

CPAPI015 - Atualização Requisição Sumariada

 Tempo aproximado para leitura: 5 minutos

Documento de API

Produto:	Datasul
Ocorrência:	Documentação de API
Nome Físico	cpp/cpapi015.p

Considerações Gerais

Objetivo

Efetuar a atualização de quantidades atendidas constantes de requisições sumarizadas.

O programa atualiza a quantidade solicitada e atendida da requisição sumarizada, baseado em todas as reservas de ordens finalizadas (ou não), ativas (ou não). Isto ocorre porque quando as reservas forem eliminadas a requisição sumarizada será automaticamente atualizada.

Quantidade solicitada = soma da quantidade original.

Quantidade atendida = soma das quantidades atendidas.

Somar na quantidade atendida, a quantidade alocada da ordem que ainda se encontra no depósito de processo, pois esta quantidade não poderá ser reaproveitada para outra ordem.

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 uma ou mais atualizações.

A sintaxe para a chamada é:

```
run cpp/cpapi015.p (input-output table tt-req-sum,  
                    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 nenhuma atualização, devendo a temp-table ser passada sem nenhum registro. Uma vez carregada, a API permite a execução de seus “métodos”, que poderão ser chamados de forma interativa, englobando todo o processo de atualização de requisições sumarizadas. 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/cpapi015.p persistent set h-handle (input-output table tt-req-sum,  
                                             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-req-sum – Temp-table contendo as informações principais da atualização das requisições sumarizadas, tais como o número da requisição a ser atualizada.

- tt-erro – Temp-table que retorna os erros e “avisos” retornados pela API durante o processamento das transferências.

- 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.

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 da atualização são especificados na temp-table tt-erro.

Principais Campos: Os principais campos, utilizados pela API são:

- **TT-req-sum:**

Nr-req-sum – Contém o número da requisição sumarizada que será processada.

It-codigo - código do item da requisição.

Cod-versao-integracao – número da versão de integração.

Cód-refer - código da referência do item.

Per-ppm – Utilizado pelo Módulo de controle de potência.

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

Pi-atualiza-req-sum(input-output table tt-req-sum,

input-output table tt-erro,

input l-deleta-erros).

Executa a atualização da requisição.

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 movimentos, de preferência na saída do programa “chamador”.

Principais validações feitas na API:

- verifica se os parâmetros globais estão cadastrados;
- verifica se o módulo de controle de produção está implantado;
- verifica se o código da versão de integração é compatível;
- verifica se o número da requisição é válido.

Tabelas Temporárias

Nos atributos das tabelas temporárias de entrada e saída:

tt-req-sum					Entrada/Saída
					-
Atributo	Tipo	Formato	Valor Inicial	Obrigatório	Evolução
Nr-req-sum	Integer	>>>, >>>, >>9	0		
It-codigo	Char	X(16)			
Cód-versao-integracao	Integer				
Cod-refer	Char	X(8)			
Per-ppm	Decimal	>>>>, >>9.9999			
Índice	Atributos	Primário	Único		
codigo	Nr-req-sum, it-codigo, cod-refer, per-ppm	X	X		

tt-erro		Entrada/Saída
Grava informações sobre qualquer inconsistência ou erro que possa ter ocorrido durante o processamento.		

Atributo	Tipo	Formato	Valor Inicial	Obrigatório	Evolução
i-sequen	Integer	>>>, >>>, >>9	0		
cd-erro	Integer	>>>, >>9	0		
Mensagem	Char	x(255)	?		

 Macro desconhecida: 'rate'

base_de_conhecimento

mcp

cpapi015



Política de
privacidade

Termos
de uso