

Projecto final de curso

Simão Melo de Sousa

Título

Geração de wrappers para C usando Ocaml

Objectivo

O objectivo deste projecto de final de curso é o desenvolvimento de um módulo em Ocaml que permita a geração automática de wrappers entre C e Ocaml. O objectivo de um wrapper é chamar uma segunda função com o mínimo de recursos computacionais.

Actualmente não existe nenhuma ferramenta/módulo que automaticamente extraia a assinatura de uma função em C e construa o wrapper que permite invocar esta mesma função a partir do Ocaml. O exemplo seguinte mostra o que se pretende com o wrapper:

Função original em C

```
int fun(int c,int a){ return(a+c); }
```

Para a seguinte função é necessário:

1. Criação do programa Ocaml que permita chamar esta função:

```
external wrap_fun: int -> int -> int = "fun";;
```

2. Criação do wrapper em C

```
#include <stdio.h>
#include <caml/mlvalues.h>
CAMLprim value
wrap_fun ( value v1, value v2 ) {
    Val_int(fun(Int_val(v1),Int_val(v2)));
}
```

Este projecto insere-se no projecto de investigação e inovação PROVA em colaboração com a Educued, empresa sediada na Silicon Valley - USA.

Assim, este trabalho de final de curso será parte integrante de uma ferramenta de geração de testes que está a ser actualmente desenvolvida pela empresa Educued em colaboração com o grupo Release. Pretende-se igualmente possibilitar o desenvolvimento futuro deste trabalho no contexto de uma tese de mestrado em parceria próxima com a Educued.

Plano de Trabalho

O plano de trabalhos pode ser dividido nas seguintes tarefas:

- Introdução aos conceitos próprios do projecto, em particular as metodologias de bindings/wrappers de bibliotecas computacionais;

- Estudo do CIL para parsing de programas na linguagem C (<http://www.cs.berkeley.edu/~necula/cil/>);
- Estudo das regras de wrapping entre C e Ocaml (<http://www.linux-nantes.org/~fmonnier/ocaml/ocaml-wrapping-c.php>);
- Implementação do módulo:
- Parsing de C para Ocaml usando o CIL;
- Representação dos tipos de dados C em Ocaml;
- Implementação das regras de tradução entre C e Ocaml para tipos simples;
- Implementação das regras de tradução entre C e Ocaml para tipos compostos (estruturas, listas, etc.);
- Testes e validação do protótipo;
- Elaboração do relatório.

Contactos

Simão Melo de Sousa