como padrão ser "1" (Reporte por Ordem), então se não for informado o valor do campo, sempre considera ser "1". Assim sendo, é importante informar o tipo principalmente quando a ordem for reporte tipo "2" ou "3", pois a API fará a consistência se os campos são iguais tt-rep-prod.tipo e ord-prod.tipo.
2. 2. quando a ordem de produção (for tipo de reporte por Operação - ord-prod.tipo = 2 ou Ponto de Controle - ord-prod.tipo = 3) e ainda não tenha sido reportada, ao fazer o primeiro reporte, caso o campo tt-rep-prod.tipo = 1 (reporte por Ordem), será automaticamente alterado o tipo de reporte para Ordem (ord-prod.tipo = 1).

Nr-report – Retorna o número do reporte que foi gerado internamente pela API.

Nr-ord-produ – Contém o número da ordem de produção que será reportada.

Data – Data de processamento do reporte

Qt-report – Quantidade a ser reportada

Cod-depos, cod-localiz – Depósito e localização de entrada do produto acabado.

Cod-depos-sai, cod-local-sai – Depósito e localização de saída dos componentes.

Op-codigo, cod-roteiro, it-oper, sequencia – Determina a operação a ser reportada (somente no reporte por operação)

Pto-controle – Determina o ponto de controle sendo reportado

Baixa-reservas – Define a forma como as reservas serão baixadas, conforme abaixo:

- 1 – Proporcionalmente à quantidade da ordem (default)

Time-out, Tentativas – Informam o tempo de espera (em segundos) e o número de tentativas que serão realizadas para aguardar a liberação de um registro que esteja “travado” por outro usuário (somente quando a API for executada via RPC).

Procura-saldos – Informa se a API irá realizar a busca dos saldos (FIFO) automaticamente no depósito de saída informado (o default para execução batch é YES).

Carrega-reservas – Informa se a API irá carregar automaticamente uma temp-table interna com as reservas a serem processadas, ou se irá recebê-la através do método pi-recebe-tt-reservas. Esse parâmetro deve ser YES para execução batch (default) e NO para execução persistente.

Requis-automática – Se for YES, as ordens que possuem itens considerados como subconjuntos ou conjuntos (possuem componentes agregados) abrirão uma janela para a informação dos dados e quantidade a ser reportada, possibilitando requisitar automaticamente, os materiais necessários para a fabricação dos componentes que são agregados ao item da ordem.

Prog-seg – Recebe nome do programa de origem, ou seja, do programa que está “chamando” a API de reporte.

Finaliza-ordem – Informa se a API deve finalizar a ordem de produção após o reporte.

Finaliza-oper – Informa se a API deve finalizar a operação após o reporte (apenas para reporte por operação e por ponto de controle).

Reserva – Define se a API irá carregar automaticamente uma temp-table interna com as reservas a serem processadas, ou se irá recebê-las, através da temp-table tt-res-neg. Esse parâmetro é utilizado em conjunto com o “Carrega-reservas” (em modo batch).

- **TT-REFUGO:**

Nr-ord-produ – Indica o número da ordem de produção a qual o movimento de refugo pertence.

Codigo-rejei – Código de rejeição

Qt-refugo – Quantidade refugada/aprovada condicional para o respectivo código de rejeição.

Nro-ord-seq – Sequência da ocorrência de refugo, para a ordem de produção.

- **TT-APONT-MOB:**

Nr-ord-produ – Indica o número da ordem de produção a qual o movimento de mão-de-obra/GGF pertence.

Tipo-movto – Indica se o movimento é de MOB direta (1) ou GGF (2).

Tipo-relógio – Indica o tipo de relógio utilizado para dar entrada nas informações de tempo do movimento:

1 – Horário

2 – Dif. Horário

3 – Centesimal

4 – Dif. Centesimal

Execução Persistente

Quando a API for executada em modo persistente, os seguintes métodos estarão disponíveis para execução:

f-gera-numero-reporte()

Retorna um número de reporte ainda não utilizado.

f-retorna-ult-oper (input i-nr-ord-prod as integer)

Retorna o ROWID da última operação da ordem de produção informada.

f-adiciona-lista (input c-lista as char

input c-item as char)

Retorna uma lista separada por vírgulas, contendo a lista original passada como parâmetro mais o item informado (c-item).

pi-recebe-tt-rep-prod (input table tt-rep-prod)

pi-recebe-tt-reservas (input table tt-reservas)

pi-recebe-tt-aloca (input table tt-aloca)

pi-recebe-tt-apont-mob (input table tt-apont-mob)

pi-recebe-tt-refugo (input table tt-refugo)

Recebem um input parameter table contendo a temp-table correspondente. Tem por objetivo enviar para a API uma temp-table, que vai ser utilizada na execução de outros métodos.

pi-retorna-tt-rep-prod (output table tt-rep-prod)

pi-retorna-tt-reservas (output table tt-reservas)

pi-retorna-tt-aloca (output table tt-aloca)

pi-retorna-tt-apont-mob (output table tt-apont-mob)

pi-retorna-tt-refugo (output table tt-refugo)

Retornam (através de um output parameter table) da API a temp-table correspondente.

pi-processa-reportes (input-output table tt-rep-prod,

input table tt-refugo

input table tt-res-neg,

input-output table tt-erro,

input l-deleta-erros,

input l-gera-reqs)

