



# About Lua

Roberto Ierusalimschy

PONTIFÍCIA UNIVERSIDADE CATÓLICA  
DO RIO DE JANEIRO



# If programming languages were cars



Lua: cute, efficient, and becoming very trendy

# If programming languages were religions

Lua would be Wicca - A pantheistic language that can easily be adapted for different cultures and locations. Its code is very liberal, and allows for the use of techniques that might be described as magical by those used to more traditional languages. It has a strong connection to the moon.

# What is Lua

- Yet another scripting language
  - not totally unlike Perl, Python, Javascript
- Used in production since 1993
  - widely used in some areas, virtually unknown in others
- Goals: simple, small, portable
- Goal: embedability
  - real emphasis on *scripting*

# Where is Lua

## embedded devices

TVs (Samsung), routers (Cisco), keyboards (Logitech), printers (Olivetti), set-top boxes (Verizon), M2M devices (Sierra Wireless), calculators (TI-Nspire), etc.

# Where is Lua

scripting applications

Wireshark, Snort, Nmap, VLC Media Player,  
LuaTeX, ...

[http://en.wikipedia.org/wiki/  
Category:Lua-scriptable\\_software](http://en.wikipedia.org/wiki/Category:Lua-scriptable_software)

Slashdot: News for nerds, Feb 1, 2012:

“Wikipedia Chooses Lua As Its New template language”

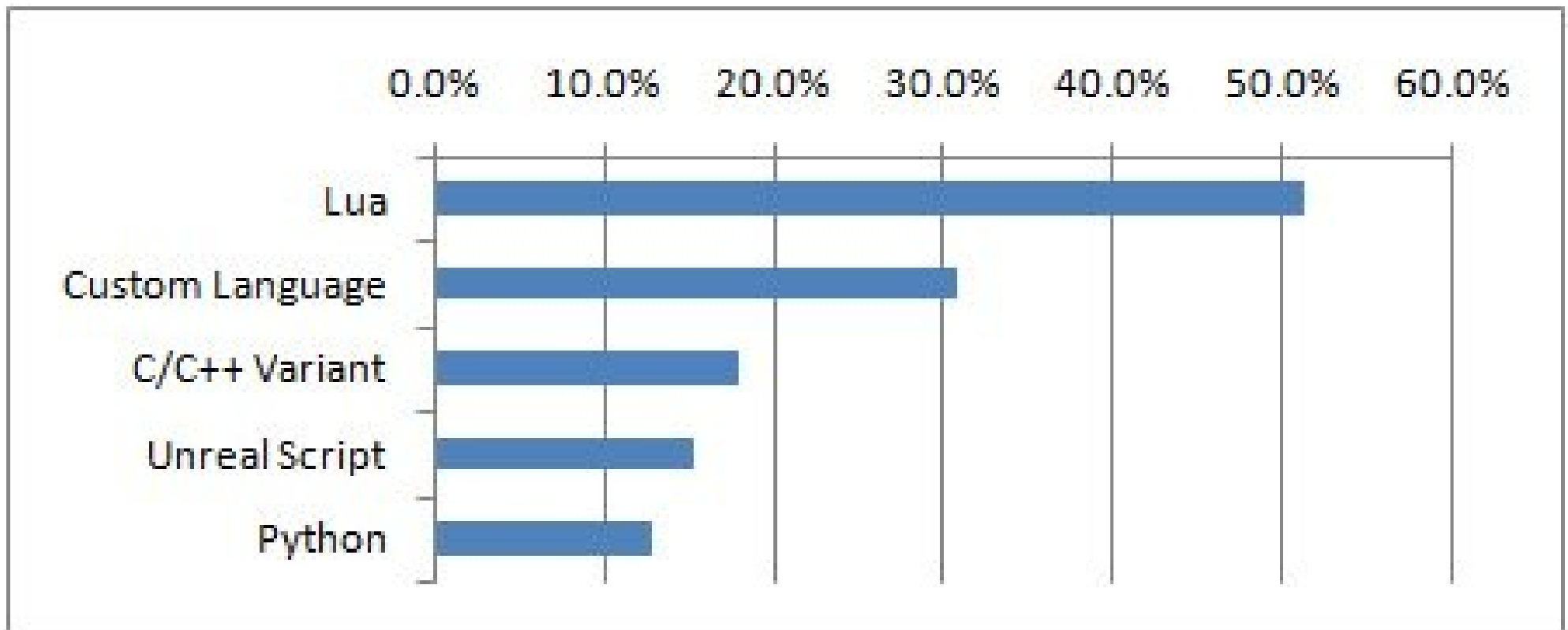
Adobe Lightroom  
one million lines of Lua code





# Lua in games

The Engine Survey (03/02/09, Gamasutra):  
What script languages are most people using?



# Apple Insider, June 11, 2010

## “Apple relaxes iOS SDK terms to allow Lua but block Flash

“Apple's iOS SDK rules for iPhone developers have relaxed the restriction of section 3.3.2 pertaining to interpreted code, enabling Apple to forbid Flash and other middleware platforms while still enabling popular game engines and libraries.”



