



A Evolução de Lua

Roberto Ierusalimschy

PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO



Lua

- O que
- Onde
- Por que
- Como



O Que é Lua

- Mais uma linguagem dinâmica
 - alguma similariedade com Perl, Python, Tcl
- Uma linguagem de descrição de dados
 - anterior a XML
- Ênfase em scripting
 - \neq linguagem dinâmica!
 - ênfase em comunicação inter-linguagens
 - enfatiza desenvolvimento em múltiplas linguagens

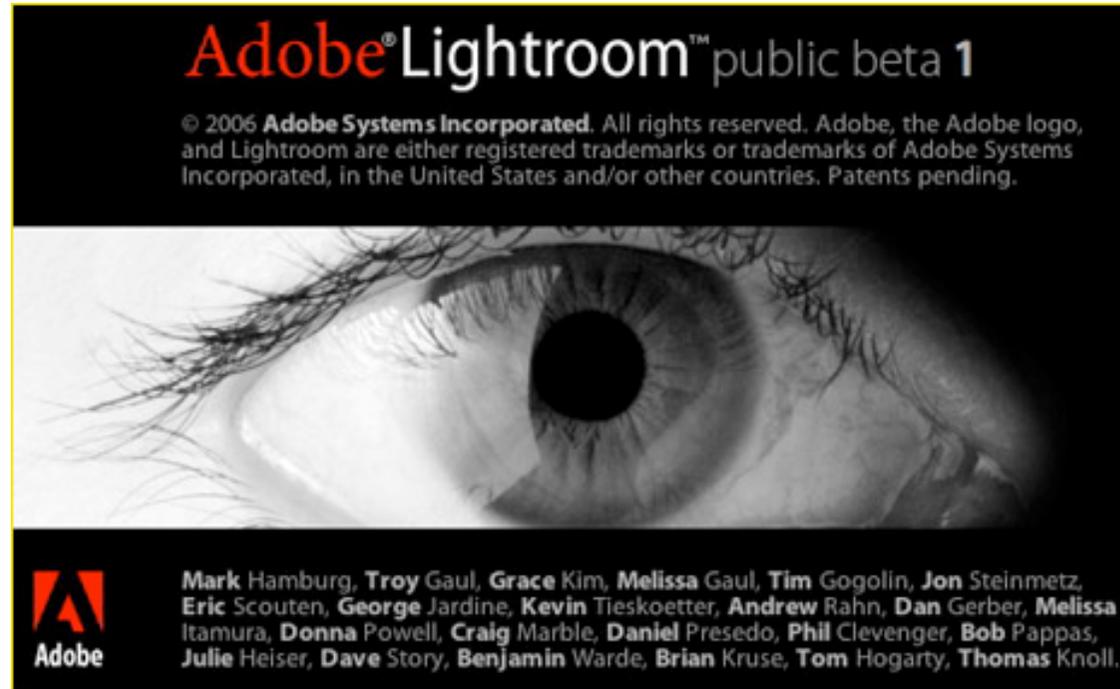
Onde Lua é Desenvolvida

- Desenvolvida na PUC-Rio
 - desde 1993
- Início modesto, para uso interno
 - expansão lenta e gradual
- "Comitê" de três pessoas
 - Roberto Ierusalimschy, Luiz H. de Figueiredo, Waldemar Celes

Onde Lua é Usada

- Todo tipo de aplicação
- Nicho em jogos





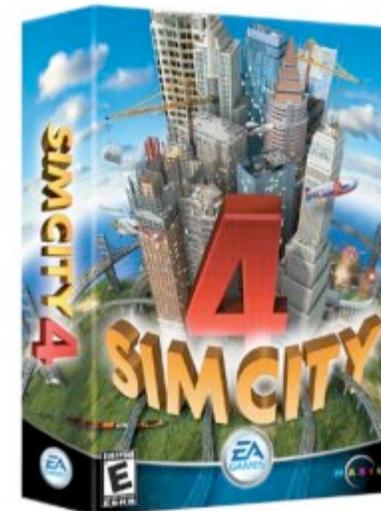
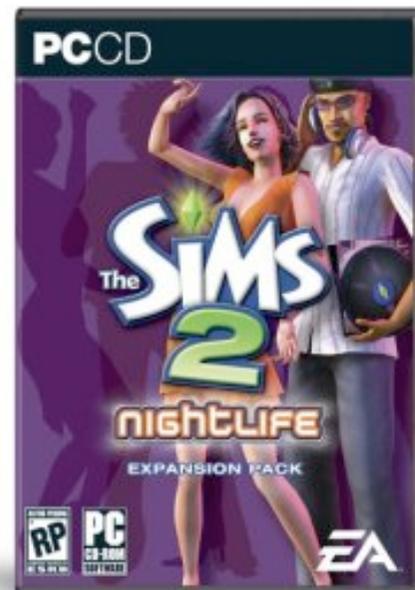
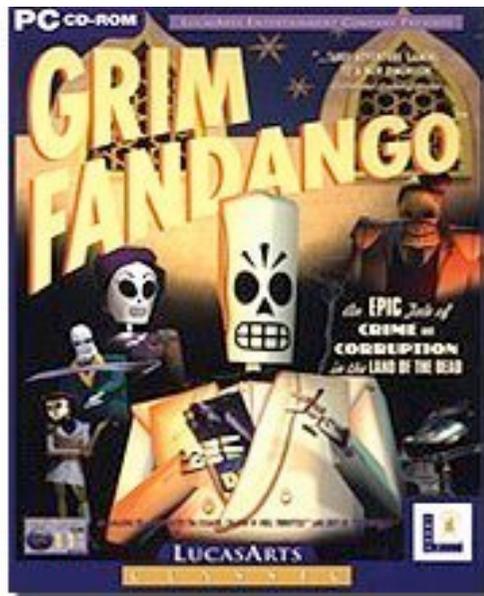
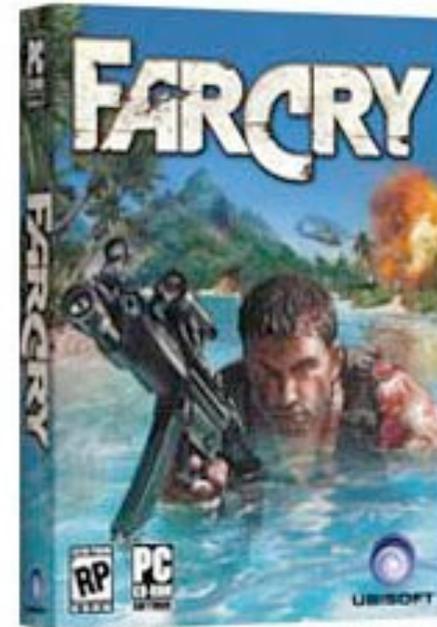
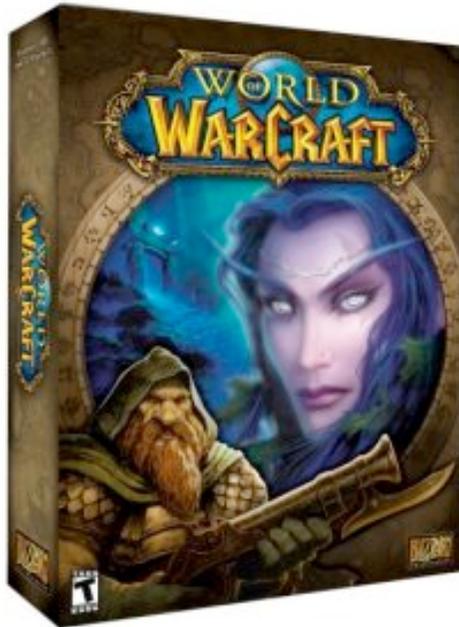
"63% of the main Lightroom-team authored code is Lua", Troy Gaul, Adobe

- Ganga
 - middleware padrão brasileiro para TV digital
- Wireshark
 - analisador de protocolos
- Snort
 - intrusion detection and prevention system
- nmap
 - rastreador de redes para segurança
- Eyeon's Digital Fusion
 - pós-produção de filmes
- E muitas outras...

