

Projeto 6.08

Documentação Essencial para Manutenção de Software

Sérgio Cozzetti Bertoldi de Souza, Nicolas Anquetil, Káthia Marçal de Oliveira

1. Introdução

Existe uma grande dificuldade das empresas que desenvolvem software cumprirem suas metas de prazo e orçamento. Segundo o Standish Group (2004), somente 16,2% dos projetos de software terminaram dentro do prazo e orçamento estimados. Muitas vezes o prazo de entrega pode significar a sobrevivência ou não de uma organização. Como uma tentativa de minimizar este problema, estudos são realizados na área da Engenharia de Software. Neste contexto, a abordagem da documentação tem grande importância, no qual o prazo de entrega do projeto pode variar de acordo com a quantidade e qualidade dos artefatos de software a serem entregues.

Ambler (2001) sugere duas razões básicas para documentar, uma serve para auxiliar a comunicação durante o projeto e outra para auxiliar o entendimento nas atividades de manutenção. Segundo Wong *et al* (1995), mais de 50% do trabalho de evolução do software é dedicado ao entendimento do programa e a tarefa mais difícil da manutenção de software é o entendimento da estrutura do sistema.

Por outro lado, os métodos ágeis vêm ganhando popularidade por causa da rapidez de entrega. Um exemplo é o Extreme Programming (XP), que tem uma visão muito diferente e vem se tornando cada vez mais popular (Briand, 2003). Se fielmente aplicado, não existe nenhum documento de análise e projeto.

A falta ou o excesso de documentação pode colocar em risco o desenvolvimento e a manutenção do software. Encontrar um ponto de equilíbrio na quantidade de documentação gerada é de fundamental importância. Este projeto buscou identificar na prática quais são os artefatos de software mais utilizados na atividade de obter o entendimento do sistema na fase de manutenção.

A seguir serão apresentados alguns aspectos relacionados à documentação de software (seção 2) e a estratégia utilizada para identificar na prática a real utilização dos artefatos de software na manutenção. As conclusões e trabalhos futuros são abordados na seção 4.

2. Documentação de Software

A documentação tem fundamental importância para a Engenharia de Software. Vários estudos (Rajlich, 2000, Tilley e Müller, 1991, Tilley e Huang, 2003) tem sido realizados para minimizar os problemas em torno da documentação de software: documentação desatualizada e de baixa qualidade, processos de documentação dispendiosos e caros, documentação em abundância e sem propósito, dificuldade de acesso, entre outros. Alguns pesquisadores propuseram o uso de hipertexto para facilitar o acesso a documentação. Tilley e Müller (1991) propuseram combinar documentação e código fonte de uma maneira fácil e eficiente, utilizando ferramenta de hipertexto. Rajlich (2000) propôs a adoção de uma estratégia de redocumentação incremental e oportunística na qual a compreensão obtida na manutenção é armazenada, por meio de anotações, em um sistema de hipertexto baseado na Web. Tilley e Huang (2003) definiram um modelo de maturidade para documentação de software com foco no produto (DMM – Documentation Maturity Model), baseado no CMM (Capability Maturity Model) do SEI (Software Engineering Institute), para estabelecer parâmetros de avaliação da qualidade da documentação.

De forma geral, os estudos sobre documentação de software visam solucionar problemas de falta de atualização, dificuldade de acesso, falta de qualidade, desorganização, documentação desnecessária e indicam para soluções simples, com o apoio de ferramentas e que sirvam a um propósito.

3. Estratégia para Identificação da Documentação Essencial na Manutenção

Para identificar qual a documentação essencial para a manutenção de software duas pesquisas foram realizadas.

