

Zumbis no Hana – Parte 2 – O impacto nos programas de hoje em dia

09/10/2014 11:11

Na segunda semana do curso da OpenSAP, os instrutores tentaram mostrar qual o impacto de uma migração para o HANA nos programas de hoje em dia e explicaram quais ferramentas existem para ajudar-nos no processo de revalidação de códigos Z (para saber se nenhum deles “quebrará” com a implantação do HANA). Antes de eu deixar as minhas considerações (já que não curti muito essa semana não), vamos ao que foi explicado.

A pergunta do milhão: como fica o que já existe?

Essa é a pergunta que 64 em cada 33 ABAPers fazem ao escutarem falar sobre o HANA. A parte boa é que praticamente tudo que você já sabe (ou deveria saber) sobre guidelines e boas práticas de performance continuam valendo, ou seja, **nada de pensar que “só porque tá no HANA o SELECT em LOOP tá liberado”**. Continue seguindo os guidelines de sempre, obrigado.

Porém há algumas pequenas coisas que serão diferentes se o sistema tiver o HANA instalado:

- **SQL Nativo:** Manja aquele comando “EXEC SQL”? Não? Então olhe [este link](#). O SQL Nativo, como o próprio nome sugere, permite que você utilize comando **específicos** de um determinado banco de dados. Se você utilizou algum comando específico do banco de dados que foi substituído pelo HANA, aqueles comandos não irão mais funcionar após a migração (master of the obvious).
- **HINTs específicos do Banco de Dados:** Manja aquele “HINTS” que a galera tem mania de enfiar nos SELECTs, utilizando de muleta para arrumar uma seleção cagada? Então, o HANA vai ignorar esses HINTs que sugerem índices que só existiam no banco de dados substituído.
- **Partes onde você assumiu que o BD fosse se comportar de uma determinada forma:** Sem saber, algumas pessoas assumem que o banco de dados vai se comportar de um jeito, e fazem seu programa em função disso. Exemplo: assumir que os dados de retorno de uma tabela sempre virão ordenados pelos campo chave. Isso muda com o HANA.

Vamos analisar um pouco mais o ponto que trará maior impacto, ou seja, o problema da ordenação. No passado eu já vi algumas coisas bizarras acontecerem relacionadas a esse ponto, mas nunca havia entendido o motivo. Eu explico com código:

```
SELECT *  
FROM vbak  
INTO TABLE t_vbak.  
  
READ TABLE t_vbak WITH KEY  
  vbeln = '12345'  
BINARY SEARCH  
TRANSPORTING NO FIELDS.
```

E aí, o READ TABLE.. BINARY SEARCH acima funciona? Muita gente irá pensar “*é claro que funciona, VBELN é a chave da VBAK, portanto, os dados virão ordenados por VBELN*”. Mas a realidade não é bem assim não, camarada. No passado, eu já vi casos onde a tabela vinha desorganizada, sem controle nenhum de ordenação implícito. Nunca consegui provar esse pqp quando mudava de projetos, pois na maioria dos lugares a tabela sempre vinha ordenada pelo chave primária, ou seja, pelo VBELN. Mas por que isso acontece?

Descobri nesse curo que isso depende do banco de dados. Há certos BDs que entregam os dados para a aplicação com uma ordenação implícita, ou seja, a não ser que você especifique a ordenação pelo ORDER BY, o BD vai tentar entregar os dados ordenados pela chave primária. Mas nem todo BD funciona assim... e o HANA é um dos que não garantem essa ordenação.

Ao implantar o HANA, o BINARY SEARCH acima provavelmente falharia. A alternativa recomendada é adicionar a cláusula ORDER BY no SELECT, especificando a ordenação.

Pode parecer uma correção simples, mas pensa na dor de cabeça que isso trará para a galera que fizer as migrações. CANSEI de ver esse tipo de coisa em códigos Zs mundo afora. Eu mesmo já tive minha época de ignorar o SORT quando eu precisasse fazer algo com a chave primária, mas depois que vi aquele caso bizarro, nunca mais confiei... (na verdade, fiquei feliz por descobrir que eu realmente não estava maluco).

Prevejo hordas ABAPers dando manutenção em códigos, igual aquelas hordas que corrigiam erros de unicode. Preparem-se!

Mas cara, vai ser impossível analisar tudo que já existe!

É exatamente por isso que a SAP vai lhe entregar ferramentas para ajudar nessas análises. Fica aqui a minha maior crítica a segunda semana do curso: muito blablabla sobre as ferramentas de análise, e poucos exemplos sobre como, de fato, o sistema irá comportar-se após a HANA.

