

DESENVOLVENDO APLICAÇÃO EM LINGUAGEM ABAP

MARCONDES JUNIOR, Mario¹; TIENGO MARTINS, João Paulo¹; SOUZA REIS, José Cláudio de².

⁽¹⁾ Acadêmicos do Curso de Ciência da Computação, UNIFENAS, câmpus Alfenas.

⁽²⁾ Professor do Curso de Ciência da Computação, UNIFENAS, câmpus Alfenas.

RESUMO

Este projeto tem por finalidade demonstrar o desenvolvimento de um simulador ERP utilizando a linguagem ABAP demonstrando suas possibilidades e limitações. A linguagem ABAP foi desenvolvida para implementação e desenvolvimento do software da empresa Alemã SAP. A linguagem pode ser implementada proceduralmente como também orientada a objetos, cada função a ser criada como: criação de tabelas, criação de programas, criação de funções e outras mais, são executadas a partir de transações, assim acontece também na criação dos programas acessados pelo usuário, cada programa só pode ser acessado por uma transação ou por uma função de menu.

O Simulador ERP tem o objetivo de demonstrar as funcionalidades, estruturas e sintaxe da linguagem ABAP em um projeto que reflete as funcionalidades de sistema ERP convencional, onde os departamentos são interligados e vincula seus dados em uma mesma base.

Foi desenvolvido tabelas para manutenção de quatro departamentos que são eles: departamento requisitante, departamento de compras, departamento fiscal e departamento de contas a pagar. O requisitante cria uma requisição com o material desejado assim o departamento de compras identifica a mesma e cria um pedido de compra com o fornecedor desejado, na próxima etapa o departamento fiscal registra a fatura para assim o departamento de contas a pagar fazer o pagamento e finalizar o ciclo.

1 INTRODUÇÃO

1.1 Contextualização

As linguagens de computação foram desenvolvidas após a introdução dos compiladores na década de 50. Veio para aprimorar até então a atividade de mudar fios de local ou perfurar cartões, recebendo o auxílio de diversas linguagens que introduziram facilidades e recursos, tornando assim a programação mais fácil, dinâmica e suscetível a erros.

Linguagem de programação são métodos padronizados para expressar instruções para um computador. É um conjunto de regras sintáticas e semânticas usadas para definir um programa de computador. Uma linguagem permite que um programador especifique sobre quais dados um computador vai atuar, como esses dados serão transmitidos ou armazenados e quais ações devem ser tomadas sob várias circunstâncias.

1.2 Objetivos

O presente trabalho tem por objetivo realizar um levantamento bibliográfico da linguagem ABAP e desenvolver aplicações para a mesma, visando mostrar suas possibilidades e limitações.

1.3 Justificativas

ABAP (Advanced Business Application Programming) é uma linguagem única e própria desenvolvida pela SAP (Systeme Anwendungen und Produkte in der Datenverarbeitung), uma das maiores empresas de software do mundo. O objetivo original da linguagem era ser uma ferramenta de programação suficientemente simples para ser usada pelos usuários finais.

1.4 Hipóteses

O aprendizado da linguagem ABAP é suficientemente simples para ser usada por usuários finais.

A linguagem ABAP apresenta os recursos existentes nas outras linguagens de programação atual.

A linguagem ABAP é uma linguagem orientada a objetos ou procedural.

1.5 Estrutura do projeto

O primeiro capítulo trata da introdução do trabalho, sobre as linguagens de programação, o objetivo da pesquisa, a escolha do tema e levantamos algumas hipóteses a serem trabalhadas ao decorrer deste trabalho. O capítulo dois apresenta o surgimento da linguagem de programação, os tipos de linguagens, a linguagem ABAP e suas características. O capítulo três descreve-se os materiais e métodos utilizados na pesquisa. O capítulo quatro mostra o desenvolvimento de uma aplicação usando a linguagem ABAP. O capítulo cinco descreve os resultados e discussão. No seis é apresentada a conclusão desta pesquisa.

2 REFERENCIAL TEÓRICO

2.1 História das linguagens de programação

Tudo começou na década de 30, com os primeiros computadores elétricos. Em 1948, Konrad Zuse publicou sua criação, a linguagem de programação Plankalkül. Na época, ainda não tinha muita utilidade, então foi esquecida. Antes da programação passar para o computador, eram usados cartões de papelão que eram perfurados, criando códigos. (PACIEVITCH, 2013).