A primeira pesquisa (Souza *et al*, 2004) buscou identificar os artefatos de software que auxiliam no entendimento do sistema na fase de manutenção. Foi elaborado um questionário direcionado a profissionais com experiência em manutenção de software no qual cada entrevistado indicava o grau de importância de cada artefato para a manutenção. A Figura 1 apresenta parte do questionário que era iniciado com a caracterização do avaliador. A maioria dos profissionais entrevistados foram analistas de sistemas (65%), seguidos por gerentes (25%), programadores (7%) e consultores (3%). Como resultado da pesquisa pôde-se constatar que os artefatos de maior importância envolvem:

- requisitos – referentes aos artefatos lista de requisitos e sua descrição na análise estruturada, e digrama de caso de uso e sua especificação na análise orientada a objetos;

- modelo de dados – referentes aos artefatos modelo lógico e físico de dados na análise estruturada, os mesmos artefatos associado ao diagrama de classe que é a base para a definição desses modelos na análise orientada a objetos; e,
- código, referente ao código fonte e comentários no mesmo nas duas abordagens.

Observou-se, ainda, que código fonte e seus comentários são artefatos mais importantes, o que é bastante aceitável se tratando de manutenção.

Artefatos		Conheço				Não Conheço
		1	2	3	4	
Levantamento						
Estruturada	Lista de Requisitos	<input type="radio"/>				
	Diagrama de Contexto	<input type="radio"/>				
	Descrição dos Requisitos	<input type="radio"/>				
Orientação a Objetos	Documento de Visão	<input type="radio"/>				
	Diagrama de Caso de Uso	<input type="radio"/>				
Estruturada e OO	Modelo Conceitual de Dados	<input type="radio"/>				
	Glossário	<input type="radio"/>				
Análise						
Estruturada	Funções Derivadas dos Requisitos	<input type="radio"/>				
	Diagrama Hierárquico de Funções	<input type="radio"/>				
	Diagrama Fluxo de Dados	<input type="radio"/>				
Orientação a Objetos	Caso de Uso Especificado	<input type="radio"/>				
	Diagrama de Classes	<input type="radio"/>				
	Diagrama de Atividades	<input type="radio"/>				
	Diagrama de Sequência	<input type="radio"/>				
Estruturada e OO	Diagrama de Estados	<input type="radio"/>				
	Protótipo Não Funcional	<input type="radio"/>				
	Modelo Lógico de Dados (MER)	<input type="radio"/>				
Estruturada e OO	Dicionário de Dados	<input type="radio"/>				
		<input type="radio"/>				
Projeto						
Estruturada	Modelo de Arquitetura	<input type="radio"/>				
	Diagrama Geral de Transações	<input type="radio"/>				

Figura 1. Identificação dos artefatos importantes

A segunda pesquisa (Souza *et al*, 2004) teve como objetivo verificar a real utilização de cada artefato de software no dia a dia da manutenção. Para isso foi elaborado um questionário (Figura 2) para o mantenedor indicar qual artefato ele utilizou para obter o entendimento do sistema no momento da manutenção. Foram apresentados os artefatos mais comuns utilizados nas abordagens da análise estruturada e da orientação a objetos. Eles foram agrupados por fase do ciclo de vida tradicional, sendo que para cada abordagem existia um formulário específico. A lista de artefatos da análise estruturada teve como base métodos utilizados em

algumas empresas nas quais os pesquisadores já trabalharam. Os artefatos da orientação a objeto tiveram como base a UML. O questionário foi distribuído para profissionais mantenedores de software em várias empresas que mantêm software.

A Tabela 1 mostra os resultados resumidos das duas pesquisas para o paradigma estruturado. É importante ressaltar que foi considerado o uso do artefato no momento de obter o conhecimento do sistema para realizar a manutenção, bem como se o artefato existe na documentação disponível de cada sistema mantido.

Questionário: Artefatos Usados na Manutenção (ES)

Este questionário tem o propósito de identificar quais tipos de artefatos de código e documentação foram utilizados para obter e entender o conhecimento de sistema, no fase de manutenção, do ponto de vista dos mantenedores de software. Ele deve ser respondido por profissionais que atuam na atividade de manutenção de sistemas.