# Lua main goals

- Portability
- Small size
- Embedability
- Simplicity

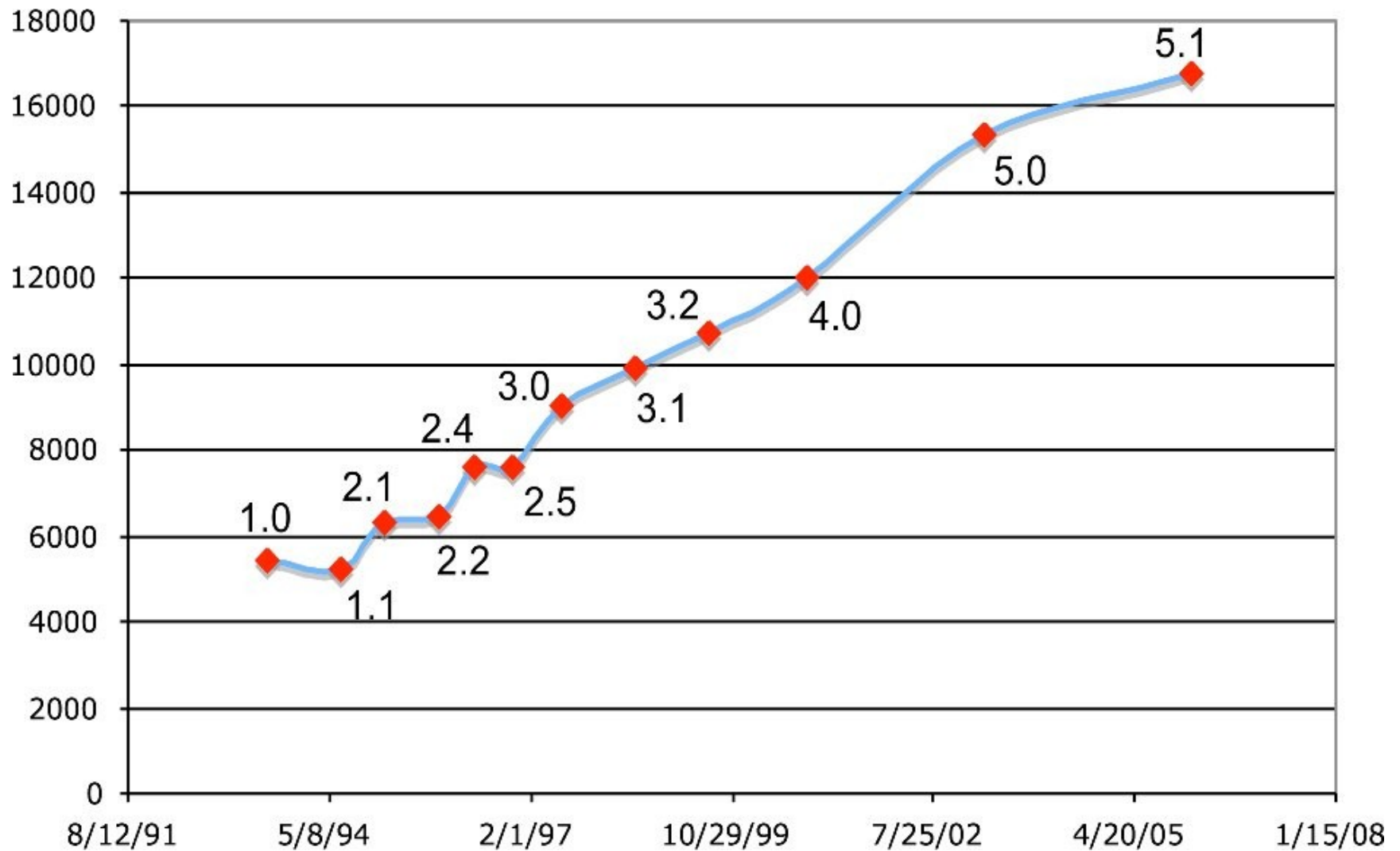
# Portability

- Written in  $\text{ANSI C} \cap \text{ANSI C++}$ 
  - avoids `#ifdef`'s
  - avoids dark corners of the C standard
- Runs on most platforms we ever heard of
  - WinCE, Symbian, iOS, BREW, Android, PS3, PSP, Nintendo DS, IBM z/OS, etc.
- Runs inside OS kernels
  - FreeBSD, Linux

# Small size

- Less than 20,000 lines of C code
  - 7,800 semicolons
- ELF binary: less than 200 KB
  - complete package
- Important for portability
  - allows Lua to run in small machines

# Small size



# Embedability

- Emphasis on scripting
  - to be used together with a *system language*
  - *tight integration between languages*
  - *not only external libraries*
- Provided as a library
- Not only an implementation issue
- Embedded in C/C++, Java, Fortran, C#, Perl, Ruby, Python, etc.



# Scripting in Grim Fandango

“[The engine] doesn't know anything about adventure games, or talking, or puzzles, or anything else that makes Grim Fandango the game it is. It just knows how to render a set from data that it's loaded and draw characters in that set. [...]

The real heroes in the development of Grim Fandango were the scripters. They wrote everything from how to respond to the controls to dialogs to camera scripts to door scripts to the in-game menus and options screens. [...]

A TREMENDOUS amount of this game is written in Lua. The engine, including the Lua interpreter, is really just a small part of the finished product.”

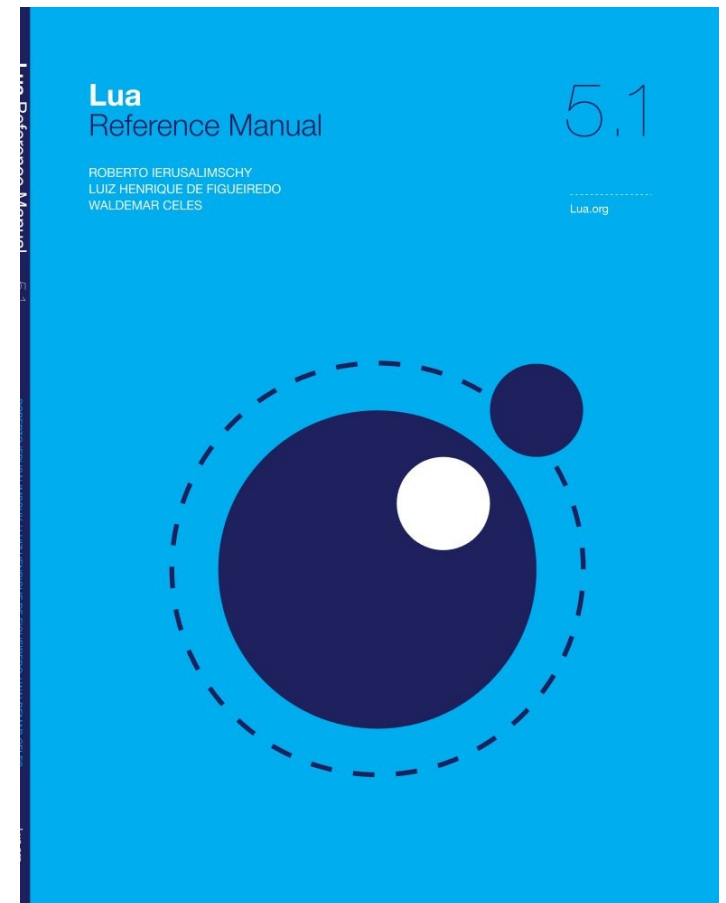
Bret Mogilefsky

# Simplicity

Reference manual with 100 pages (proxy for complexity)

documents language,  
libraries, and C API

(spine)