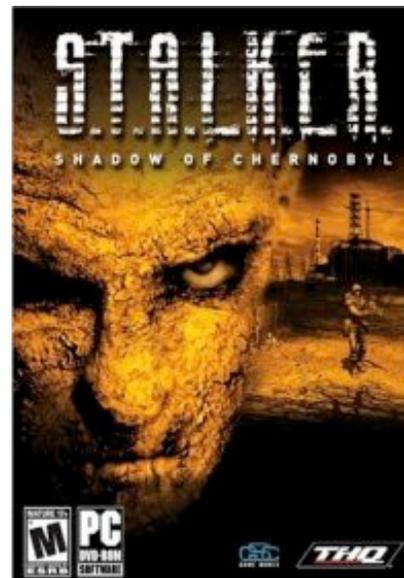
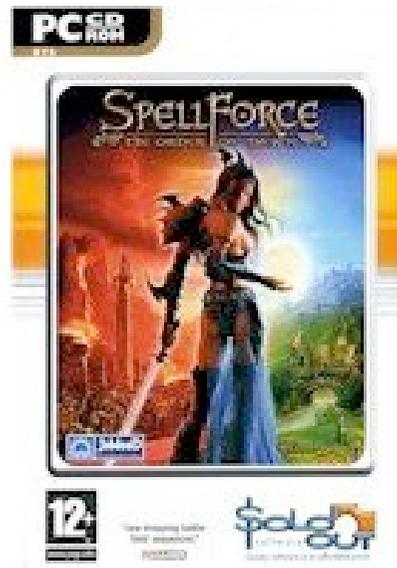
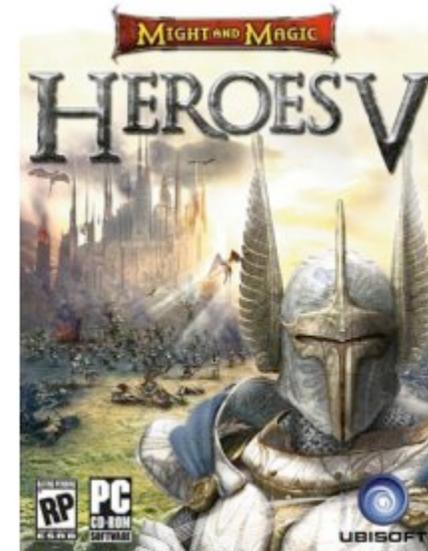
Lua em Jogos

- "It is easy to see why Lua is rapidly becoming the de facto standard for game scripting." *Artificial Intelligence for Games*, Morgan Kaufmann, 2006.
- "It's quite possible that game developers will look back at the 2000s as the decade of Lua"., *Game Programming Gems 5*, Charles River Media, 2005.

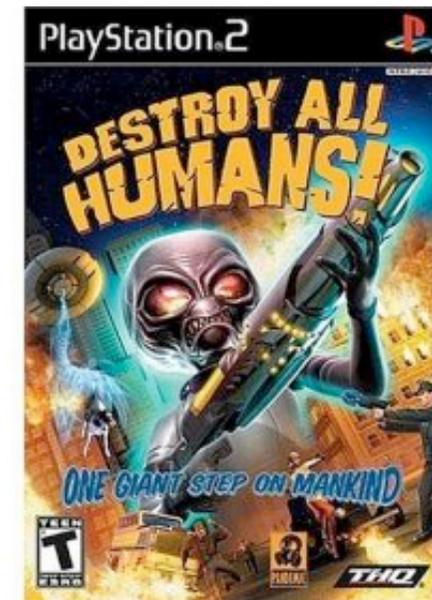
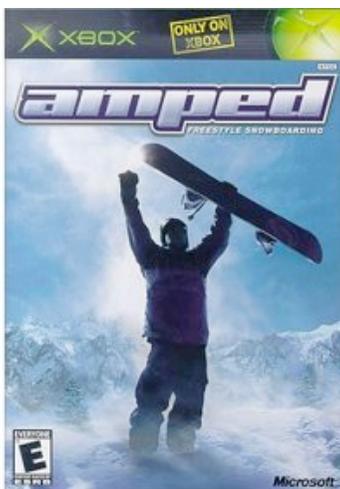
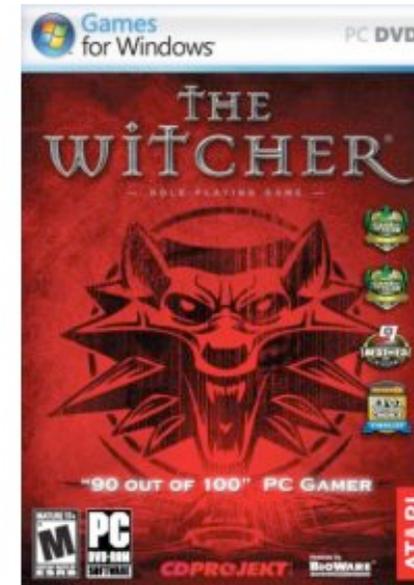
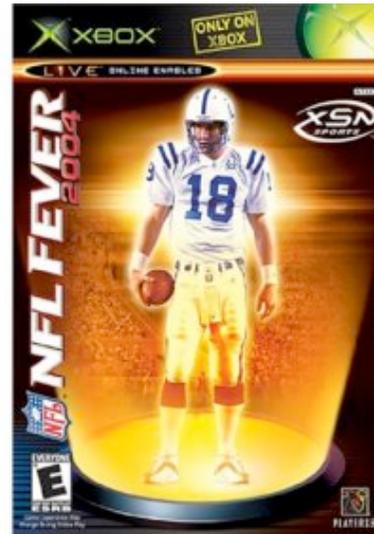
Alguns Jogos que usam Lua



