

How Lua Brought the Dead to Life

A historic footnote and dangerous precedent

Bret Mogilefsky
bmogilefsky@gmail.com

Don't sue me!

- This is a presentation about how Lua was used in Grim Fandango, an adventure game by LucasArts
- I don't work for LucasArts anymore
 - However, all materials are used with their permission (official text to follow)
- I do work for Sony Computer Entertainment America
 - However, I am not speaking in any official capacity
 - I have to go back to work later... :(

What's an adventure game?

- In this case:
 - Characters (3D models and 2D sprites)
 - Environments (pre-rendered backgrounds with z-channel)
 - Story (text display engine, cutscenes)
 - Puzzles (creative writing and design)
 - Includes 8000 lines of dialog!

It looks like this

Why use a scripting language?

- Glue!
 - 95% of animations and cutscenes are custom
 - Scripting language is key to reducing weight
 - Clean separation of engine from content
 - Example:
 - actor guybrush walk-to banana-tree
 - wait-for-actor
 - actor guybrush say-line “Mmm, bananas...”
 - actor guybrush face-camera
 - actor guybrush say-line “Wish I had a banana-picker”

What existed (I)

- SCUMM: Script Creation Utility for Maniac Mansion
 - SCUMM is really just the language
 - SPUTM (interpreter aka game engine)
 - FLEM, BYLE, etc. for creating assets
- Proven technology
 - Used in dozens of shipping titles
 - Multi-platform

What existed (II)

- Huge pedigree of beloved games
 - Maniac Mansion, Secret of Monkey Island I & II, Sam and Max, Day of the Tentacle, Full Throttle...
- Talented designers and artists

What existed (III)

- ...and me!
 - One naïve coder fresh out of school

Why screw it up then?

- Concurrent projects pushing technology envelope
 - Jedi Knight, a 3D FPS with great coders
 - Outlaws, a 2.5D FPS with great coders
 - Smart people I could steal from!
- New look needed
 - 3D was the new black
 - SCUMM didn't speak floating-point
- Nobody stopped me
 - Though they predicted doom

Lua hacks made

- I had no idea what to call these at the time
 - Cooperative tasking
 - Added language primitives for iterating over/inspecting them
 - Stackless Lua
 - Completely by accident
 - More could have been done
 - State serialization/deserialization

What parts are in Lua

- Everything that matters
 - Dialogue
 - Puzzle logic
 - UI/controls
 - Menus
- Engine handles only animations, backgrounds, sound, rendering, choreography, etc etc etc... but those aren't Grim Fandango

What went right

- Lua!
 - Time went to other systems
- Smart people I stole from
 - Rendering
 - Movie playback
 - Anything else hard

What went wrong

- Lack of disciplined Lua code
 - Heavy memory requirements
 - Difficult debugging
- Debugger went stale

What saved my ass

- Lua!
- Virtual memory
- Large caches
- Really talented artists/storytelling
 - You notice that the framerate sucks a lot less

What happened next

- Game of the year...almost
 - Half-Life relegated us the Adventure Game of the Year
- GDC 1999 (2000?)
 - Panel discussion of scripting languages
 - Rob Huebner on embedding Java
 - Kevin Bruner on interpreted C++
 - Seamus McNally on not using a scripting language
 - 200 miserable people
 - “Or you could just use Lua...”
 - Furious scribbling

Lua in my life now

- I haven't read the mailing list in four or five years :(
- But... Lua keeps appearing
 - PS3 toolset
 - PS2 network setup disc
 - Various games
 - Mojib Ribbon
 - Psychonauts
 - America's Army
- I am honored to be here

Shout-outs

- LEC folk
 - Tim Schafer
 - Designer/scripter
 - Kevin Bruner
 - Co-engine programmer
 - Chuck Jordan and Chris Purvis
 - Lua scripting and some tools
 - Winston Wolff
 - Suggested I read the Dr. Dobb's article
- Lua mailing list