# Simplicity

- Only one numerical type
  - double
- Only one data structure mechanism
  - tables (associative arrays)
- Few but powerful mechanisms
  - closures, tables, (full) coroutines
- Important for small size

# An Overview of Lua

- Conventional syntax
  - somewhat verbose (end-user programming)

```
function fact (n)
  if n == 0 then
    return 1
  else
    return n * fact(n - 1)
  end
end
```

```
function fact (n)
  local f = 1
  for i = 2, n do
    f = f * i
  end
  return f
end
```

# An Overview of Lua

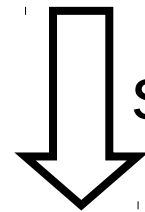
- Semantically somewhat similar to Scheme
- Also many similarities with JavaScript
  - Lua predates JavaScript by two years

# An Overview of Lua

- Dynamically typed
- Garbage collected
  - incremental garbage collector
- Functions are first-class values with static scoping
  - closures

# BTW...

```
function fact (n)
  local f = 1
  for i=2,n do f = f * i; end
  return f
end
```



syntactic sugar

```
fact = function (n)
  local f = 1
  for i=2,n do f = f * i; end
  return f
end
```

# Tables

- Associative arrays
  - any value as key
- Only data-structure mechanism in Lua



# Data structures

- Tables implement most data structures in a simple and efficient way
- Records: syntactical sugar `t.x` for `t["x"]`:

```
t = {}  
t.x = 10  
t.y = 20  
print(t.x, t.y)  
print(t["x"], t["y"])
```

# Data Structures

- Arrays: integers as indices

```
a = {}  
for i=1,n do a[i] = 0 end
```

- Sets: elements as indices

```
t = {}  
t[x] = true      -- t = t ∪ {x}  
if t[x] then     -- x ∈ t?  
    ...
```

# Modules

- Tables populated with functions

```
local math = require "math"  
print(math.sqrt(10))
```

- Several facilities come for free
  - submodules
  - local names

```
local m = require "math"  
print(m.sqrt(20))  
local f = m.sqrt  
print(f(10))
```

# Constructors

Flexible expressions to create and initialize a table

```
p = {x = 10.5, y = -7.6}  
days = {"Sun", "Mon", "Tue", ...}
```

```
Point{x = 1, y = 0}  
– sugar for  
Point({x = 1, y = 0})
```

# Data Description

- Constructors are a good format for data description
- Not totally unlike JSON
  - in use since 1994

<POEM>

<TITLE>Lines Written in Early Spring</TITLE>

<AUTHOR>

<FIRSTNAME>William</FIRSTNAME>

<LASTNAME>Wordsworth</LASTNAME>

</AUTHOR>

<STANZA>

<LINE N="1">I heard a thousand blended notes,</LINE>

<LINE N="2">While in grove I sate reclined,</LINE>

<LINE N="3">In that sweet mood when pleasant thoughts</LINE>

<LINE N="4">Bring sad thoughts to the mind.</LINE>

</STANZA>

<STANZA>

<LINE N="5">To her fair works did nature link</LINE>

<LINE N="6">The human soul that through me ran;</LINE>

<LINE N="7">And much it griev'd me my heart to think</LINE>

<LINE N="8">What man has made of man.</LINE>

</STANZA>

</POEM>

```
POEM{
  TITLE="Lines Written in Early Spring",
  AUTHOR={firstname="William", lastname="Wordsworth"};
  STANZA{
    LINE{N=1;"I heard a thousand blended notes,"},
    LINE{N=2;"While in grove I sate reclined,"},
    LINE{N=3;"In that sweet mood when pleasant thoughts"},
    LINE{N=4;"Bring sad thoughts to the mind."},
  },
  STANZA{
    LINE{N=5;"To her fair works did nature link"},
    LINE{N=6;"The human soul that through me ran;"},
    LINE{N=7;"And much it griev'd me my heart to think"},
    LINE{N=8;"What man has made of man."},
  }
}
```

# Coroutines

- Lua implements asymmetric, first-class, “stackfull” coroutines
- We can implement `call/cc1` on top of them
- We can implement cooperative (non-preemptive) multithreading on top of them
- Useful for *who-has-the-main-loop* problem
  - embedding



# The Beginning of Lua

- 1993: Tecgraf - a partnership between PUC-Rio and Petrobras
- Two programs with configuration problems
  - each with its own mini-language
  - both languages with severe limitations
- idea: a generic configuration language
  - Lua

# International Exposure

Newsgroups:

comp.compilers,comp.lang.misc,comp.programming,comp.lang.c

From: lhf@csg.uwaterloo.ca (Luiz H de Figueiredo)

Organization: Computer Systems Group, University of Waterloo

Keywords: tools, available

Date: Fri, 8 Jul 1994 11:51:45 GMT

This is the first public release of Lua.