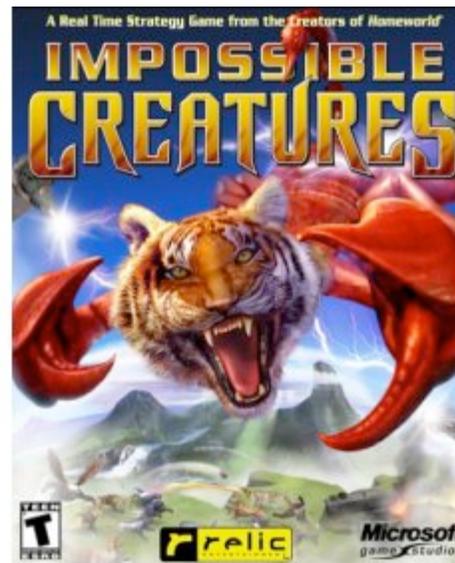
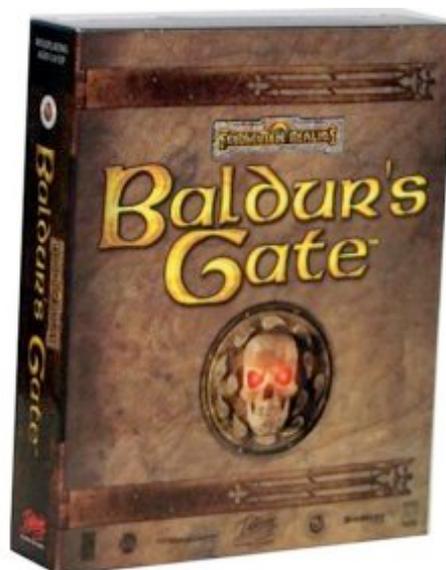
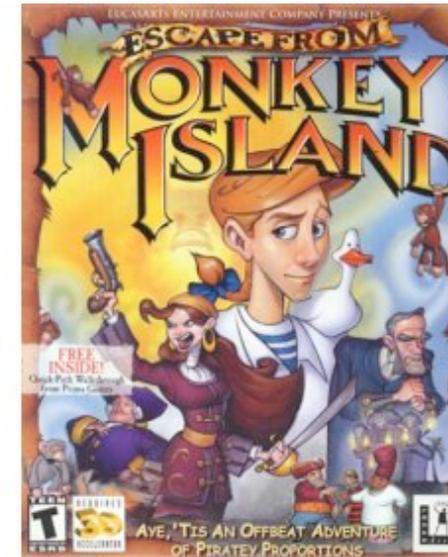
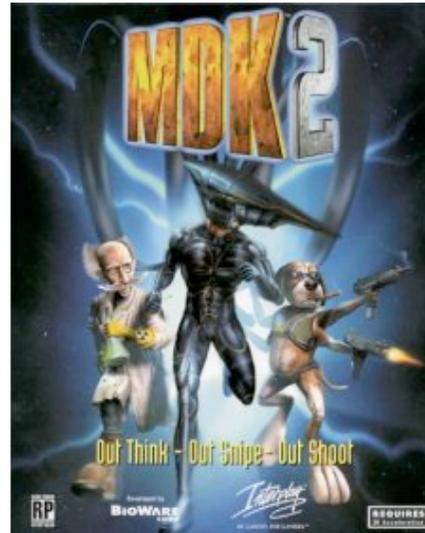
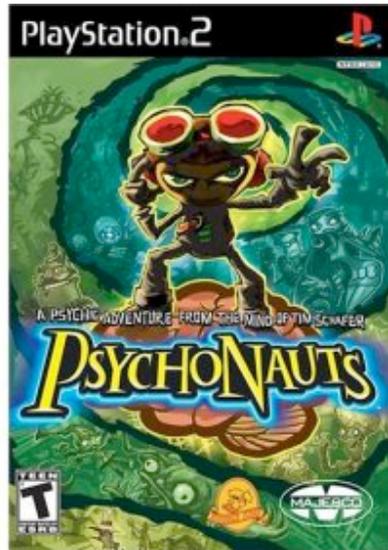
Alguns Jogos que usam Lua



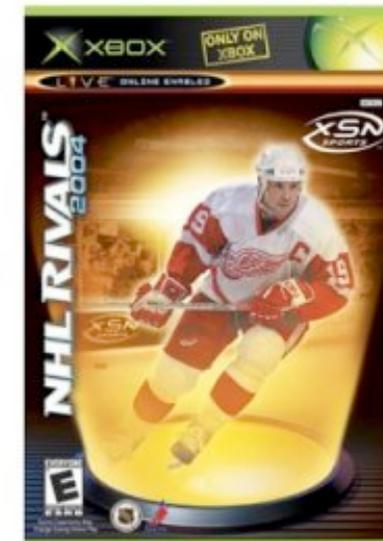
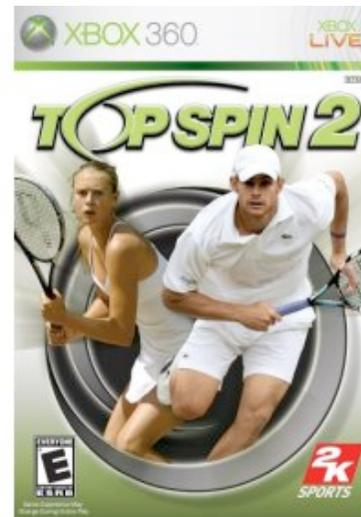
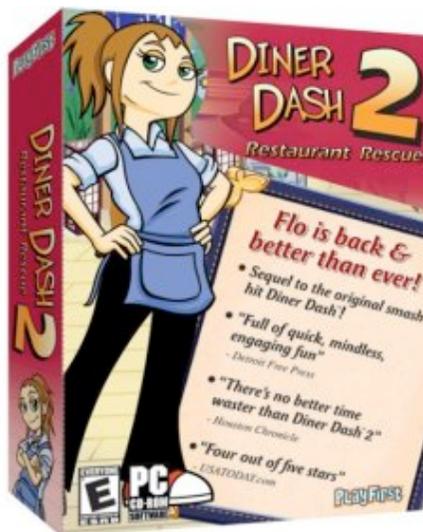
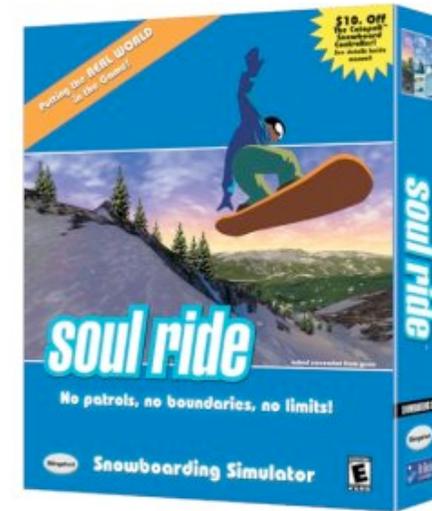
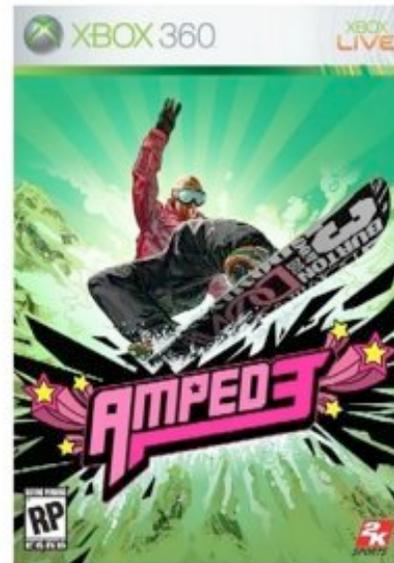
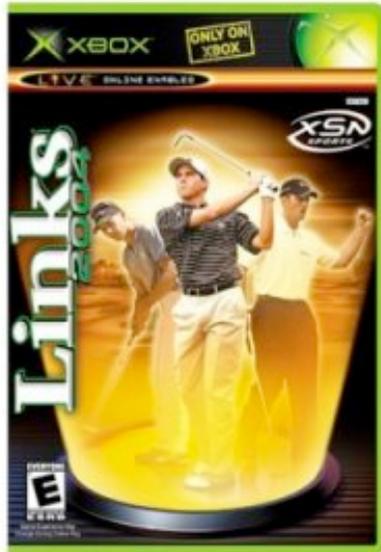
Alguns Jogos que usam Lua



