

発表概要

CにLuaを容易に埋め込むための言語LuCa

谷村 明^{1,a)} 岩崎 英哉^{1,b)}

2014年11月11日発表

既存の多くのスクリプト言語は、Cプログラムとの連携を意図したAPIを提供している。特にスクリプト言語 Lua は、Cプログラムに Lua インタプリタが組み込まれることを前提にして設計されており、Cで実装された多くの既存のソフトウェアも Lua を組み込んでいる。しかし、Cプログラムと Lua を連携する際、プログラマは C と Lua の間のデータのやりとりで煩雑な C API を利用しなければならない。このことは、プログラマにとって実装コストの増加の原因となり、さらに、C API の規約に従う必要がある結果、ソースコードの見通しも悪くなる。この問題を解決するため、我々は C 言語を拡張し、Lua コードを C コード中に直接埋め込んで記述することのできる言語 LuCa を提案し、その処理系を実装する。LuCa では、埋め込まれた Lua コードから C コードの変数や関数にアクセスすることができる。そのため、C API を陽に使うことなく Lua コードらしい自然な記述を行うことが可能になり、プログラマの負担を軽減することができる。LuCa 処理系は与えられた LuCa コードを C コードへと変換する。Lua との連携部分は C API を適切に利用したコードとするため、LuCa 処理系は特殊な C コンパイラや特殊な Lua 処理系を用意する必要はない。本発表では、LuCa 言語の設計とその処理系の実装について述べる。

LuCa: A Language for embedding Lua in C

AKIRA TANIMURA^{1,a)} HIDEYA IWASAKI^{1,b)}

Presented: November 11, 2014

Most scripting languages provide C APIs that make these languages cooperate with programs written in C. Especially, scripting language Lua is designed on the assumption that its interpreter is embedded in a C program. In fact, a lot of existing software written in C embeds the Lua interpreter. However, when embedding a Lua program in a C program, it is necessary for the programmer to use complicated C APIs provided by Lua for transferring data between C and Lua. This increases the developing cost of programs within which C and Lua cooperate. Worse, obeying protocols of Lua's C APIs spoils the readability of source programs. To resolve this problem, we propose a new language LuCa that extends C to make it possible to embed Lua code directly, and implement a LuCa processor. In a LuCa program, embedded Lua code can access variables and functions in C. Thus, using LuCa enables the programmer to write Lua-like code without explicit C APIs and reduces the programmer's burden. The LuCa processor converts a given LuCa code into a C code. Since the cooperating part of C and Lua is converted into C code with existing C APIs, the LuCa processor does not require special C compiler and Lua interpreter. We describe the LuCa language and the implementation of its processor.

¹ 電気通信大学大学院情報理工学研究科
Graduate School of Informatics and Engineering, The University of Electro-Communications, Chofu, Tokyo 182-8585, Japan

a) tanimura@ipl.cs.uec.ac.jp

b) iwasaki@cs.uec.ac.jp