

ESTUDO DE FERRAMENTAS DE MODELAGEM EM RELAÇÃO À UML 2.0

STUDY OF TOOLS FOR THE MODELING IN RELATION TO THE UML 2.0

*Sergio A. Tanaka**
*Thiago A. Herek***
*Simone S. Tanaka****
*Ruy T. Nishimura*****

RESUMO:

A necessidade de se ter uma linguagem unificada para comunicação e sincronização dos envolvidos em um projeto de desenvolvimento de software, estimulou a criação de uma linguagem padrão de modelagem para software orientado a objeto. A UML é considerada o auge de todos os esforços para padronizar as ferramentas utilizadas para expressar os conceitos do software. Atualmente é utilizada em todo mundo, em conjunto com diversas ferramentas de apoio ao desenvolvimento de Software. Este artigo utiliza a UML 2.0, para confrontar a capacidade de algumas ferramentas de modelagem disponíveis no mercado na aplicação desta linguagem.

PALAVRAS-CHAVE: UML, Ferramentas de CASEs

ABSTRACT:

The need for a unified language for communication and synchronization of those involved in a project of software development stimulated the creation of a standard language of modeling for software guided by an object. The UML is considered the summit of all efforts to standard the used tools to express the software concepts. Now on days it is used all around the world, together with a whole of support tools to the development of software. This article uses the UML 2.0 to face the capacity of some tools of modeling available in the market in the application of this language.

KEY-WORDS: UML, Tools of CASEs

1. INTRODUÇÃO

A grande importância e complexidade do software necessitam, que se tenha métodos de engenharia sólidos o suficiente para garantir a qualidade do produto final. O mesmo modo, também é preciso manter a equipe e todos os envolvidos atualizados com os requisitos de software e a análise do sistema.

Este artigo aborda a utilização da UML 2.0, quatro ferramentas de Modelagem foram analisadas, a IBM Rational Software Modeler 7.0, Sparx Enterprise Architect 6.5.802, o Borland Together for Eclipse 2006 R2 e o Poseidon for UML CE 4.2.1, comparando as características de cada ferramenta em relação a UML.

*Coordenador do curso de Pós-Graduação - Engenharia de Software com UML da UniFil, e-mail: sergio.tanaka@unifil.br

** Discente do curso de Especialização Engenharia de Software e UML em 2006, e-mail: thiagoh@yahoo.com

*** Docente do curso de Sistemas de Informação da UniFil, e-mail: simone.tanaka@unifil.br

**** Docente do curso de Engenharia de Software do curso de Pós Graduação da UniFil

No capítulo 2 são apresentados a linguagem UML e seus diagramas, no capítulo 3 são apresentados os conceitos de ferramentas de modelagem e as ferramentas utilizadas no trabalho. O capítulo 4 apresenta o estudo comparativo entre as ferramentas e finalmente no capítulo 5 as conclusões e trabalhos futuros.

2. UML

A linguagem UML é mundialmente conhecida e utilizada largamente para a visualização, especificação, construção e documentação de projetos de software.

A UML surgiu da unificação de outras linguagens gráficas de modelagem utilizadas na década de 80 e início da década de 90. Seu primeiro esboço, a versão 0.8, surgiu no final de 1994, e em 1997 na versão 1.1 passou a ser controlada pela OMG, um consórcio aberto de empresas.

De acordo com Pender (2004), a versão 2.0 é uma melhoria substancial da arquitetura básica, acertando muitas das definições fundamentais e melhorando o alinhamento com outras tecnologias importantes, patrocinadas pelo OMG.

A UML 2.0 descreve 13 diagramas que são divididos em três tipos: Diagramas de Estrutura, Diagramas de Comportamento e Diagramas de interação:

- diagramas da Estrutura: Diagrama da Classe, Diagrama do Objeto, Diagrama Componente, Diagrama de Estrutura Composta, Diagrama do Pacote e Diagrama de Distribuição.
- diagramas do Comportamento: Diagrama do Caso do Uso, Diagrama da Atividade e Diagrama da Máquina do Estado.
- diagramas da Interação: Diagrama de Seqüência, Diagrama de Comunicação, Diagrama de Sincronismo e Diagrama Visão Geral da Interação.