Basicamente, a SAP adicionou algumas novas verificações no Code Inspector, e criou algumas ferramentas para ajudar no monitoramento dos acessos ao banco de dados. Já tem SE30, SAT, ST05... mas eles preferiram criar novas ferramentas, recriando as arquiteturas para que não existam grandes impactos em análise de ambientes produtivos.

- **Validações Estáticas:** Uma pancada de novos checks do Code Inspector, que eles estão mostrando como parte do [ABAP Test Cockpit](#), já integrado no Eclipse para fazer as validações. Todos esses checks visam alertar os desenvolvedores sobre os pontos de atenção e erros nos códigos que serão utilizados com o HANA. Essa é uma ferramenta que deveria ser utilizada ANTES da migração, para sanar os problemas antes do HANA ser implantado.
- **SQL Monitor:** A ST05 não foi feita para analisar todo o ambiente, ela é orientada a análises pontuais (e consome bastante recursos). Com o SQL Monitor, você pode ligar a análise em PRD por algum tempo e depois extrair esses dados e avaliar os pontos onde a performance está ruim. Uma coisa bacana é que você pode extrair esses dados de PRD e importar no DEV ou QAS para ajudar nas correções. Ela foi desenvolvida para não criar o caos na produção, já que o consumo de recursos para fazer as análises é mínimo. Você encontra uma descrição detalhada da ferramenta [neste link](#).
- **SQL Performance Tuning Worklist:** E daí você pode pensar "putz, mais ferramentas separadas?". A SAP criou essa ferramenta de nome enome para combinar as funcionalidades das Validações Estáticas e do SQL Monitor. Afinal, para que simplificar, não é verdade?

Com esse monte de ferramentas, ABAPer nenhum vai deixar seu código com baixa performance e incompatível com o HANA!

...

Certo?

Parafernália novas

Ficou um pouco desconexo com o resto do tema da semana, mas mostraram algumas novas features das versões mais novas do ABAP quando integrado ao HANA.

Um novo formato de Search Help foi implantado. Uma imagem vale mais do que mil palavras para explicá-lo:

Auto-complete with steroids? pic.twitter.com/nmhfBPINnj

— Flávio Furlan (@furlan) [October 8, 2014](#)

Você configura esse novo tipo de Search Help pela SE11 mesmo. Funciona bem, achei bacana e para o usuário é uma mão na roda!

No caso do ALV, eles mostraram o **ALV List Viewer with Integrated Data Access**, uma forma de criar e mostrar ALVs sem a lógica do SELECT to internal table > cria o container > cria a classe > bla,bla,bla,bla...

Basicamente o que você estiver vendo é o que o SAP acabou de buscar no banco de dados. Conforme você dá um PageDown/PageUp, o sistema vai até o banco de dados, busca os dados da nova página e retorna isso para a tela. É completamente diferente do que fazemos hoje em dia. [Aqui](#) tem uma comparação interessante entre o modo "IDA" e o modo "Clássico", e mostra o quão diferente é essa nova abordagem. Um vídeozinho firmeza que mostra isso com mais detalhes:

Pensamentos randômicos

Chegamos ao final de mais uma semana, e com ela surgem mais pensamentos randômicos.

- Posso estar viajando, mas essa história de usar um monte de coisas "nativas" do HANA vai exigir profissionais que conheçam muito bem como o HANA funciona. Pode ser que, no futuro, tenhamos mais uma camada de

complexidade no projetos: o HANA Developer. O key user manda para o funcional, que manda para o ABAP, que pede para o cara do HANA fazer umas paradas....

- Sim, o cara do ABAP também pode fazer a parte de HANA. Mas acredito que, a longo prazo, o mais provavel é que sejam trabalhos especializados feitos por pessoas diferentes, principalmente se levarmos em consideração como funciona a dinâmica de estudo no nosso mercado.
- Ah, e se tiver o Fiori fica ainda mais legal: O key-user fala com o funcional, que fala sei lá com quem, que fala com o cara do Frontend, que fala com o ABAP, que fala com o cara de HANA... 😊

E assim termina a segunda parte dessa saga. Semana que vem tem mais!

Dúvidas? Pensamentos randômicos? Eu entendi alguma coisa errada e mereço ser xingado? Deixe um comentário!

Abraços a todos que acharam que o SELECT em LOOP seria o futuro!