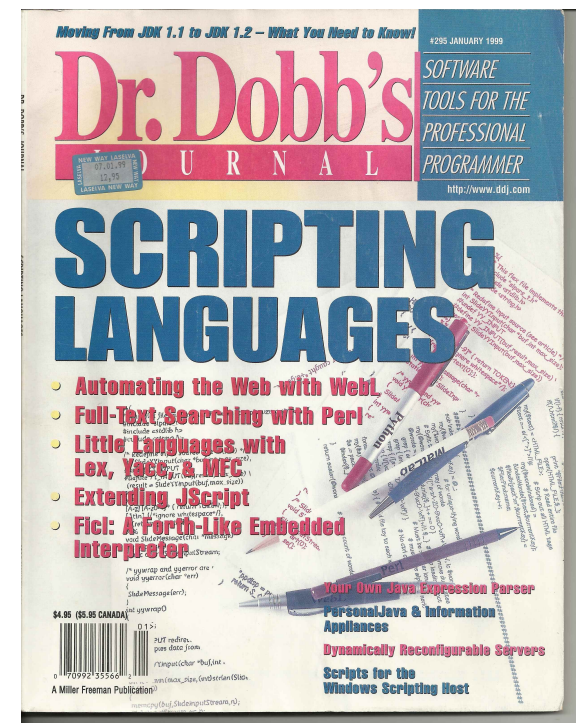
\* What is Lua?

Lua is a simple, yet powerful, language for extending applications. Lua has been developed by TeCGraf, the Computer Graphics Technology Group of PUC-Rio, the Catholic University of Rio de Janeiro, Brazil.

Dozens of industrial products developed by TeCGraf use Lua.

[...]

L. H. de Figueiredo, R. Ierusalimschy, W. Celes.  
Lua: an extensible embedded language. *Dr.  
Dobb's Journal*, 21(12):26-33, 1996.



# Lua in Games (the beginning)

From: Bret Mogilefsky <mogul@lucasarts.com>

To: "'lua@icad.puc-rio.br'" <lua@icad.puc-rio.br>

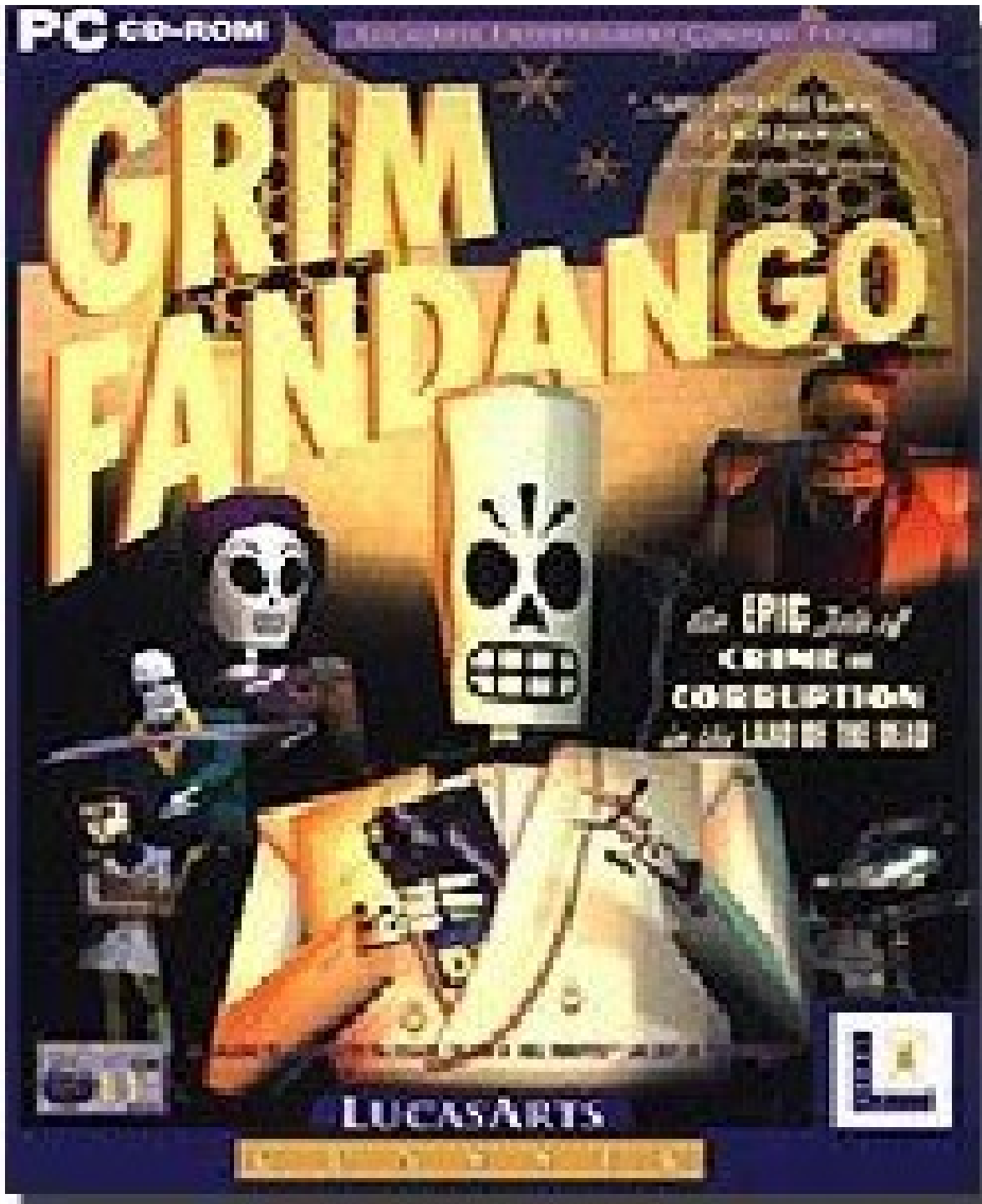
Subject: LUA rocks! Question, too.

Date: Thu, 9 Jan 1997 13:21:41 -0800

Hi there...

After reading the Dr. Dobbs article on Lua I was very eager to check it out, and so far it has exceeded my expectations in every way! It's elegance and simplicity astound me. Congratulations on developing such a well-thought out language.

Some background: I am working on an adventure game for the LucasArts Entertainment Co., and I want to try replacing our older adventure game scripting language, SCUMM, with Lua.