Alguns Jogos que usam Lua



Alguns Jogos que usam Lua



Porque Lua

- Portabilidade
- Simplicidade
- Pequeno tamanho
- “Acoplabilidade” (embedding)
 - scripting
- Eficiência

Portabilidade

- Roda em praticamente todas as plataformas que já ouvimos falar
 - Unix, Windows, Windows CE, Symbian, BREW, hardware dedicado, Palm, PSP, etc.
- Escrita em $\text{ANSI C} \cap \text{ANSI C++}$
 - evita `#ifdefs`
 - evita pontos obscuros do padrão
- Núcleo é praticamente uma aplicação *free-standing*

Simplicidade

- Um único tipo de estrutura de dados
 - tabelas
- Um único tipo numérico
 - tipicamente double
- Mecanismos ao invés de políticas
 - e.g., orientação a objetos

Pequeno Tamanho

- Menos de 200K
- Distribuição completa (tar.gz) com menos de 210K
- Núcleo + bibliotecas
 - interface bem definida
 - núcleo com menos de 100K
 - bibliotecas independentes (e removíveis)

Acoplabilidade

- Lua é uma biblioteca C
- API simples e bem definida
 - tipos simples
 - operações primitivas
 - modelo de pilha
- Bi-direcional!
- Acoplada em C/C++, Java, Fortran, C#, Perl, Ruby, Ada, etc.

Eficiência

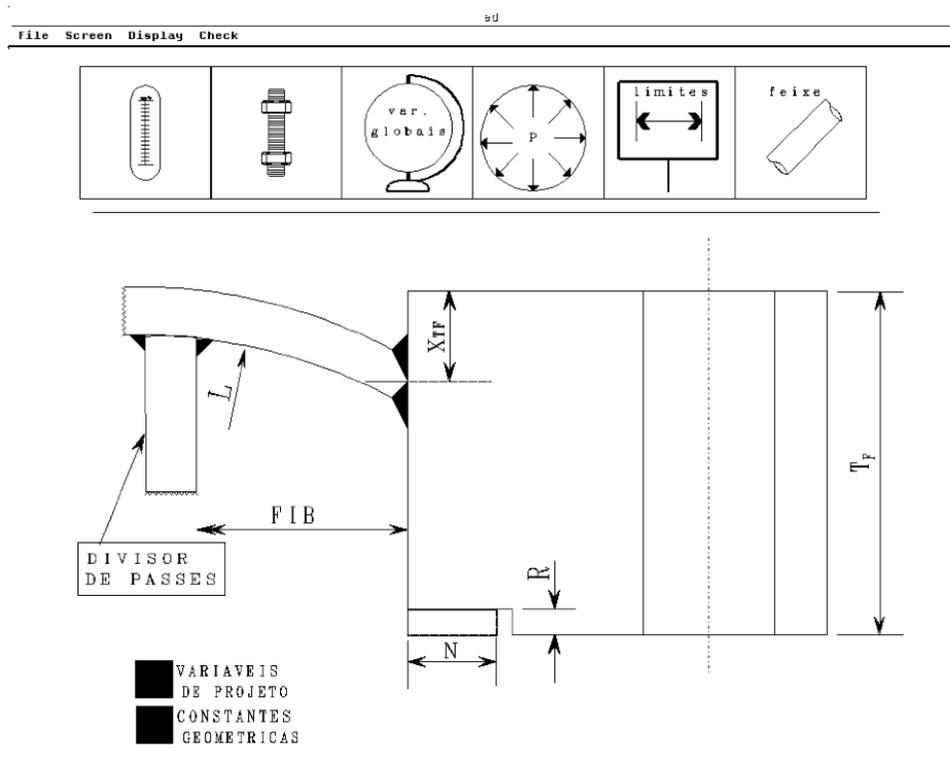
- Benchmarks independentes mostram Lua entre as mais rápidas no grupo de linguagens interpretadas com tipagem dinâmica
- Mistura de algumas técnicas especiais e simplicidade

O Início de Lua



Interfaces Gráficas

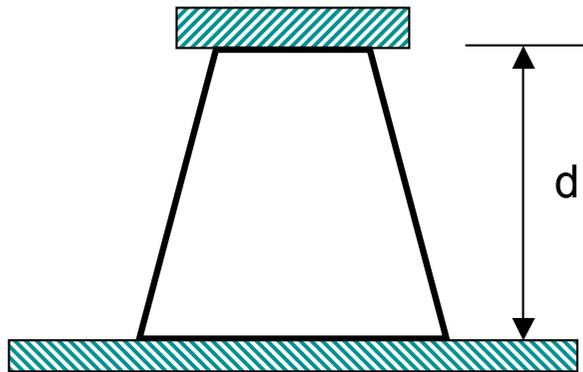
- 1992: Tecgraf - projeto entre PUC e PETROBRAS (CENPES)
 - Construção de interfaces gráficas para diversos programas de simulação



DEL - Linguagem para Especificação de Diálogos



- Definição de formulário
 - lista de parâmetros
 - tipos e valores default



```
:e gasket "gasket properties"  
mat      s                # material  
d        f      0         # distance  
y        f      0         # settlement stress  
t        i      1         # facing type
```