A UML recebeu novos diagramas após a versão 1, outros receberam algumas mudanças, na Tabela 1 estão os diagramas da UML 2.0 e seus objetivos.

Tabela 1 – Diagramas Oficiais da UML (FOWLER 2004, p. 33).

Diagrama	Objetivo	Versões
Atividades	Comportamento Procedimental e paralelo	UML 1
Classes	Classe, características e relacionamentos	UML 1
Comunicação	Interação entre os objetos; ênfase nas ligações	Diagrama de colaboração da UML 1
Componentes	Estrutura e conexão de componentes	UML 1
Estrutura Composta	Decomposição de uma classe em tempo de execução	UML 2
Distribuição	Distribuição de artefatos nos nós	UML 1
Visão Geral da Interação	Mistura de Diagrama de Seqüência e de Atividades	UML 2
Objetos	Exemplo de configurações de instâncias	UML 1
Pacotes	Estrutura hierárquica em tempo de compilação	UML 1
Seqüência	Interação entre objetos; ênfase na seqüência	UML 1
Máquina de estado	Como os eventos alteram um objeto no decorrer de sua vida	UML 1
Sincronismo	Interação entre objetos; ênfase no sincronismo	UML 2
Casos de Uso	Como os usuários interagem com um sistema	UML 1

124

3. FERRAMENTAS DE MODELAGEM

Pressman (1995) define ferramentas de engenharia de software como ferramentas que proporcionam apoio automatizado ou semi-automatizado aos métodos de planejamento e estimativa de projeto, análise de requisitos de software e de sistemas, projeto da estrutura de dados, arquitetura do programa e algoritmo de processamento, codificação, teste e manutenção. Quando as ferramentas são integradas de forma que a informação criada por uma ferramenta possa ser usada por outra, é estabelecido um sistema de suporte ao desenvolvimento de software, chamado engenharia de software auxiliada por computador (CASE – Computer-Aided Software Engineering).

De acordo com Pender (2004), essas ferramentas eram consideradas como a “bola de prata” da época, pois prometiam geração de código completa a partir de modelos, de modo que os modeladores de negócios poderiam projetar o *workflow* dos negócios e apertar um botão para gerar as aplicações de suporte.

Sabe-se que tal promessa não foi cumprida, e segundo Pender (2004), por esse motivo o termo CASE ficou um tanto manchado, e tem sido evitada a utilização desse termo nas novas ferramentas de modelagem, porém com o passar dos anos observou-se que tais ferramentas eram extremamente importantes na ajuda aos usuários para analisar e projetar seus sistemas.

Por esse motivo ferramenta de modelagem tornou-se mais apropriado para descrever essas ferramentas, já que as mesmas auxiliavam na modelagem, captura e disseminação de informações do projeto e não geravam o código como inicialmente prometido.

Segundo Pender (2004) a tecnologia avançou tanto que a geração completa de código de uma ferramenta de modelagem novamente está sendo examinada como uma possibilidade real. O alinhamento direto do código orientado a objeto com modelos orientados a objeto, a maturidade contínua da especificação UML e o advento de serviços WEB e linguagens de execução comercial, tornaram a geração completa de código a partir de modelos de negócios e tecnológicos um objetivo imaginável.

As ferramentas de modelagem modernas geram algumas partes de código, o que realmente automatiza a construção de uma parte do sistema, porém por mais especificado e detalhado que seja a modelagem do sistema, o código gerado ainda é limitado a algumas especificações básicas dos modelos.

Nas próximas seções são apresentadas algumas informações das ferramentas utilizadas neste trabalho.

3.1 Rational Software Modeler

A Rational Software Modeler (RSM, 2006), é uma ferramenta de modelagem da fabricante Rational (IBM), atualmente (Dezembro – 2006) está na versão 7.0.