2.2 A linguagem de programação ABAP

A Empresa SAP criou sua própria linguagem de programação ABAP. Em alemão ABAP quer dizer (*Allgemeiner Berichtsaufbereitungsprozessor*) ou seja preparador genérico para preparação de relatórios, sendo posteriormente batizado como *Advanced Business Application Programming*. Foi uma linguagem de quarta geração criada pela SAP em 1980 apenas como uma linguagem de preparação de relatórios do SAP R/2. Atualmente é a linguagem mais utilizada para o desenvolvimento de programas acoplados ao SAP, também sendo possível na plataforma *Netweaver* o desenvolvimento em Java (*Web Dynpros*). (WIKIA, 2013).

O objetivo original da linguagem era ser uma ferramenta de programação suficientemente simples para ser usada pelos utilizadores finais. A linguagem ABAP é uma linguagem que necessita de conhecimentos avançados de programação e é principalmente utilizada pelos programadores da SAP que desenvolvem o software e por consultores informáticos que adaptam o software às necessidades dos clientes. O ABAP é a principal linguagem utilizada no software cliente-servidor SAP R/3. Em 1999, com a versão R/3 4.5, a SAP lançou uma versão de ABAP que suporta programação por objetos. (WIKIA, 2013).

ABAP guarda muitas semelhanças com COBOL e Fortran. Inicialmente uma linguagem estruturada, ABAP exigia a escrita de palavras reservadas (do, while, etc.) em maiúscula, o que reduzia a legibilidade. Atualmente, em sua quarta versão,

ABAP agora suporta orientação a objetos (Object ABAP) e permite escrita do código em letras minúsculas. (WIKIA, 2013).

ABAP se comporta como Java, ou seja: o código-fonte é "compilado" para uma linguagem intermediária, que por sua vez é interpretado e traduzido para o código de máquina. Os programas ABAP são executados em um runtime - o núcleo do SAP - que também residem neste. Ou seja, em ABAP não se programa em arquivos separados (como em C ou Java), e sim "direto" sobre o ambiente. (WIKIA, 2013).

2.2.1 Estruturas da linguagem de programação ABAP

Podemos citar algumas características como: versatilidade R/3 é uma “*meta-application*” que é totalmente configurável e possui três camadas. R/3 é um sistema interpretativo, portanto, nunca existe o ciclo “*shutdown-change-recompile-start*”. A logística do software de desenvolvimento permite centenas de desenvolvedores trabalharem ao mesmo tempo e a logística disso pode ser escondida pela linguagem. O processo de mudança deve ser considerado e suportado por certas características da linguagem (geralmente resultando em alguns desvios das fortes políticas do sistema). Você não pode assumir que os desenvolvedores sempre podem fazer mudanças globais sincronizadas. (PRIDAYANA, 2013).

Os Objetos ABAP são baseados em classes. Classes são peças do código do programa que descrevem objetos através da definição dos seus componentes. Componentes típicos de objetos são atributos (dados), que descrevem o estado de um objeto e funções, que descrevem o comportamento do objeto. A semântica de referência para objetos. ABAP é completamente “*value-based*”. Não existem referências na linguagem e um movimento sempre cópia seus valores (mesmo com campo-símbolo). Objetos, por outro lado, somente podem ser acessados por referência. Os objetos ABAP têm seu próprio gerenciamento de *storage* (*garbage collection*) e um conjunto de ferramentas como *Class Builder*, *Class Browser* e *debuggersupport*. (PRIDAYANA, 2013).

2.2.2 Características de um programa ABAP

Segundo (COSTA, 2010), programas executáveis podem ser executados diretamente a partir do ABAP editor ou de uma transação que o programa esteja associado. Em geral programas executáveis são para criação de telas de seleção e relatórios, mas também, podem criar qualquer tipo de aplicações de diálogo.

Um programa do tipo *Module Pool* somente pode ser executado a partir de uma transação ou uma função de menu. Por exemplo, você cria um programa do tipo *Module Pool*, este relatório somente será executado se você criar uma transação, e associar esse programa à esta transação. (COSTA, 2010).