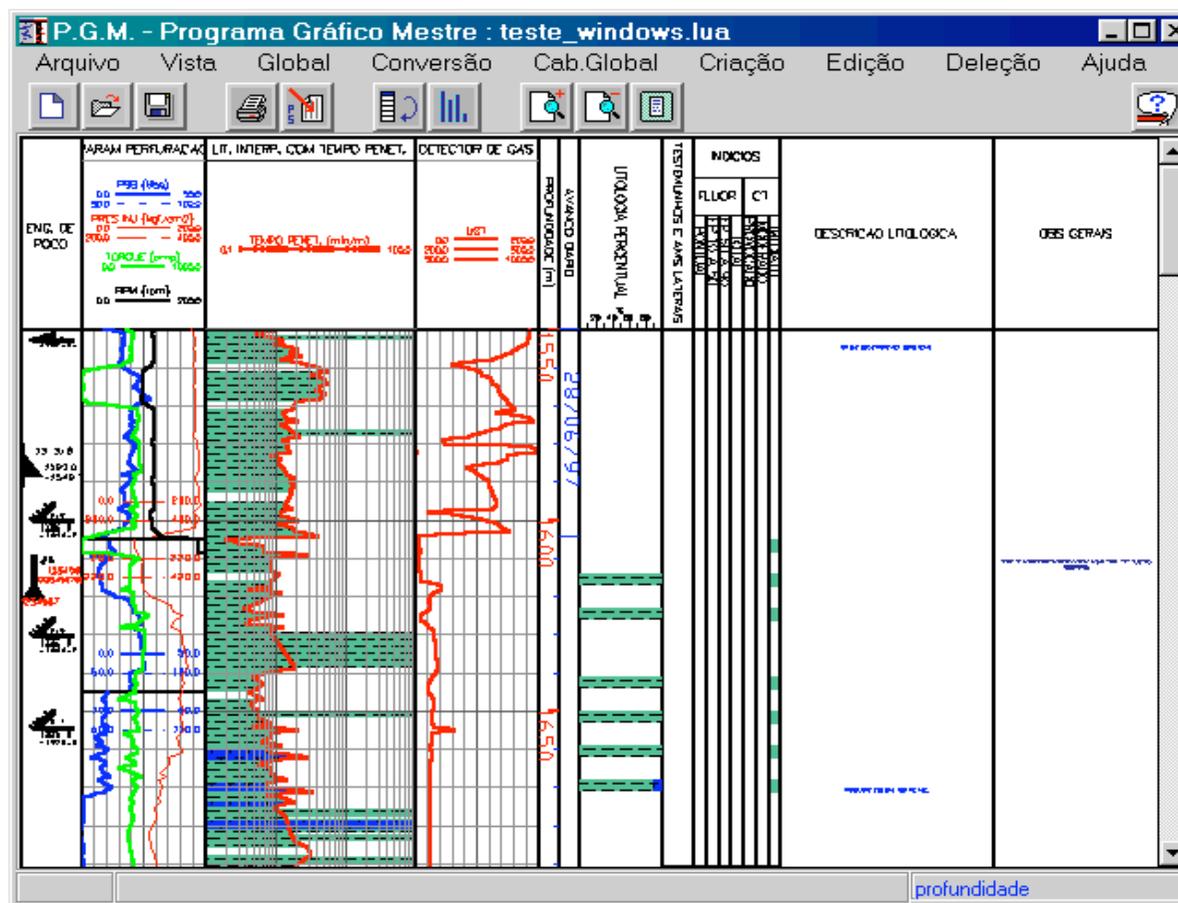
Limitações de DEL



- Tomada de decisão
- Predicados
- Necessidade de maior poder de expressão

Programa Gráfico Mestre

- 1993: Projeto com a PETROBRAS
 - programa para visualização de perfis geológicos
 - configurável



SOL

Simple Object Language



- Linguagem para descrição de objetos
 - sintaxe inspirada no BibTeX

```
type @track {x:number, y:number=23, z}

type @line {t:@track=@track{x=8}, z:number*}

-- create an object 't1', of type 'track'
t1 = @track{y=9, x=10, z="hi!"}

l = @line{t=@track{x=t1.y, y=t1.x}, z=[2,3,4]}
```

Limitações de SOL



- Recursos para construção de diálogos
- Mecanismos de programação procedural
- Mecanismos de abstração mais poderosos

1993

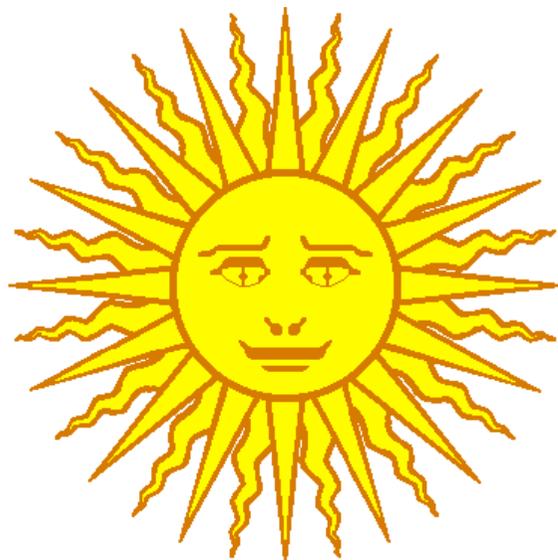


- DEL e SOL tinham vários problemas em comum
- Roberto Ierusalimschy (PGM), Luiz Henrique de Figueiredo (DEL) e Waldemar Celes (PGM) se juntaram para achar uma solução comum a seus problemas...

O Que Precisávamos?

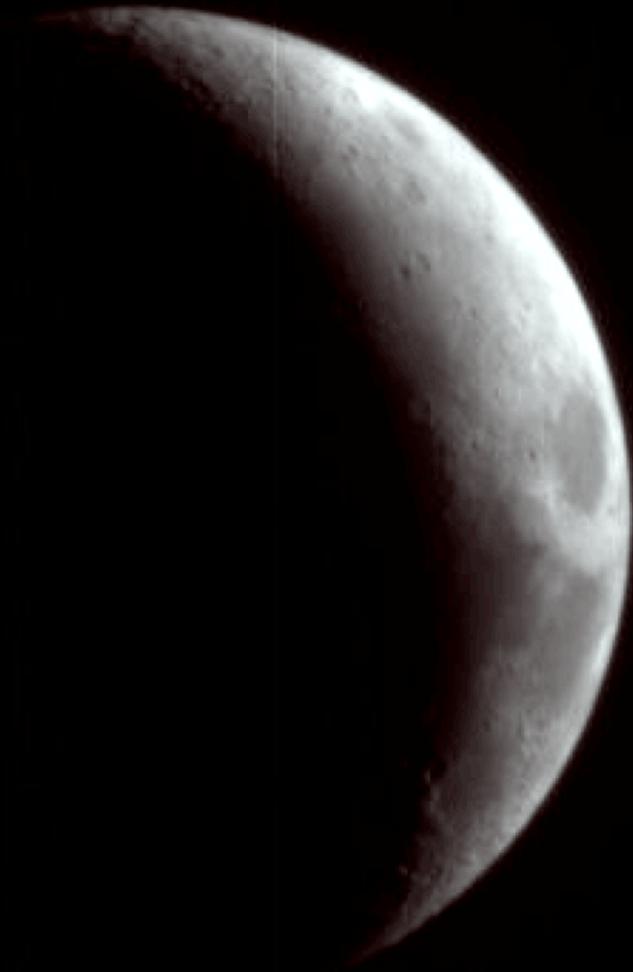
- Uma "linguagem de configuração genérica"
- Uma linguagem "completa"
- Facilmente acoplável
- Portátil
- o mais simples possível
- sintaxe não intimidante
 - para usuários finais (engenheiros, geólogos, etc.)

Como estávamos largando Sol,



um amigo sugeriu um novo nome...

...e surgiu Lua



Como era Lua 1.0?

- Não muito diferente de SOL...

```
t1 = @track{x = 10.3, y = 25.9,  
           title = "depth"}
```

Como era 1.0?

- Mas bem diferente...

```
t1 = @track{x = 10.3, y = 25.9,  
           title = "depth"}
```

```
function track (t)  
  if not t.x then t.x = 0.0 end  
  if type(t.x) ~= "number" then  
    print("invalid 'x' value")  
  end  
  if type(t.y) ~= "number" then  
    print("invalid 'y' value")  
  end  
end  
end
```