De acordo com as especificações da IBM, a versão 7.0 do RSM, que foi objeto de estudo deste artigo, possui suporte para modelagem na UML 2.1 na análise e projeto usando Casos de Uso, Classe, Seqüência, Atividade, Diagramas de Estruturas Compostos, Máquina de Estado, Comunicação, Componente e Distribuição e suporte a transformação MDA.

125

3.2 Sparx Enterprise Architect

O Enterprise Architect - EA (SPARX, 2006) é uma ferramenta de modelagem da fabricante australiana Sparx Systems e atualmente (Dezembro – 2006) está na versão 6.5.

De acordo com as especificações da Sparx Systems, a versão 6.5, que foi utilizada neste artigo, possui suporte aos diagramas da UML 2.1 e suporte à transformação MDA.

3.3 Borland Together

Together (BORLAND, 2006) é uma ferramenta de modelagem da fabricante Borland e atualmente (Dezembro – 2006) está na versão 2006 R2, neste artigo foi utilizado a versão “Together 2006 Release 2 for Eclipse”, a Borland é muito conhecida na área de desenvolvimento devido a ferramenta IDE Delphi.

De acordo com a fabricante, Together é uma plataforma de modelagem visual baseada em papéis que amparam arquitetos, desenvolvedores, designers de UML, analistas de processos de negócios e modeladores de dados na criação acelerada de aplicações de software de alta-qualidade.

3.4 Gentleware Poseidon for UML

O Poseidon (GENTLEWARE, 2006) é uma ferramenta de modelagem da fabricante Gentleware, empresa fundada em 2000 em Hamburgo na Alemanha. A empresa possui 5 versões para a ferramenta de modelagem, entre elas uma voltada para usuários não comerciais chamada de Community Edition (CE).

De acordo com o fabricante, a versão CE, é perfeita para estudantes, iniciantes e usuários não comerciais, suporta a UML 2.0 e engenharia reversa com JAVA.

4. ESTUDO COMPARATIVO

Além dos treze diagramas da UML foi avaliada a geração de códigos a partir da modelagem e a engenharia reversa nas linguagens JAVA, C++ e Visual Basic, com o objetivo de constatar o suporte a linguagem UML e avaliar os recursos largamente utilizados pela comunidade de engenharia de software.

A Tabela 2 apresenta as principais informações sobre as ferramentas testadas neste artigo.

Tabela 2 - Resumo das Ferramentas

Fabricante	Ferramenta	Versão
IBM Rational	Software Modeler	7.0
Borland	Togheter for Eclipse	2006 R2
Sparx	Enterprise Architect	6.5.802
Gentleware	Poseidon for UML	CE 4.2.1

No item “facilidade de uso” foi avaliado o desempenho e o ambiente de trabalho das ferramentas levando em conta o suporte multilíngüe, ícones intuitivos, organização das janelas, opções e o conteúdo de ajuda.

Para o suporte a UML 2.0 foram avaliados os 13 diagramas, com os novos recursos da linguagem. Enquanto que na geração de código foi avaliada a codificação a partir do diagrama de classe.

Na avaliação do recurso engenharia reversa foram utilizados os próprios códigos gerados pelas ferramentas para gerar os modelos em UML.

A Tabela 3 demonstra o resultado do estudo entre as ferramentas.

Tabela 3 - Comparativo Entre Ferramentas

Ferramentas	Diagramas e Capacidades																											
	Diagramas	Casos de Uso	Classes	Seqüência	Objetos	Estrutura Composta	Pacotes	Máquina de Estados	Atividade	Comunicação	Componentes	Distribuição	Sincronismo	Visão Geral da Interação	Geração de Códigos	Java	C++	Visual Basic	Engenharia Reversa	Java	C++	Visual Basic	Multiplataforma	Windows 2k ou mais	Linux Kernel 2.4	Outros Requisitos	Facilidade de Uso	
Rational Software Modeler		1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	3		1	1	3		1	1	1	1	1
Borland Together for Eclipse		1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	3		1	1	3		1	1	1	3	3
Sparx Enterprise Architect		1	1	1	1	1	1	1	1	1	1	1	1	1		1	1	1		1	1	1		1	1	1	1	1
Poseidon for UML		1	1	1	1	1	1	1	1	1	1	1	1	1		1	3	3		3	3	3		1	1	1	2	2