Em caso de dúvida ou sugestão favor entrar em contato: caes.gilson@ufpb.edu.br

Caracterização do Mantenedor	
Nome (aproximado):	
Qual o seu cargo?	() CSM () SM
Como recebeu o trabalho de pesquisa?	() 1 a 3 () 3 a 5 () 5 a 10 () mais de 10 anos
Qual o tempo que atua em MANUTENÇÃO de sistemas (em anos)?	() 1 a 3 () 3 a 5 () 5 a 10 () mais de 10 meses
Qual o tempo de experiência (em anos) que você tem em manutenção de sistemas?	() Cometa - para corrigir erros não detectados no fase de teste () Adaptar - para acompanhar a evolução natural à computação () Prevenir - para garantir o funcionamento adequado do sistema () Promover - para melhorar a confiabilidade ou melhorar a vida útil.

Quantidade:

Até final de cada linha, marque com "X" quais artefatos de documentação você utiliza para obter e entender o conhecimento de sistema que você realiza manutenção.

IMPORTANTE: Marque somente os artefatos utilizados para obter o entendimento do sistema e NÃO os que você apenas tem que analisar.

Obs: Marque:

1 - Utiliza o conhecimento adquirido para o tipo de abordagem que o sistema foi desenvolvido (Estruturado, Orientação a Objeto);

2 - Os artefatos estão agrupados por fase de processo de desenvolvimento ou fase de manutenção.

Escreva aqui a sua análise, dúvida ou sugestão. (Opcional)

Escreva aqui a sua análise, dúvida ou sugestão. (Opcional)

Ferramentas para sistemas desenvolvidos pela abordagem ESTRUTURADA.

Data de início: ___/___/___ Data fim: ___/___/___

Indicações para preenchimento:

1) Marque com "X" quais artefatos de documentação você utiliza para obter e entender o conhecimento de sistema para realizar a manutenção realizada;

2) Caso o artefato não exista, marque somente a coluna "Não Existe".

IMPORTANTE: Marque somente os artefatos utilizados para obter e entender o sistema e NÃO os que você apenas tem que analisar.

Relação de artefatos por fase	Tipo de Artefato	Manutenção								
		1	2	3	4	5	6	7	8	
Leitura										
Lista de Requisitos										
Diagrama de Sistema										
Diagrama dos Requisitos										
Modelo Lógico de Dados										
Diagrama										
Planos de Teste dos Requisitos										
Diagrama Hierárquico de Módulos										
Diagrama Fluxo de Dados										
Modelo Físico de Dados										
Modelo Lógico de Dados (cont.)										
Diagrama de Testes										
Exatidão										
Modelo de Argumentos										
Diagrama Fluxo de Interpelações										
Especificações de Coligação										
Modelo Hierárquico de Dados										
Diagrama Hierárquico										
Condições										
Comentários de Código Fonte										
Código Fonte										
Verificação										
Planos de Teste de Módulos										
Planos de teste de dados										
Planos de teste Hierárquico										
Integração										
Planos de Integração de Dados										
Planos de Integração										
Planos de Integração										
Outros artefatos de documentação										

Figura 2. Identificação dos artefatos usados na manutenção.

Várias análises puderam ser feitas a partir dos resultados obtidos em cada pesquisa bem como na comparação entre ambas. Na pesquisa sobre a importância dos artefatos pôde-se observar o quanto a documentação é considerada importante para a manutenção na opinião dos profissionais da área. Na pesquisa sobre o uso efetivo dos artefatos pôde-se notar o quanto cada artefato é utilizado pelo mantenedor no momento da obtenção do entendimento do sistema e novamente constatar a importância da documentação.

Os artefatos mais utilizados foram "Código Fonte", "Comentários do Código Fonte" e Modelo Lógico de Dados, todos com 100% de utilização. Logo em seguida vem "Dicionário de Dados" e "Modelo Físico de Dados", ambos com 85,7%. Com um pouco menos de indicação seguem-se os planos de

teste, “Plano de Teste Unitário” (50%) e “Plano de Teste de Sistema” (42,9%).