Lucas Arts,  
1998: First AAA  
game to use Lua



Enter door to LUA Bar

© 2001 LucasArts Entertainment Company, LLC and its licensors. All rights reserved. 08.07.01

CLOSE WINDOW

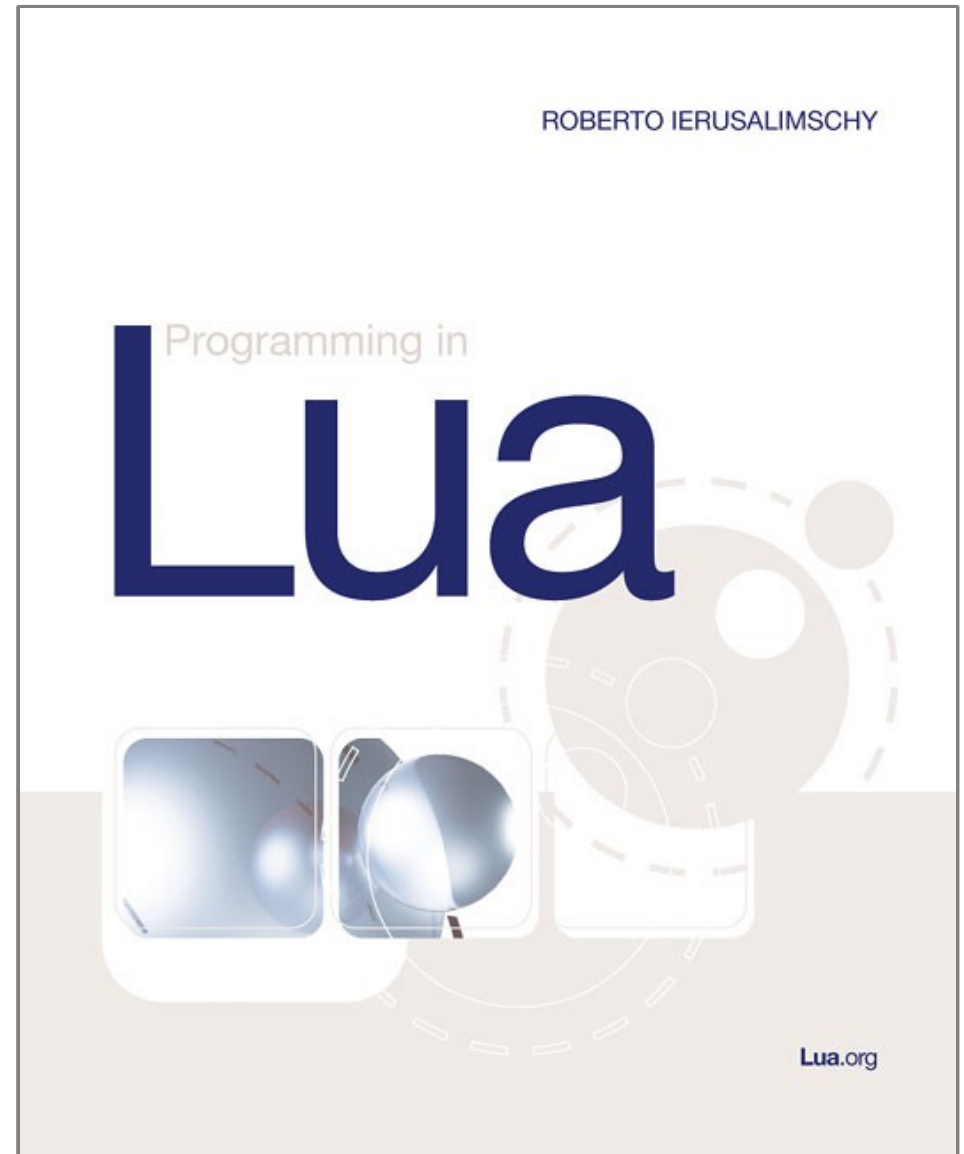


PlayStation.2

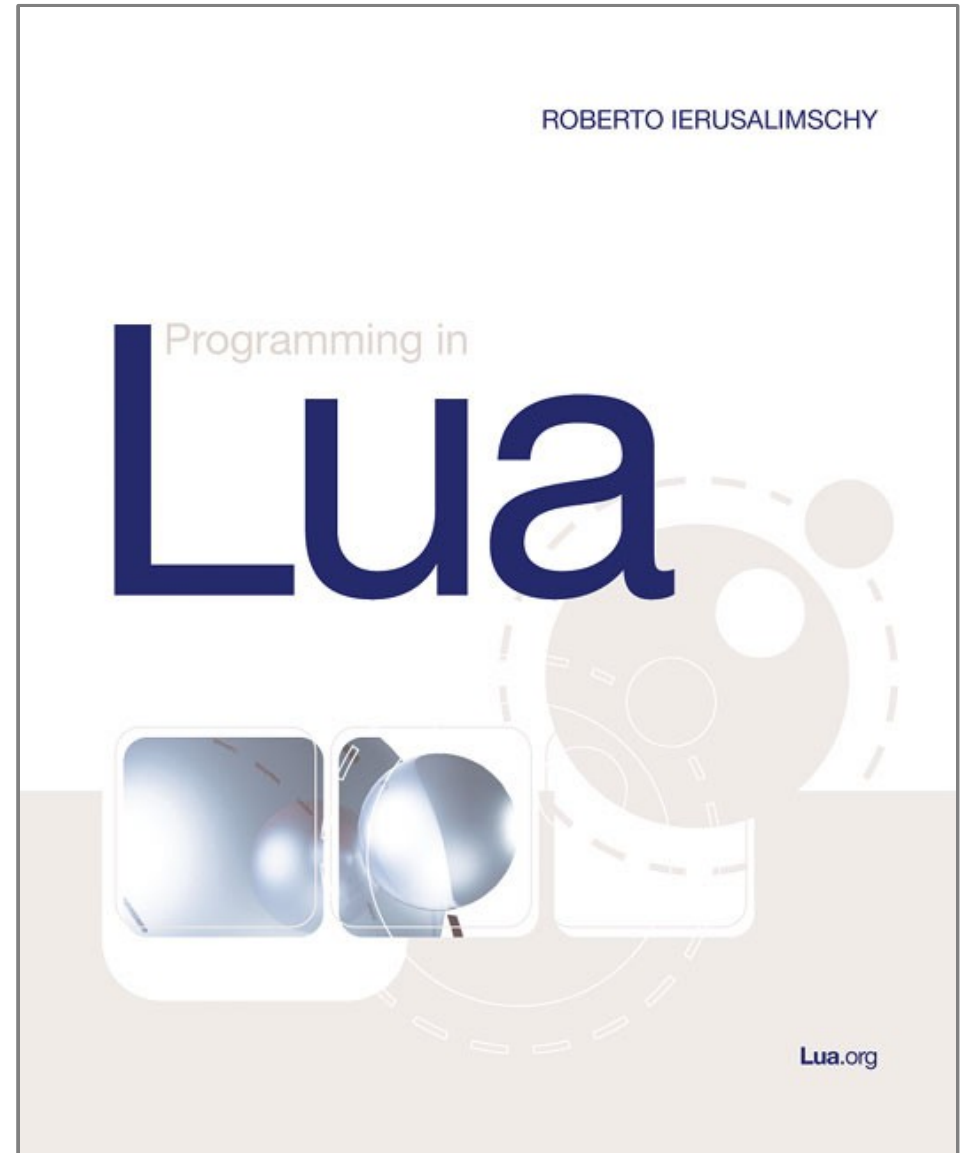
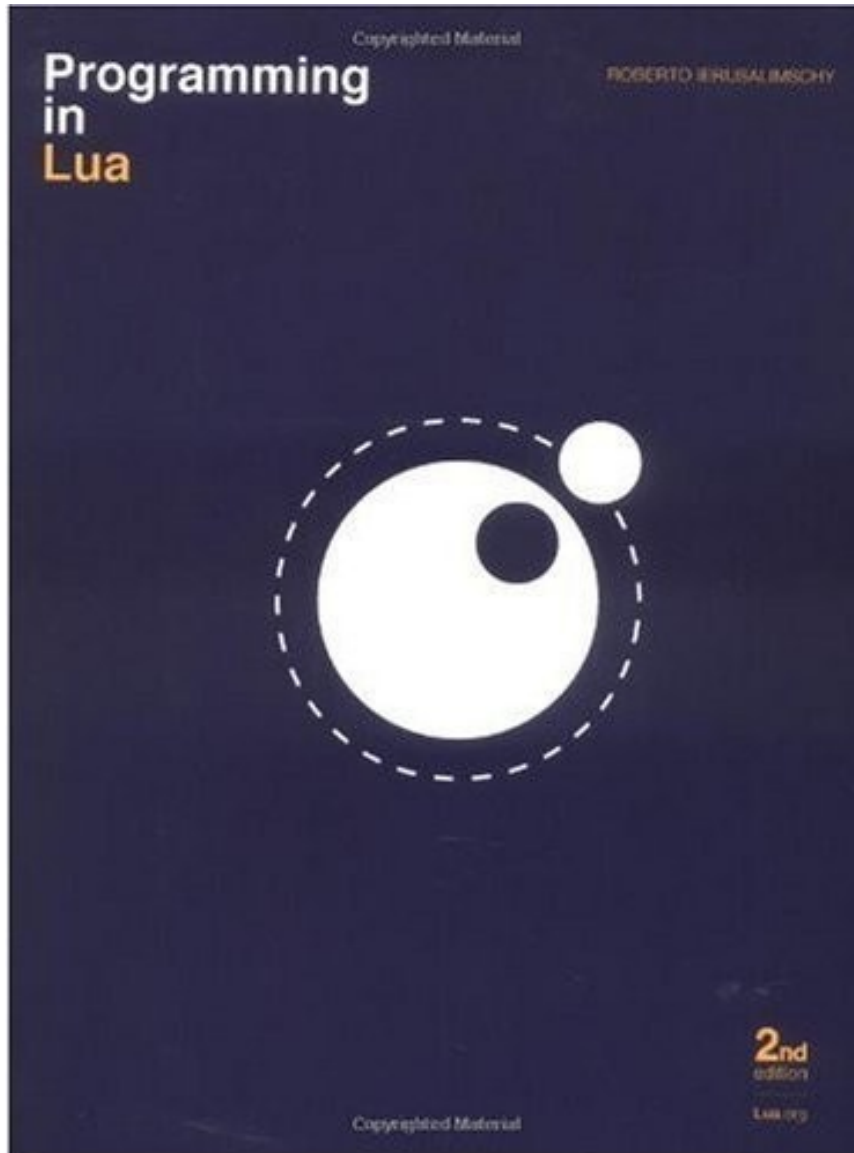
13 of 24