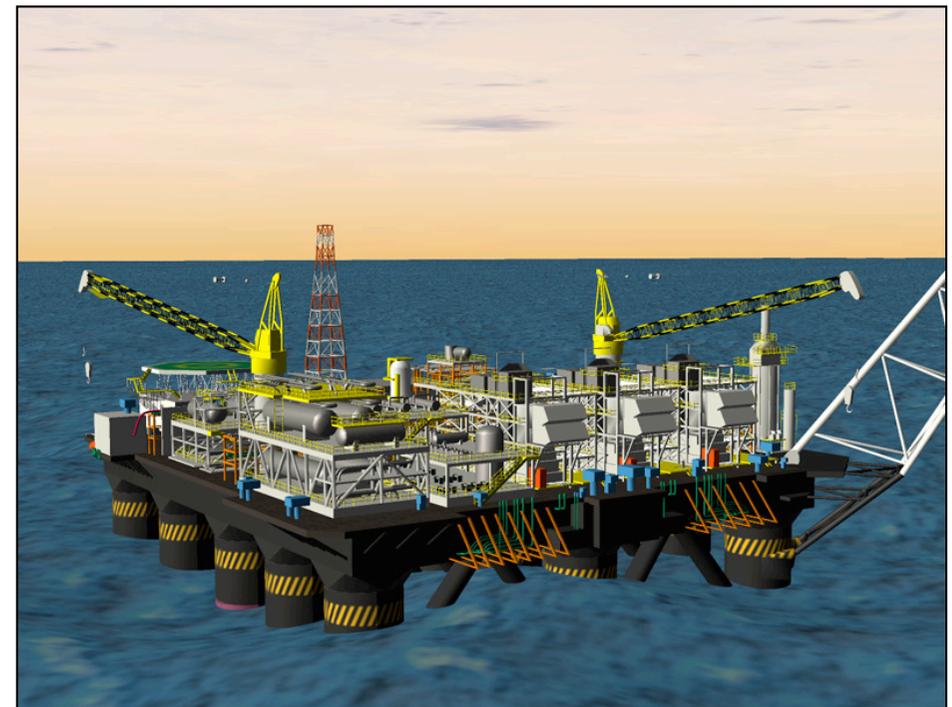
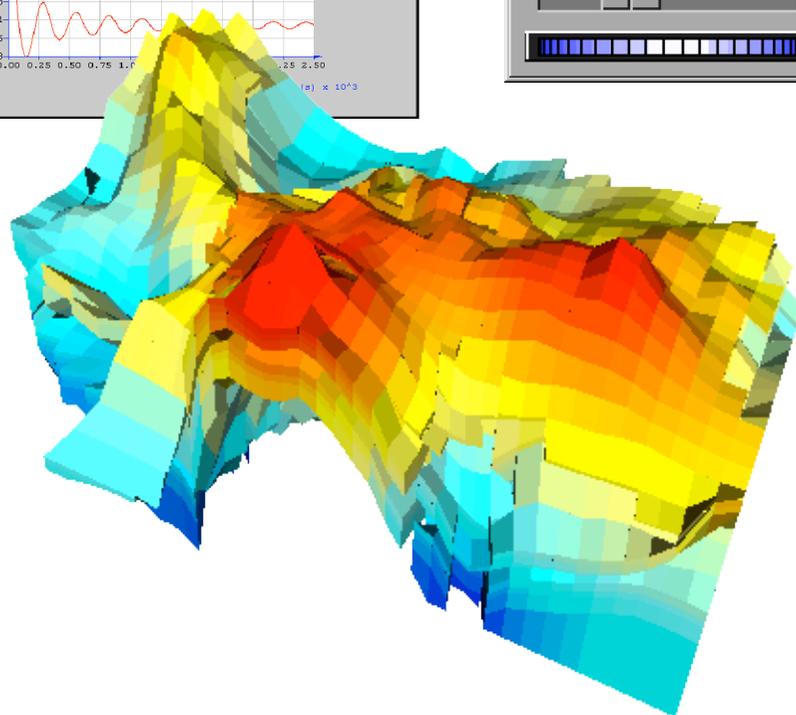
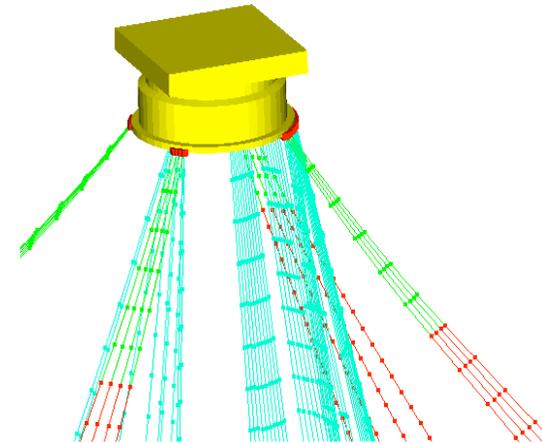
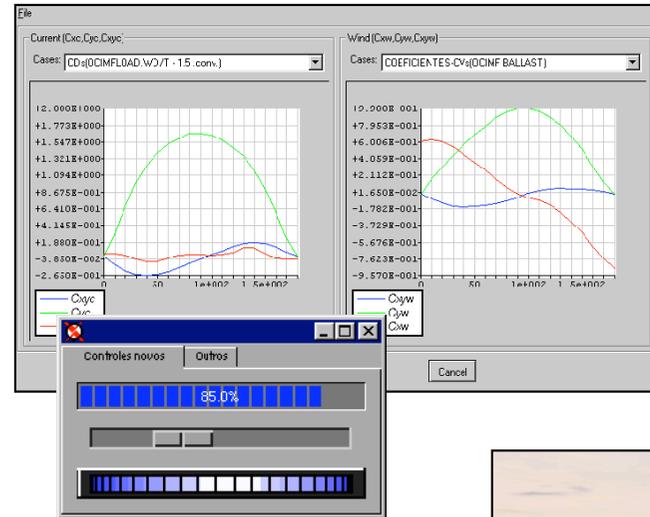
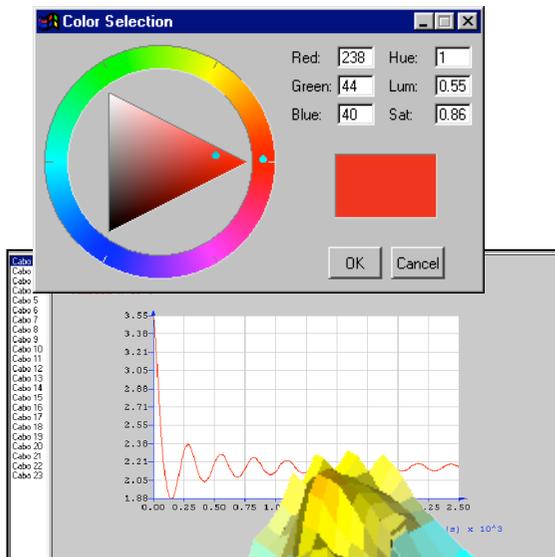
Lua 1.0

- Implementada como uma biblioteca
- Chamada de 1.0 posteriormente
- "The simplest thing that could possibly work"
 - tabelas implementadas via listas!
- Implementação padrão
 - pré-compilador com yacc/lex
 - opcodes para uma máquina virtual baseada em pilha
- Menos de 6000 linhas de código C

Lua 1.0

- Expectativas: solucionar nossos problemas com PGM e DEL
 - poderia ser usada em outros projetos do Tecgraf
- Satisfez nossas expectativas
 - tanto DEL quanto PGM usaram Lua com sucesso
 - PGM ainda em uso hoje
- Foi um grande sucesso no Tecgraf

Logo, vários projetos no Tecgraf estavam usando Lua



Lua 1.1

- Já tínhamos vários "usuários reais"
- Novos usuários trazem novas demandas
- Manual de referência
- API com C bem definida e bem documentada

Lua 2

- De Lua 2.1 (02/1995) até 2.5 (11/1996)
- Suporte para OO
 - delegação
- Casamento de padrões
 - biblioteca própria
- Interface para depuração



Exposição Internacional

- anúncio em grupos *Usenet*
- primeira página na Web em 1995
 - <http://www.inf.puc-rio.br/~roberto/luas>
- primeiros contatos por e-mail com usuários remotos
- em 1997: lista de discussões
 - permitir usuários responderem perguntas
 - final de 97: mais de 100 assinantes: devíamos tentar um novo grupo?