Os artefatos considerados muito importantes na primeira pesquisa foram “Código Fonte” com 93,8% de indicação, “Comentários do Código Fonte” (77,3%), “Modelo Lógico de Dados” (71,9%), “Descrição dos Requisitos” (60,3%), “Modelo Físico de Dados” (59,4%), “Lista de Requisitos” (51,6%) e “Dicionário de Dados” (47,6%).

Tabela 1. Resultados das duas pesquisas

Abordagem Estruturada	Importância				Uso	
	Artefatos	Sem Importância	Pouco Importante	Importante	Muito Importante	Uso
Código Fonte	0,0%	0,0%	6,2%	93,8%	100,0%	100,0%
Comentários do Código Fonte	0,0%	7,6%	15,2%	77,3%	100,0%	100,0%
Modelo Lógico de Dados	0,0%	4,7%	23,4%	71,9%	100,0%	85,7%
Descrição dos Requisitos	4,8%	11,1%	23,8%	60,3%	40,0%	71,4%
Modelo Físico de Dados	0,0%	3,1%	37,5%	59,4%	85,7%	100,0%
Lista de Requisitos	9,7%	12,9%	25,8%	51,6%	40,0%	71,4%
Dicionário de Dados	1,6%	17,5%	33,3%	47,6%	85,7%	100,0%
Modelo Conceitual de Dados	6,5%	12,9%	35,5%	45,2%	40,0%	71,4%
Plano de Testes de Homologação	9,8%	14,8%	31,1%	44,3%	0,0%	42,9%
Manual do Usuário	9,5%	17,5%	34,9%	38,1%	16,7%	85,7%
Plano de Testes de Sistema	6,3%	17,5%	39,7%	36,5%	42,9%	100,0%
Plano de Implantação	7,8%	15,6%	42,2%	34,4%	33,3%	85,7%
Plano de Testes Unitário	8,1%	21,0%	37,1%	33,9%	50,0%	85,7%
Protótipo Funcional	11,5%	21,3%	36,1%	31,1%	20,0%	71,4%
Diagrama de	8,1%	16,1%	45,2%	30,6%	0,0%	42,9%

Abordagem Estruturada	Importância				Uso		
	Artefatos	Sem Importância	Pouco Importante	Importante	Muito Importante	Uso	Existência
Fluxo de Dados							
Modelo de Arquitetura	8,8%	26,3%	35,1%	29,8%	0,0%	71,4%	
Especificação de Componentes	6,8%	16,9%	47,5%	28,8%	0,0%	57,1%	
Plano de Migração de Dados	11,7%	18,3%	41,7%	28,3%	0,0%	57,1%	
Diagrama Hierárquico de Funções	8,6%	22,4%	46,6%	22,4%	0,0%	42,9%	
Diagrama de Contexto	6,7%	35,0%	36,7%	21,7%	0,0%	57,1%	
Funções Derivadas dos Requisitos	8,5%	31,9%	38,3%	21,3%	0,0%	57,1%	
Protótipo não Funcional	13,8%	22,4%	43,1%	20,7%	0,0%	57,1%	
Glossário	7,9%	33,3%	38,1%	20,6%	20,0%	71,4%	
Diagrama Geral de Transações	10,2%	26,5%	42,9%	20,4%	0,0%	57,1%	

4. Discussão e Conclusão

Os resultados da pesquisa constataram a importância da documentação na manutenção de software, pois os artefatos com menor índice de importância tiveram mais de 50% de indicação dos mantenedores. Quanto ao grau de utilização, dos vinte e quatro artefatos indicados, quatorze foram utilizados pelos mantenedores. Este fato comprova o risco que representa a falta de documentação para a manutenção de software. Por outro lado, nem todos os documentos foram utilizados pelos mantenedores para buscar o entendimento do sistema. Isto reforça também a necessidade de se ter um propósito para o qual a documentação é criada e mantida.