3 MATERIAL E MÉTODOS

3.1 Etapas do trabalho

Este trabalho está dividido em três etapas. A primeira etapa baseia-se em um levantamento bibliográfico exploratório referente à linguagem ABAP. A segunda etapa define-se o desenvolvimento de uma aplicação que gerencia o abastecimento de materiais de consumo de uma empresa, completando todo o estágio desde a requisição do que será comprado pela produção, passando a requisição para o departamento de compras criar o pedido de compras, passando pelo departamento fiscal registrar o documento fiscal e pôr fim a criação da compensação pelo departamento de contas a pagar. A terceira etapa analisa as possibilidades e limitações da linguagem de programação ABAP.

3.2 Ferramentas e tecnologias utilizadas

O aplicativo desenvolvido para a plataforma SAP foi implementado em ambiente Windows 7 64x, utilizando a IDE (interface de desenvolvimento) MiniSAP 7.03 EP4. A IDE MiniSAP foi adquirida através do site oficial da SAP criando um cadastro para o uso da mesma somente para estudo, após a instalação é necessário solicitar uma licença de uso válida por noventa dias com inspiração do acesso após esta data.

Os testes das aplicações foram realizados apenas dentro do MiniSAP conforme especificação da IDE.

Foi desenvolvida uma aplicação que simula um sistema ERP convencional para demonstrar as funcionalidades da linhagem ABAP.

4 APLICAÇÃO ABAP

4.1 Sistema Simulador ERP

O Simulador ERP tem o objetivo de demonstrar as funcionalidades, estruturas e sintaxe da linguagem ABAP em um projeto que reflete as funcionalidades de sistema ERP convencional, onde os departamentos são interligados e vincula suas informações em uma mesma base de dados.

O caso de uso utilizado pelo requisitante ou a produção, consiste na manutenção dos materiais e das requisições, como mostra a.

O caso de uso do departamento de compras faz a manutenção de fornecedores, pedidos de compras e visão para consulta a requisições e materiais.

O caso de uso do departamento fiscal faz a manutenção das operações de faturas e exibição dos fornecedores e pedidos de compras.

O caso de uso de contas a pagar faz a manutenção nas operações de pagamento e tem visão para os atributos das faturas.

O diagrama de classe mostra que as tabelas de material e fornecedor são agregadas da tabela de requisição de compra que está agregada a tabela de pedido de compra que está agregada a tabela de faturas que está agregada a tabela de contas a pagar todas as classes tem os métodos de exibição, modificação, exclusão e inserção.

4.2 Ambiente de desenvolvimento

A Aplicação desenvolvida simula a dinâmica de um sistema ERP convencional que tem por finalidade integrar todos os processos administrativos de uma empresa e centralizar as informações e atividades de ambiente empresarial, organizando, agilizando e mantendo a confiabilidade das informações em todos os níveis de negócios. Para isso o simulador desenvolvido abrange os departamentos financeiro, fiscal/contábil, compras e os demais solicitantes de requisições.

4.3 Características gerais do código

No SAP cada criação de tabelas, funções, procedimentos, classes, programas e outros são criados através de transações e não codificado. A criação das tabelas no banco de dados é feita através da transação SE11 ABAP Dictionary Maintenance, em sua tela inicial é fornecido o nome da tabela e clicando no botão “Create”, aqui também pode ser visualizada a tabela pelo botão Display e Modificado pelo botão Change.

Em ABAP não é possível definir diretamente um tipo a um atributo da tabela, assim é definido um elemento de dado que faz referência para este tipo, por exemplo, o atributo “DESCRIÇÃO” de elemento de dado “ZDESCR” faz referência ao tipo char definido para 30 caracteres conforme projeto de dados.

Para que um objeto seja enxergado em qualquer lugar dentro do SAP é necessário que este objeto seja ativado, para isso após qualquer alteração, o procedimento a seguir é SAVE, CHECK que fornecerá qualquer warning ou erro e posteriormente ACTIVATE que deixará o objeto disponível para execução.

