

Joenio Marques da Costa

***Extração de Informações de Dependência entre
Módulos de Programas C/C++***

Salvador-Bahia

2009

Joenio Marques da Costa

***Extração de Informações de Dependência entre
Módulos de Programas C/C++***

Ante-projeto apresentado ao Curso de graduação em Bacharelado em Informática, Universidade Católica do Salvador, como requisito parcial para obtenção do grau de Bacharel em Informática.

Orientador:

Prof. M.Sc. Antonio A. S. Terceiro

Co-orientador:

Prof(a). M.Sc. Maria Mesquita Mota

UCSAL - UNIVERSIDADE CATÓLICA DO SALVADOR
FACULDADE DE INFORMÁTICA

Salvador-Bahia

2009

SUMÁRIO

1	Justificativa	3
2	Objetivo	4
3	Metodologia	5
3.1	Etapas	5
3.2	Cronograma	5
	Referências Bibliográficas	6

1 JUSTIFICATIVA

A arquitetura de software define o relacionamento entre os grandes componentes estruturais de um projeto (PRESSMAN, 2005). A documentação desta arquitetura é útil para possibilitar, por exemplo, a extração de métricas como coesão e acoplamento ou para estudar o impacto de possíveis alterações (rastreadabilidade) de um projeto (TERCEIRO, 2008).

Uma ferramenta de extração de informações de dependências entre módulos é útil para gerar esta documentação a partir do código fonte do próprio projeto. Existem ferramentas capazes de realizar esta tarefa, mas entre as opções pesquisadas (HASSAN; JIANG; HOLT, 2005) a maioria delas faz uso do código objeto como dados de entrada da análise, ou seja, faz uso de dados resultante da compilação do código fonte.

Um problema em extrair estas informações do código objeto é que algumas informações conhecidas pelos desenvolvedores do projeto não estão presentes, como por exemplo macros em projetos C/C++ que são substituídas pelo pré-processador (HASSAN; JIANG; HOLT, 2005) durante a compilação.

Este problema pode ser solucionado analisando diretamente o código fonte do projeto. Esta análise traria uma informação mais próxima da arquitetura do projeto conhecida pelos desenvolvedores e adicionalmente possibilitaria analisar projetos que por algum motivo não compilem mais, seja por dependências antigas ou por pequenas falhas no código fonte.

2 OBJETIVO

Construir uma ferramenta para análise de dependência entre módulos de programas escritos em C/C++ utilizando como base apenas o código fonte (i.e. não dependa de compilar o código).

A construção desta ferramenta terá como base o software Egypt¹, um software desenvolvido por Andreas Gustafsson para análise de dependência de módulos em programas C/C++ que utiliza como base código objeto gerado pelo GCC. O Egypt é software livre e em Janeiro de 2009 foi completamente reestruturado por Antonio Terceiro o qual tem mantido em². Esta versão modificada será o alvo deste trabalho.

Para fazer a análise do código fonte pretende-se utilizar uma ferramenta como antlr³ ou Doxygen⁴ que ofereça a capacidade de analisar projetos C/C++ sem necessidade de compilar o código fonte.

Por fim será feito um estudo comparativo entre a ferramenta desenvolvida por este trabalho com o extrator atual do Egypt (baseado no GCC), em termos de informações que um consegue extrair e o outro não.

¹<http://www.gson.org/egypt>

²<http://github.com/terceiro/egypt>

³<http://www.antlr.org>

⁴<http://www.doxygen.org>

3 METODOLOGIA

Primeiramente será feito um estudo sobre Arquitetura e Engenharia de Software, Engenharia de Software Experimental e as ferramentas antlr, Doxygen e Egypt, a partir deste estudo será implementado no Egypt um módulo, plugin ou extensão que permita analisar e extrair informações de dependência entre módulos de softwares escritos em C/C++.

3.1 ETAPAS

1. Estudar Egypt, antlr e Doxygen e escolher entre antlr ou Doxygen.
2. Implementar projeto e escrever monografia.
3. Fazer testes de extração de informação de dependências em projetos de software livre.
4. Avaliar testes da etapa anterior e documentar resultados.
5. Revisar monografia com orientador.

3.2 CRONOGRAMA

Tabela 3.1: Etapas do projeto

Etapa	Fev	Mar	Abr	Mai	Jun
1	xxxx	x			
2		xxxx	xxxx	xxxx	x
3			xxx	xxxx	
4			x	xxxx	x
5				xxx	xxxx

REFERÊNCIAS BIBLIOGRÁFICAS

HASSAN, A. E.; JIANG, Z. M.; HOLT, R. C. Source versus object code extraction for recovering software architecture. In: *Proceedings of the 12th Working Conference on Reverse Engineering (WCRE'05)*. [S.l.: s.n.], 2005.

PRESSMAN, R. S. *Engenharia de Software*. sexta. [S.l.]: Makron Books, 2005.

TERCEIRO, A. A. S. *Verificando as Leis de Crescimento Contínuo e Complexidade Crescente em um projeto de software livre de pequeno porte*. Salvador-BA-Brasil, 2008. [Online; acessado 15 de Janeiro de 2009]. Disponível em: <<http://disciplinas.dcc.ufba.br/pub/MATA26/TrabalhoTerceiro/mata26-terceiro-projeto-piloto.pdf>>.