

Novo ABAP Debugger – Porque largar o velho? – Parte 2

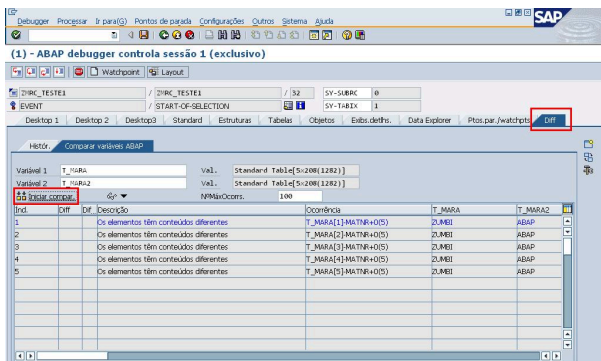
26/01/2011 15:05

Continuando minha saga para **desvendar o Debugger Novo**, vamos dar uma olhada agora em algumas ferramentas “divertidas” que a SAP implantou.

Essas daqui são daquelas que você nunca usa – ou *porque não sabe como*, ou *porque nunca viu*, ou *porque você é o deus do abap e não encontra dificuldades nenhuma pelo seu caminho*.

Como eu não sou deus de nada, eu digo: **Sigam-me os não-deuses!** (tá, viajei 🤪).

1 – Comparação de variáveis no Debug:



Vamos imaginar que eu tenho duas tabelas derivadas da MARA, e eu quero ver se existem diferenças entre os campos das linhas das tabelas. É só clicar em DIFF, colocar as duas tabelas nos campos "Variável" 1 e 2, e comparar! (No caso da foto, eu "marretei" ABAP e ZUMBI no MATNR de cada tabela).

Você pode fazer comparações entre vários tipos de variáveis, não precisa ser tabela. Se tivessem inventado esse negócio antigamente, eu não precisaria ficar fazendo vlookup no excel pra comparar tabelas monstro 🤪

2 – Ferramentas Especiais

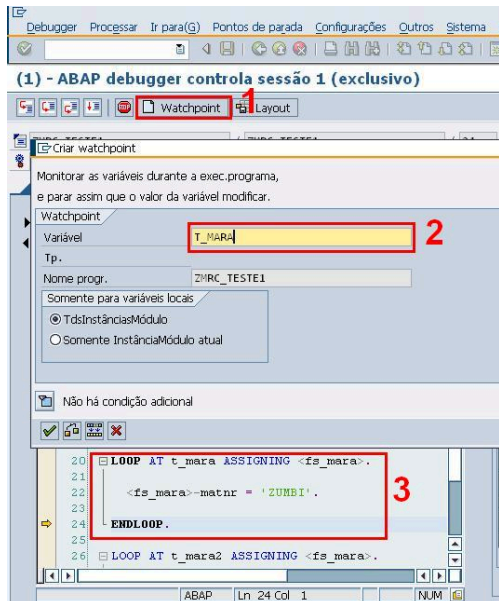
Clicando ali no ícone destacado, vai brotar essa tela que permite adicionar uma nova janela dentro do Debugger novo. Ali tem a parte de "Ferramentas Especiais"

Na esqueda, Ferramenta "Áreas de Sistema (internas)", que mostra uma visão da memória do sistema parecida com o Debug Antigo. Na direita, Ferramenta "Análise de Memória", que nos ajuda a segmentar a memória utilizada por cada programa chamado numa mesma execução.

As outras também são legaiszinhas (ou legalzinhas? sei lá) o negócio é fuçar quando precisar 😊

3 – Watchpoints

Watchpoints é o amigo número contra "debugs intermináveis". O debugger irá **"vigiar"** a variável até que ela atinga uma condição pré-estabelecida por quem estiver debugando. Quando atingir, ele para! Isso já existia no Debugger Velho, mas a do debugger novo é mais legal:



Clicando em Watchpoint, eu escolhi "vigiar" a variável T_MARA (é! agora você pode colocar watchpoints em tabelas!). No meu caso, eu não coloquei mais nada, e cliquei no OK. Como não tem condição, o debugger vai parar TODAS as vezes que o valor da variável mudar, não importando para qual valor ela for alterada. Na lógica da parte de baixo, vai parar uma vez para cada linha.

Você pode colocar uma condição ali para o programa parar na hora que a variável atingir a condição. Mas lembre-se que ele vai parar após a variável atingir a condição. No caso ali, vai parar na linha 39.

E é isso!

Os novos watchpoints e as comparações de tabelas ajudam pra caramba. As ferramentas especiais servem para análises mais complicadas, como de estouro de memória e etc. Fica inviável cobrir tudo que cada ferramenta faz por aqui, mas a idéia é uma só: **fuçar**.

A parte 3 vai fechar a sequência, com algumas coisinhas a mais. Se quiser ver alguma coisa, comente aí que eu procuro!

Abraços!

Ps.: Watchpoint requested by Vitão! NERDS!

Comentários

Sérgio Fraga — 22/03/2014 17:15

Boas,

Eu também ando a investigar o novo debugger e li um artigo muito interessante sobre o tópico.

Aconcelho vivamente!

Aqui está o link:

<http://sap4u.org/media/userfiles/articles/pdf/26.pdf>

Depois podes adicionar ao teu post mais uma dica sobre “watchpoint clones” 😊

Deixo uma dica que passei ao abapinho que também é interessante:

<http://abapinho.com/2014/03/fecha-a-janela-encalhada/>

Abraço de Portugal

Sérgio

Victor Galindo Cleme — 04/04/2011 13:59

Caramba já precisei ler esse post umas 37 vezes e hj que fui ver que tem um Vitão Nerds rsrsrs aqui é assim olha pro que interessa e o resto não tem pressa! Hié Hié Issssssáááá

Mauricio Cruz — 05/04/2011 07:43

No mesmo dia que eu escrevi você falou sobre os watchpoints, por isso coloquei hahahahahaha.

NERDS!