

# An Empirical Study of Static Call Graph Extractors [1]

MAT08 - Tópicos em Engenharia de Software 1 (2012.2)

Joenio Marques da Costa

17 de janeiro de 2013

## Resenha

Informalmente, um grafo de chamada representa chamadas entre entidades em um dado programa. Além dos compiladores, muitas ferramentas de engenharia de software extraem grafos de chamada com a expectativa de ajudar no entendimento do programa. Estas ferramentas no entanto dão resultados distintos, dos quais podem ou antender a certo cenário ao engenheiro d software. Foi feito um estudo empírico com 3 programas C, mapmaker, mosaic e gcc utilizando 5 ferramentas de extração, cflow, CIA, Field, mkfunctmap, and rigiparse. Uma análise quantitativa dos grafos de chamada gerados por cada ferramentas mostram considerável variação. Uma análise qualitativa revelam os motivos: tratamento diferenciado para macros, ponteiros de função, formatos de entrada.

Compiladores geralmente calculam grafos de chamada para fins de otimização, grafo de chamada pode representar uma séria de relações binárias entre entidades de um programa, por exemplo, procedimentos, módulos, arquivos, etc. Enquanto grafo de chamada construído por compiladores são bem conservadores, as ferramentas utilizadas pelos engenheiros de software são mais flexíveis, admitindo certos falsos negativos, visto que o grafo será consumido por humanos a fim de entendimento do software.

Este trabalho foca em ferramentas de análise estática, ao invés de utilizar dados de run-time do programa.

Os experimentos mostraram que todas as ferramentas geram tanto falsos positivos quanto falsos negativos, e é impossível determinar exatamente o número de falsos emitidos por cada ferramenta uma vez que o grafo de chamada verdadeiro é indefinido.

Apesar dos resultados empiricos baseados em poucas ferramentas de extração de design eles jogam luz a algumas questões que qualquer desenvolvedor de uma ferramenta de extração deve levar em conta (3 Design Decision).

Uma decisão chave que o desenvolvedor deve tomar é sobre a forma de entrada, ela pode ser uma lista de arquivos ou diretórios de código fonte, ou mesmo um arquivo executável (binário).

Uma segunda decisão importante é sobre como lidar com MACROS. Uma terceira decisão importante é sobre a linguagem de programação, a ferramenta suporta quais variantes de C? Ou suporta quais outras linguagens de programação além de C? E ainda, a ferramenta suporta análise (parser) do código fonte legado que não compila mais? Ou somente códigos compiláveis?

Outros aspectos a respeito do comportamento da ferramenta também são importantes, como o algoritmo utilizado para inspecionar o código-fonte. Outro aspecto é sobre se o grafo de chamadas é local ou global.

## Referências

- [1] G. C. M. D. N. E. S.-C. Lan, “An empirical study of static call graph extractors,” p. 10, 2009.