Exposição Internacional



- junho 1996 - artigo na S:P&E
 - R. Ierusalimschy, L. H. de Figueiredo, W. Celes, Lua - an extensible extension language, *Software: Practice & Experience* 26(6):635-652, 1996.
- dezembro 1996 - artigo na Dr. Dobb's
 - L. H. de Figueiredo, R. Ierusalimschy, W. Celes, Lua: an extensible embedded language, *Dr. Dobb's Journal* 21(12):26-33, 1996.

Lua em Jogos (o início)

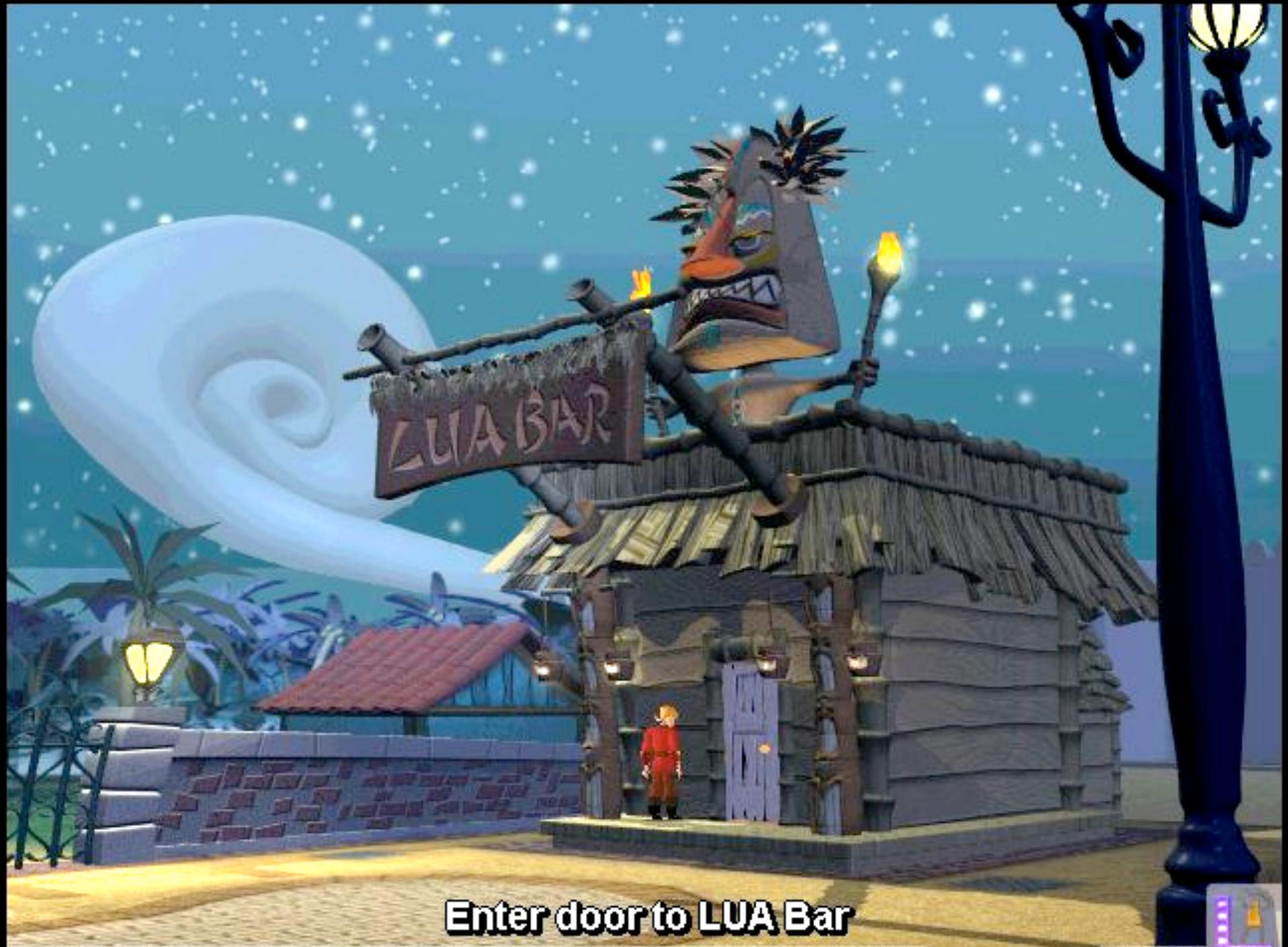


From: Bret Mogilefsky <mogul@lucasarts.com>
To: "'lua@icad.puc-rio.br'" <lua@icad.puc-rio.br>
Subject: LUA rocks! Question, too.
Date: Thu, 9 Jan 1997 13:21:41 -0800

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.



Enter door to LUA Bar

Lua 3

- De Lua 3.0 (07/1997) até Lua 3.2 (07/1999)
- Melhor suporte a funções
 - funções anônimas, *upvalues*
- Grande reorganização interna

Lua 4

- Uma única versão: Lua 4.0, 11/2000 até 04/2003
- Nova API com C
 - baseada em pilha
- Múltiplos estados
- Todas as bibliotecas implementadas via API oficial