Em geral, pôde-se constatar que os artefatos indicados como mais importantes pelos profissionais são também os mais utilizados na prática

pelos mantenedores. O artefato mais importante sem nenhuma dúvida é o código fonte, que juntamente com os comentários nele existentes reforçam a importância da adoção de padrões de codificação e de organização interna do código. Ambos foram usados por todos os entrevistados. Outros artefatos que foram bastante utilizados foram o modelo de dados (físico e lógico) e o dicionário de dados, o que comprova a importância atribuída a eles e ressalta a necessidade desses modelos estarem atualizados e disponíveis no momento da manutenção. Foi dada uma importância muito grande a descrição dos requisitos na primeira pesquisa, no entanto sua utilização prática ficou um pouco abaixo média (40%). Por outro lado, os planos de testes unitário e de sistema tiveram uma utilização prática razoável (50% e 42,9% respectivamente) enquanto não eram considerados muito importantes pela maioria dos profissionais na primeira pesquisa. Pôde-se notar também que a maioria dos artefatos considerados menos importantes não foi utilizada pelos mantenedores.

Os dados desta pesquisa podem ser úteis para várias análises em torno dos artefatos de documentação de software no sentido de buscar o equilíbrio na produção de artefatos. O trabalho completo desta pesquisa pode ser encontrado em (Souza, 2005).

5. Referências Bibliográficas

1. AMBLER, Scott W. Agile Documentation. 2001-2004, The Official Agile Modeling (AM) Site, 2001, Disponível em : <<http://www.agilemodeling.com/essays/agileDocumentation.htm>>, Acesso em: 02 abr. 2001.
2. BRIAND, Lionel C.. Software Documentation: How Much is Enough?, Proceedings of the 7th European Conference on Software Maintenance and Reengineering (CSMR'03), IEEE, 2003, p.13.
3. FORWARD, Andrew. Software Documentation – Building and Maintaining Artefacts of Communication, 2002. Teste de Mestrado, Universidade de Ottawa, Ottawa, Toronto, Canadá.
4. RAJLICH, Václav. Incremental Redocumentation Using the Web, IEEE Software, Setembro/Outubro 2000, p.102-106.
5. SOUZA, Sérgio C. B. de; NEVES, Wesley C. G. das; ANQUETIL, Nicolas; OLIVEIRA, Káthia M. de. Investigação da Documentação de Maior Importância para Manutenção de Software, 4ª Jornadas Iberoamericanas de Ingeniería del Software e Ingeniería del Conocimiento (JIISIC'04), Outubro, 2004.
6. SOUZA, Sérgio C. B. de; NEVES, Wesley C. G. das; ANQUETIL, Nicolas; OLIVEIRA, Káthia M. de. Documentação Essencial para Manutenção de Software II, Workshop de Manutenção de Software

Moderna, Simpósio Brasileiro de Engenharia de Software, Outubro, 2004.

7. SOUZA, Sérgio, Documentação Essencial para Manutenção de Software, Tese de Mestrado, Universidade Católica de Brasília (a ser publicada em abril de 2005)
8. THE STANDISH GROUP. The Standish Group Report - CHAOS. The Standish Group, 1995, Disponível em: <<http://www.scs.carleton.ca/~beau/PM/Standish-Report.html>>, Acesso em: 07 abr. 2004.
9. TILLEY, Scott R.; MÜLLER, Hausi A.. INFO: A Simple Document Annotation Facility, Proceedings of the 9th annual international conference on Systems documentation (SIGDOC'91), ACM Press, 1991, p.30-36.
10. TILLEY, Scott R; HUANG, Shihong. Towards a Documentation Maturity Model Proceedings of the 21st annual international conference on Documentation (SIGDOC'03), San Francisco, Califórnia, EUA, Outubro, 2003, p.93-99.
11. WONG Kenny, TILLEY, Scott R.; MÜLLER, Hausi A.; STOREY Margaret-Anne D.. Structural Redocumentation: A Case Study, IEEE Software, Janeiro, 1995, p.46-54.