# Books

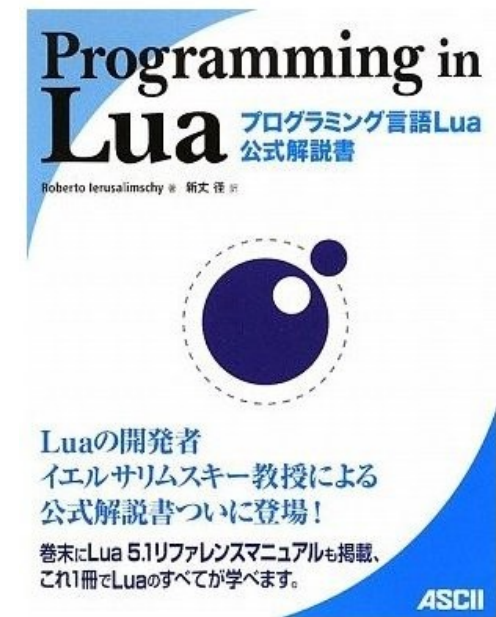
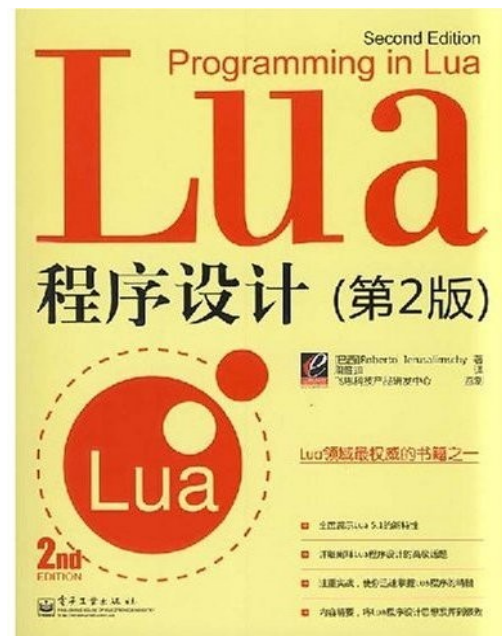
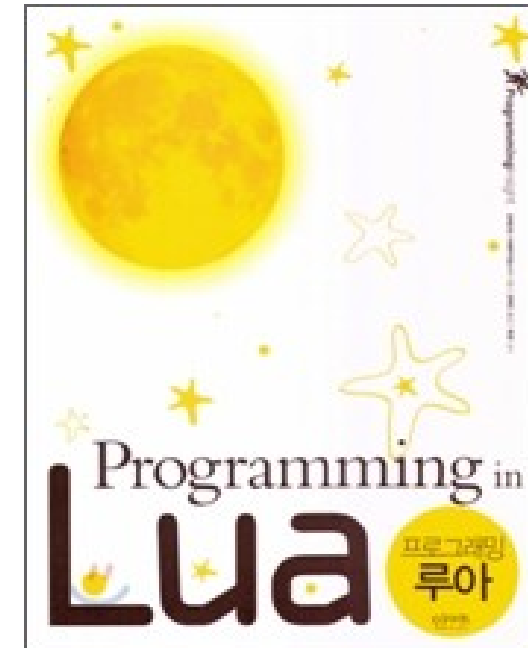
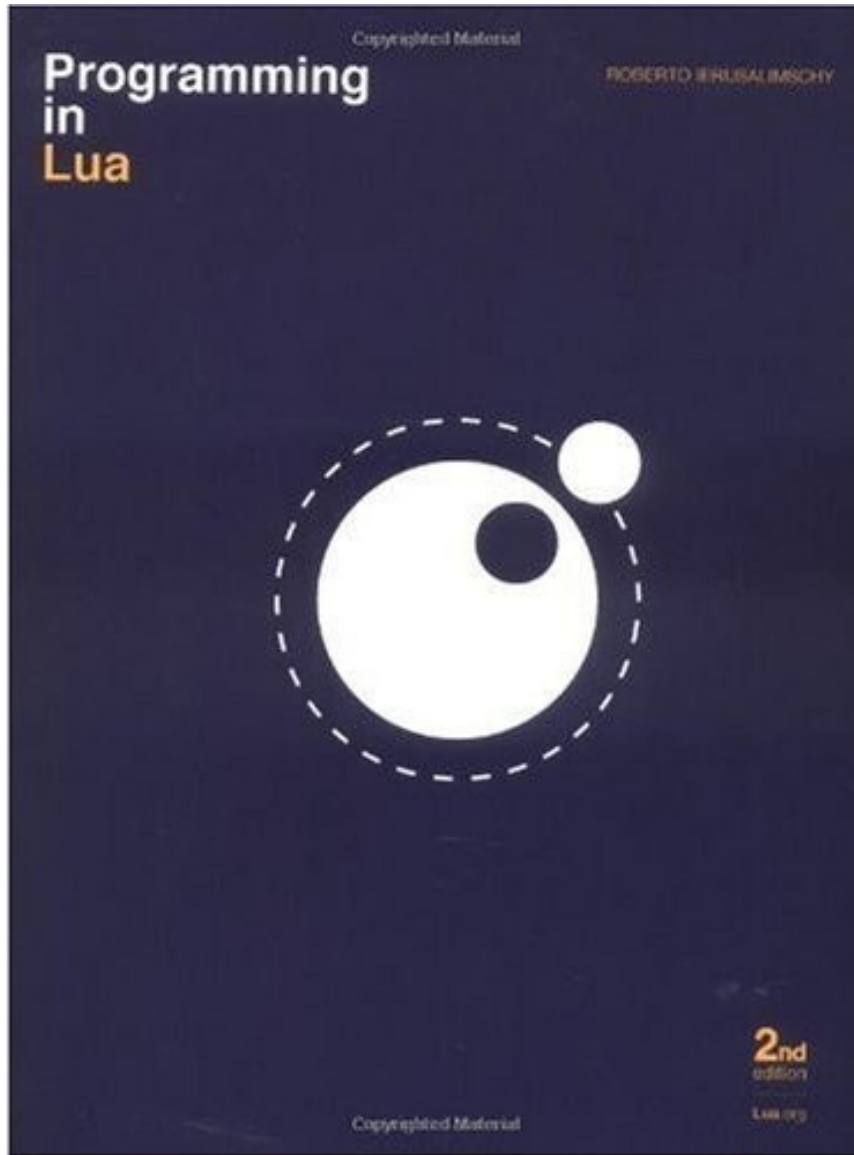
Programming in Lua.  
1<sup>st</sup> edition, 2003