Legenda	
1	Suporta / Bom
2	Suporta Parcialmente / Regular
3	Não Suporta / Ruim

5. CONCLUSÕES E TRABALHOS FUTUROS

As ferramentas de modelagem estão evoluindo rapidamente para atender as novas versões da UML, é possível constatar que todas as ferramentas estudadas foram capazes de suportar completamente a UML 2.0 com seus 13 diagramas.

Além da linguagem UML, foram confrontadas as capacidades de transformação de códigos, recurso que auxilia e economiza muitas linhas de código para os analistas de sistema, a engenharia reversa e também a capacidade multiplataforma das ferramentas.

Após a comparação entre as ferramentas, pode-se verificar que as diferenças são pequenas. Mesmo em ferramentas de baixo custo a qualidade nos recursos para a modelagem do sistema se mantém da mesma maneira, como nas ferramentas mais conceituadas.

A transformação e a engenharia reversa podem ser consideradas satisfatórias, a maioria das ferramentas atende as linguagens largamente usadas no mundo. No caso da ferramenta Poseidon alguns destes recursos são limitados pela versão “Community Edition”, sendo que a fabricante Gentleware possui outras versões mais completas, voltadas para profissionais.

Outro requisito avaliado foi a capacidade multiplataforma, para qual todas as ferramentas possuem no mínimo versões para duas plataformas distintas. Foram avaliadas as capacidades para as plataformas Windows e Linux, porém algumas ferramentas suportam outras plataformas como Solaris, Mac entre outras.

Verificou-se que as ferramentas de modelagem estão evoluindo em conjunto com as novas versões da UML, possibilitando aos analistas manterem seus projetos atualizados com a última versão da linguagem padrão de modelagem. Este papel é muito importante, pois os esforços da OMG e seus colaboradores em relação à criação e a manutenção da UML seriam em vão se não houvesse ferramentas capazes de atender as necessidades dos analistas na utilização desta linguagem.

O estudo deste trabalho possibilita a iniciação de confrontos com novas ferramentas de modelagem que estão no mercado, mas seria atraente avaliar a comparação da transformação MDA, a geração de códigos em outras linguagens, e a modelagem UML com as próximas versões que certamente surgirão.

6. REFERÊNCIAS

BOOCH, Grady; FOWLER, Martin; KOBRYN, Cris. *UML Essencial*. São Paulo: Bookman, 2005.

BORLAND, Together. *Borland Together Technologies*. Disponível em: <<http://www.borland.com>>. Acesso em 27 de dezembro de 2006.

GENTLEWARE, Poseidon. *About the Community Edition*. Disponível em: <<http://www.gentleware.com>>. Último acesso em 20 de dezembro de 2006.

PENDER, Tom. *UML a Bíblia*. Rio de Janeiro: Campus, 2004.

PRESSMAN, Roger S. *Engenharia de Software*. São Paulo: Makron Books, 1995

RATIONAL. *Rational Software Modeler*. Disponível em <<http://www-306.ibm.com>>. Acesso em 27 de dezembro de 2006.

SPARX, Enterprise Architect. *UML Design Tools*. Disponível em <<http://www.sparxsystems.com.au>>. Acesso em 27 de dezembro de 2006.

UML, Unified Modeling Language. *Introduction to OMG UML*. Disponível em <<http://www.omg.org>>. Acesso em 22 de dezembro de 2006.

UML, Unified Modeling Language. *UML Resource Page*. Disponível em <<http://www.uml.org>>. Acesso em 22 de dezembro de 2006.