Procedure principal da API de reporte, responsável pelo processamento e geração de todos os movimentos relativos ao reporte. Recebe os mesmos parâmetros que a API de reporte quando executada em modo batch, mais o parâmetro l-gera-reqs que informa se o reporte será feito parcialmente, processando o movimento de estoque relativo ao acabado e os movimentos de mão-de-obra e GGF, não realizando o processamento do movimento de estoque das requisições (utilizado para execução assíncrona, ainda não disponível). Atualmente, deve ser definido como YES. Através do return-value verifica-se a execução da procedure. Se for "OK", o reporte foi realizado com sucesso. Se for "NOK", o reporte não foi realizado.

pi-valida-ordem (input l-imediato,

input i-nr-ord-prod,

input i-tipo-reporte,

output c-erro,

output c-texto)

Realiza validações relativas à ordem de produção a ser reportada. O parâmetro l-imediato informa à procedure se ela deve retornar imediatamente após detectar o primeiro erro, ou se deve realizar todas as validações antes de retornar. O parâmetro i-nr-ord-prod informa o número da ordem de produção que será validada. O campo i-tipo-reporte permite informar o tipo de reporte que será realizado (1-Ordem, 2-Operação, 3-Pto. Controle). O retorno da procedure ocorre através dos output parameters c-erro (que contém uma lista dos códigos de erros encontrados) e c-texto (que contém uma lista de textos eventualmente associados a cada erro). Também é utilizado o return-value ("OK" ou "NOK").

pi-valida-rep-prod (input l-imediato,

input i-nr-ord-prod,

output c-erro,

output c-texto)

Realiza todas as validações relativas ao movimento de reporte. Os parâmetros e o return-value funcionam da mesma forma que a procedure anterior. Para executar essa procedure, é necessário que a API tenha recebido a temp-table de reporte (pi-recebe-tt-rep-prod). Obs.: Essa procedure executa internamente a pi-valida-ordem (se o parâmetro l-imediato for NO).

pi-valida-reserva (input i-nr-ord-prod,

```
input i-seq,  
input l-imediato,  
output c-erro,  
output c-texto)
```

Valida um registro da temp-table de reservas, localizado através dos parâmetros i-nr-ord-prod e i-seq, que contém respectivamente o número da ordem e a sequência da reserva da ordem, sendo que esses campos constituem a chave primária e única da temp-table de reservas. Os parâmetros l-imediato, c-erro, c-texto e também o return-value funcionam como nas procedures anteriores. É necessário que a temp-table de reservas já esteja carregada na instância da API.

```
pi-valida-alocacao (input i-nr-ord-prod,  
input i-seq,  
input l-imediato,  
output c-erro,  
output c-texto)
```

Realiza a validação de um registro da temp-table de alocações. Os demais parâmetros e o return-value seguem o funcionamento das procedures anteriores.

pi-finalizar

Finaliza a execução persistente da API, eliminando o código da memória e tornando o handle inválido. Deve ser executada após o término de todos os reportes, de preferência na saída do programa “chamador”.

pi-finaliza-ordem (input i-nr-ord-prod)

Realiza a finalização (ou término, se o módulo de custos não estiver implantado) da ordem de produção informada.

pi-verifica-saldo (input c-cod-estabel)

Verifica se as requisições existentes na temp-table tt-reservas possuem ou não saldo disponível para serem atendidas. O campo c-cod-estabel indica o estabelecimento onde será realizada a busca de saldos. Para que essa procedure seja executada, é necessário que a temp-table tt-reservas tenha sido previamente carregada.

```
pi-pesquisa-rede-pert (input-output table tt-oper,  
input rw-oper,  
output r-saida)
```

Retorna na temp-table tt-oper os rowid’s das operações vinculadas a um determinado ponto de controle. A temp-table tt-oper deve ter um registro criado contendo o rowid da operação com o ponto de controle. O parâmetro rw-oper deve conter o rowid da operação com o ponto de controle. Essa procedure é executada de forma recursiva, criando registros na temp-table tt-oper contendo os rowid’s das operações vinculadas ao ponto de controle. A definição da temp-table tt-oper é a seguinte:

```
def temp-table tt-oper no-undo  
field rec-oper as rowid  
INDEX rw-oper IS UNIQUE PRIMARY rec-oper.
```

```
pi-carrega-tt-reservas (input l-reserva,  
input i-tipo,  
input rw-oper-ord)
```

Cria internamente as temp-tables de reservas (tt-reservas) e de alocações (tt-aloca), podendo as mesmas serem obtidas através da execução dos métodos pi-retorna-tt-reservas e pi-retorna-tt-aloca. Para que a execução dessa procedure tenha êxito, a temp-table de reporte (tt-reporte) deve ter sido previamente carregada. O parâmetro l-reserva informa se a procedure irá utilizar as reservas da ordem (YES) ou da temp-table tt-res-neg (NO) para a criação da tt-reservas. O parâmetro i-tipo informa o tipo do reporte e o parâmetro rw-oper-ord contém o rowid da operação (nos reportes por operação e por ponto de controle), devendo ser ? caso o tipo de reporte seja por ordem.

 Macro desconhecida:'rate'

base_de_conhecimento cpapi001 mcp reporte operacao
ponto_decontrole datasul manufatura