4.4 Interfaces gráficas

Em Abap as interfaces gráficas podem ser construídas através dos construtores nos próprios códigos ou graficamente arrastando os objetos. Sendo que arrastando os objetos fica diferente dos métodos convencionais que se arrasta depois codifica suas classes, aqui o código deve estar pronto primeiro para ligar aos objetos posteriormente. Assim a montagem da interface gráfica através dos construtores no próprio código fica mais prático e simples e pode-se atribuir as tabelas apenas usando o comando *LIKE*.

Os programas dentro do SAP são chamados de *REPORTS*, através da transação SE38 tem-se acesso a criação dos *REPORTS*. São através deles que a manutenção das tabelas é aplicada.

No SAP todos os programas *STANDARTS*, ou seja, criado fora de seu escopo original deve conter a letra “Z” no seu início.

5 RESULTADOS

O presente trabalho teve o intuito de explorar o ambiente e a linguagem criados pela SAP para desenvolvimento de seu software, foi aplicado uma metodologia exploratória para o desenvolvimento deste projeto.

Foi demonstrado a criação do programa principal e a criação das transações responsáveis pela manutenção do sistema as sintaxes de programação e os processos internos da SAP que não são parecidos com outras linguagens convencionais.

Foi obtido dificuldade na parte da licença da IDE MiniSAP, pois esta versão disponível para estudo permite que o usuário tenha acesso a mesma por noventa dias.

Por se tratar de um programa privado e de acessos limitados, foi obtido uma certa dificuldade para localizar material didático do mesmo.

6 CONCLUSÃO

Conclui-se que a linguagem ABAP contém alguns nomes de comandos parecidos com as linguagens convencionais. As formas e os ambientes que são usados diverge de das sintaxes das linguagens convencionais. Seu aprendizado não é complicado, apenas não tão comum quanto as outras linguagens de programação convencionais.

A linguagem ABAP contém todos recursos oferecidos em outras linguagens, cada função ou serviço do projeto são criados em uma transação diferente que atribui-se ao programa principal.

A linguagem ABAP é tanto procedural quando orientada a objetos, sendo que na sua mais nova versão que ora foi utilizada neste projeto utiliza-se quase tudo orientado a objetos. Podem-se criar os objetos nas interfaces arrastando os mesmos ou aplicando comando como foi utilizado neste projeto.

Nesta versão para estudo, foi encontrada certa dificuldade com a licença de uso do software sendo que após a aquisição do software é necessário pedir ao fabricante uma chave de acesso que inspira após 90 dias, devido a isto foi preciso implementar este projeto mais de uma vez.

REFERÊNCIAS

COSTA, Márcio Brenner. **ABAP passo a passo**: aprenda a principal linguagem de programação SAP de forma rápida, clara e definitiva. 1. ed. Brasília, BR: Clube de Autores, 2010.

DCA, Departamento de Engenharia de Computação e Automação. Disponível em: <<http://www.dca.ufrn.br/~affonso/DCA800/pdf/linguagens.pdf>>. Acesso em: 09 nov. 2013

DEVMEDIA, 10/03/2014. Disponível em: <<http://www.devmedia.com.br>>. Acesso em: 10 nov. 2013

DIGITALDEV – Pra quem gosta de programar. Disponível em: <<http://www.digitaldev.com.br/linguagens>>. Acesso em: 03 nov. 2013

HISTORIA das Linguagens de Programação. Disponível em: <<http://historiadaslinguagensdeprogramacao.blogspot.com.br>>. Acesso em: 03 nov. 2013

IC, Instituto de Computação. Disponível em: <<http://www.ic.unicamp.br/>>. Acesso em: 10 nov. 2013

PACIEVITCH, Yuri et al. INFOESCOLA, Navegando e Aprendendo. Disponível em: <<http://www.infoescola.com/informatica/historia-da-programacao>>. Acesso em: 09 nov. 2013

PRIDAYANA. Disponível em: <<http://ridayana.no.comunidades.net/index.php?pagina=1307981035>>. Acesso em: 03 nov. 2013

SAP Software & Soluções. Disponível em: <<http://global.sap.com/brazil/about/historico/index.epx>>. Acesso em: 09 nov. 2013

WIKIA. Disponível em: <http://pt-br.grupo10lp.wikia.com/wiki/Semin%C3%A1rio_1>.

Acesso em: 10 nov. 2013