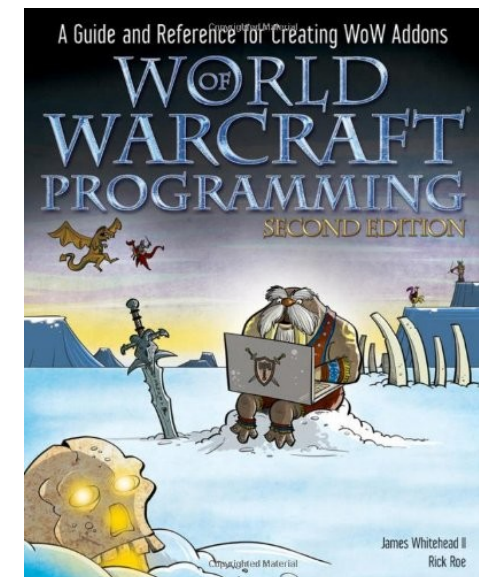
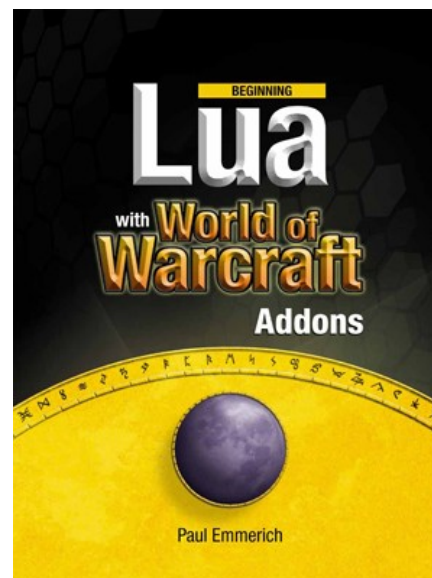
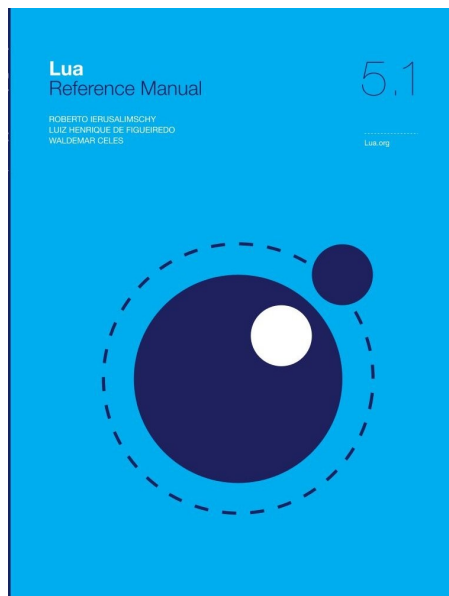
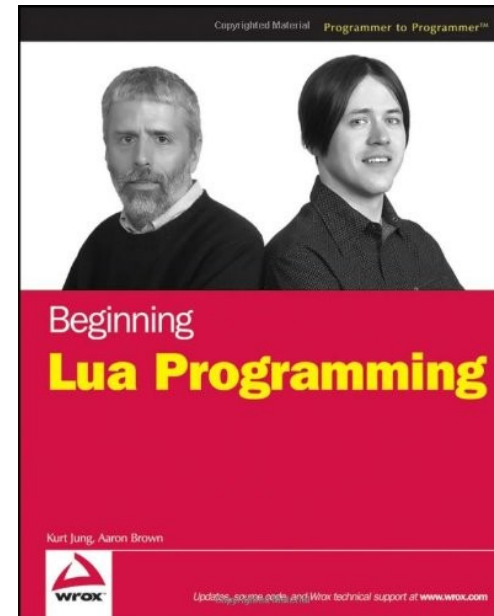
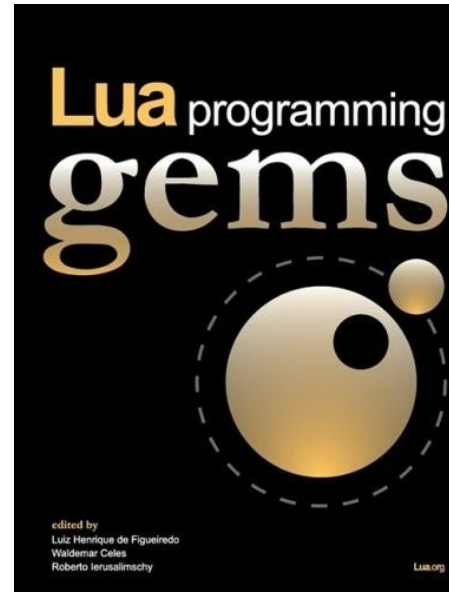
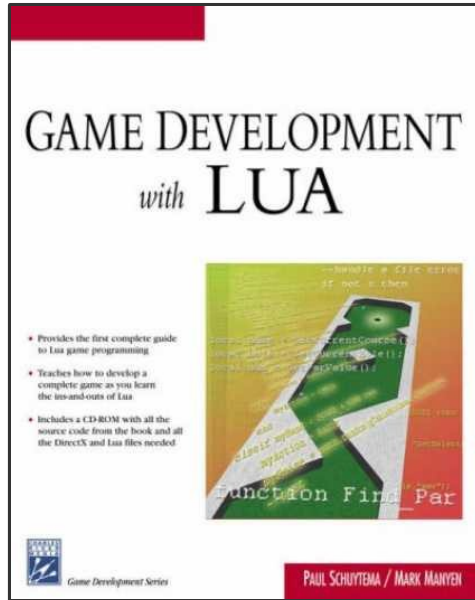
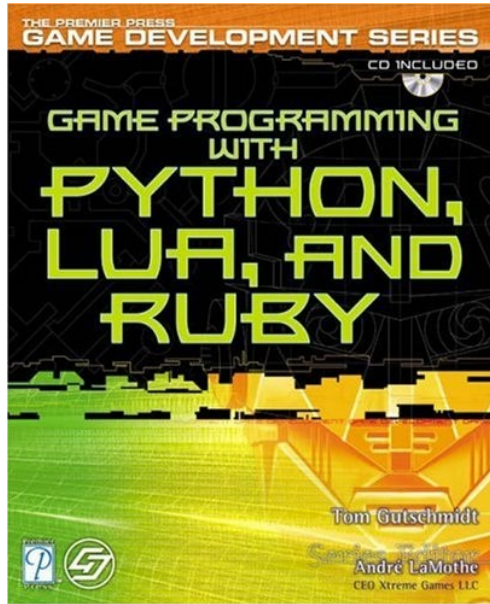


2<sup>nd</sup> edition, 2006









# So, What About Lua?

- Emphasis on scripting
- Small and simple
- Leading scripting language in games
- Very popular in embedded devices
- Several other big applications
- Virtually unknown in some circles
  - both games and embedding have some cultural barriers
  - not a “Web language”?



[www.lua.org](http://www.lua.org)