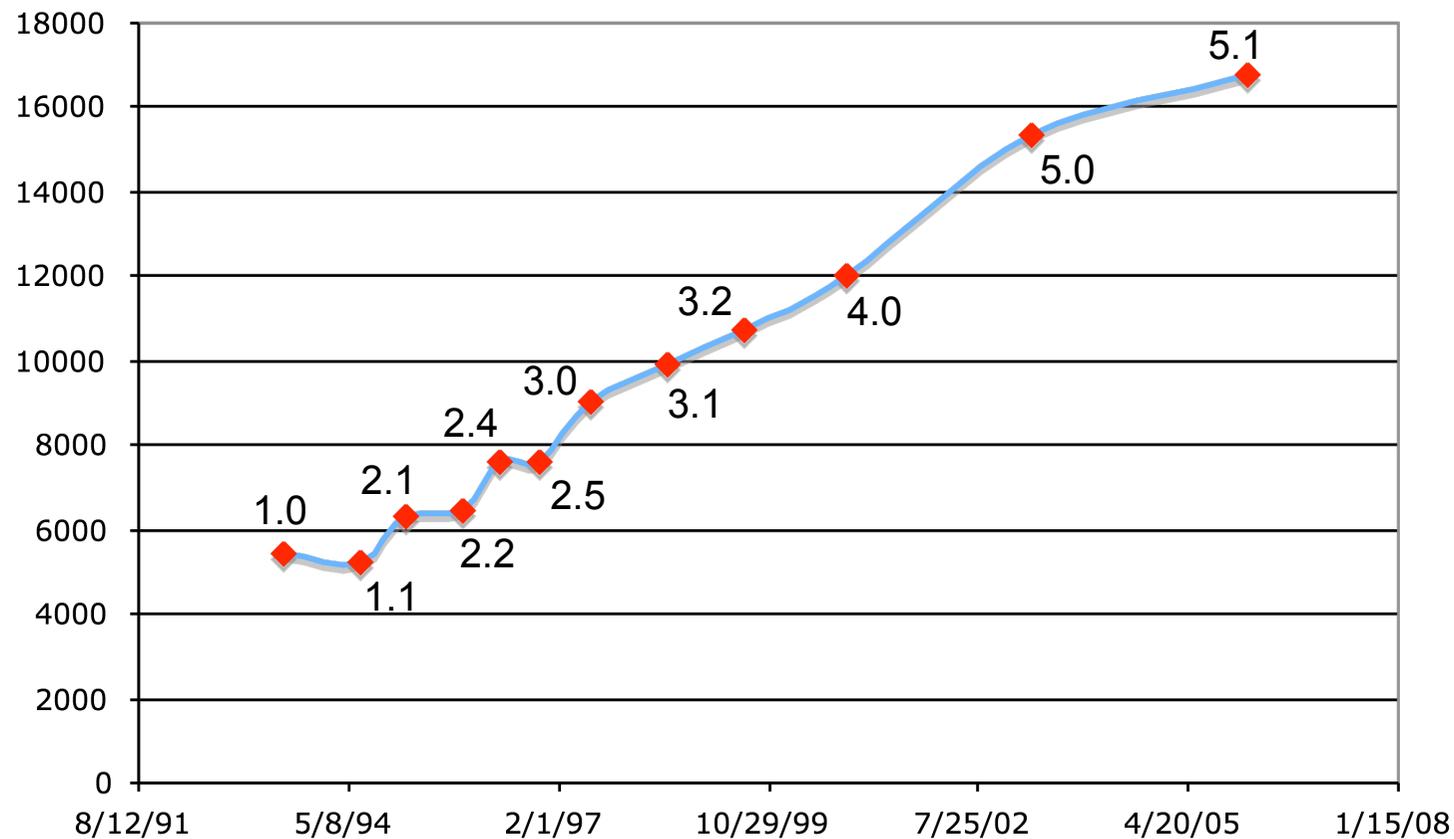
Lua 5

- Lua 5.0 (04/2003) e 5.1 (02/2006)
- "Maturidade" da linguagem
 - livro texto
- Co-rotinas
- Sistema de módulos
- Funções anônimas com escopo léxico
- Máquina virtual de registradores
- Coleta de lixo incremental

Crescimento em Linhas de Código



- aproximação para complexidade...



Evolução



- Ênfase nos requisitos originais
 - "minimalismo pragmático"
 - implementação "free-standing"
 - mecanismos mais gerais
 - ênfase em funções
- Aspectos não funcionais
 - documentação, robustez, mensagens de erros, ...

Lua Atualmente

- Site com >4.000 visitas por dia
 - de mais de 100 países
 - >14.000 downloads por mês
- Compromissos fortes com estabilidade
 - Lua 4.0 ficou mais de dois anos sem alterações
 - Lua 5.1 lançada em fevereiro 2006

Lua Atualmente

- Licença atual: MIT
 - uso livre, não “viral”
- Modelo de Desenvolvimento centralizado
 - importante no caso de Lua, onde tamanho é fundamental (evitar efeito comitê)
- mas com grande participação da comunidade
 - lista, versões variantes, site de usuários

Lua Atualmente

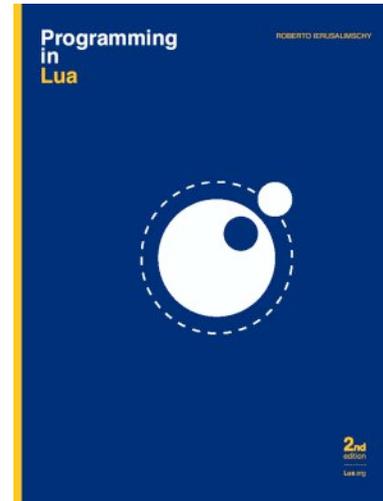
- Requisitos originais mantidos
 - simplicidade, portabilidade, extensibilidade, pequeno tamanho
- Única linguagem desenvolvida fora do eixo EUA/Europa/Japão a ser adotada mundialmente
 - entre as 20 mais populares no índice TIOBE
 - Ruby é (a única) do Japão

Livros

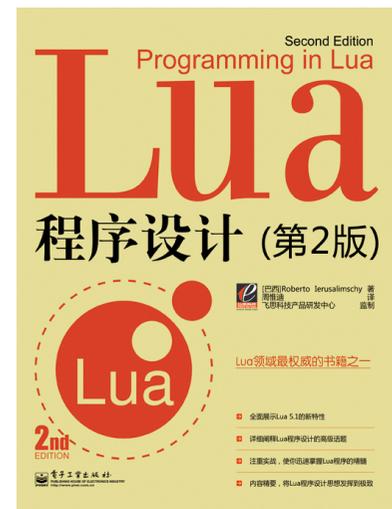
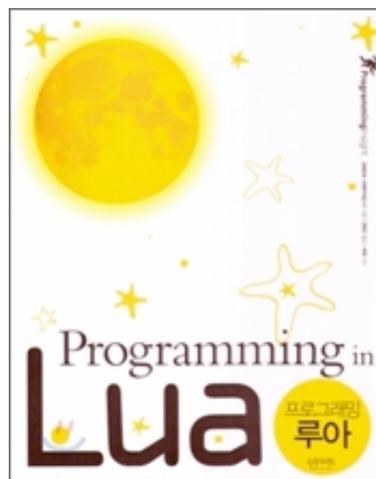
Programmieren mit Lua
Open Source Press, 2006



Programming in Lua, 2nd edition
Lua.org, 2006

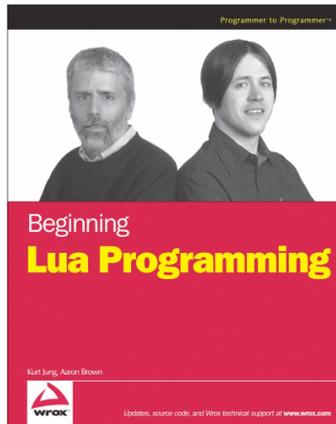


프로그래밍 루아
Insight, 2007



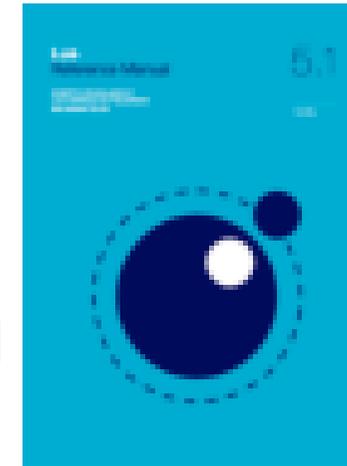
Lua程序设计 (第2版)
PHEI, 2008

Livros

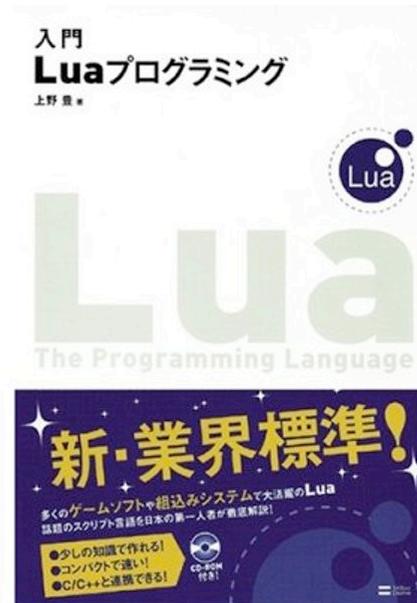


Beginning Lua Programming
Wrox, 2007

Lua 5.1 Reference Manual
Lua.org, 2006



Game Development with Lua
Charles River Media, 2005



入門 Lua プログラミング (単行本)
Softbank Creative, 2007

Para Saber Mais...



<http://www.lua.org/>