

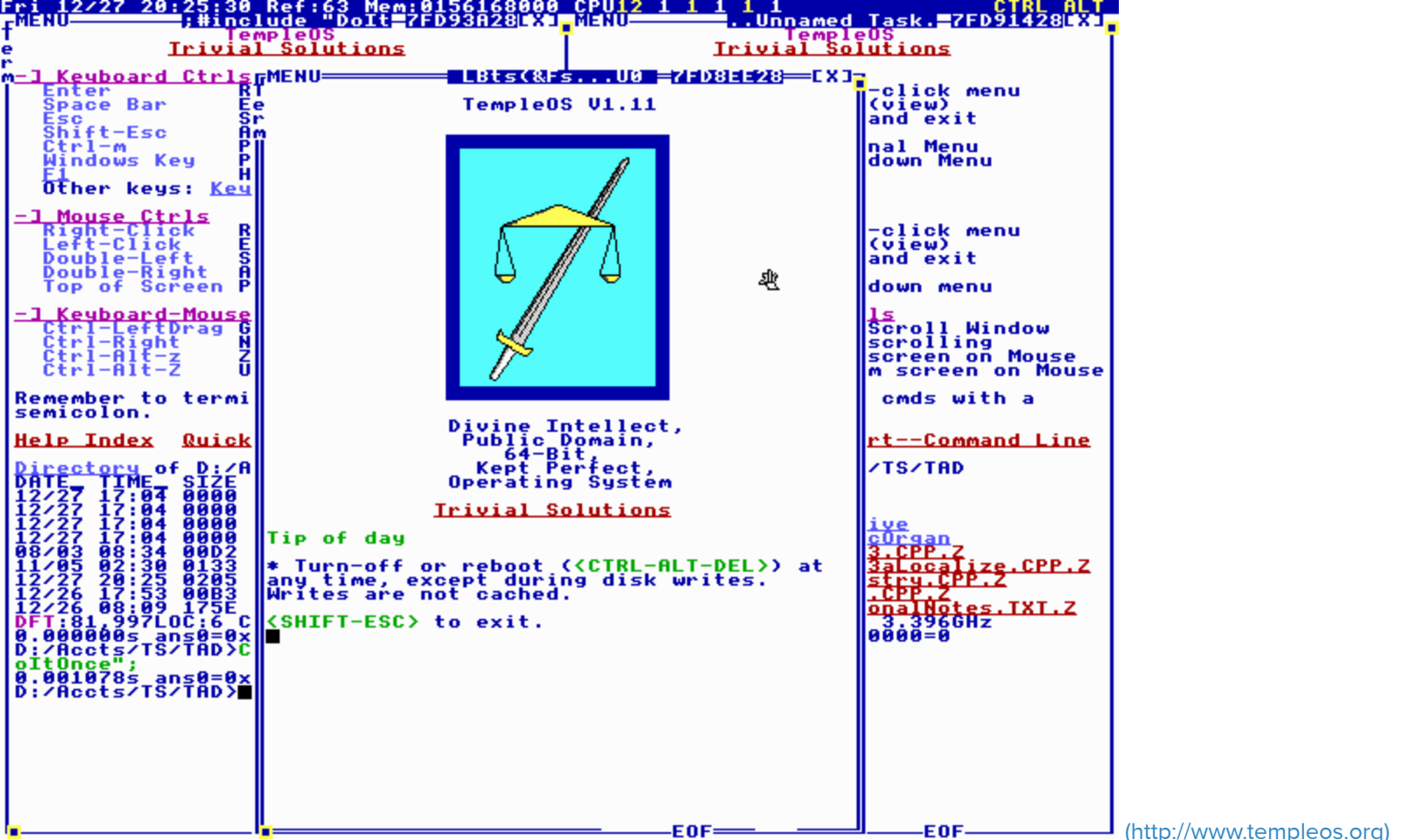
TempleOS: an educational tool for programming experiments

James Sanders explores what drives one man to write an operating system for x86_64, and what the programmer's finished product looks like.

By James Sanders | in Software Engineer, January 20, 2014, 11:33 PM PST

Creating an entire operating system from scratch is not a task that programmers bother with under normal circumstances. This isn't to say most programmers aren't capable of creating an OS — I think many could do it, given enough time. Linux was "just a hobby" of Linus Torvalds, who sat down and, over the span of several months, wrote the Linux kernel in August 1991.

Another project that started as one programmer's hobby in 1993 has, through brief periods of development and longer periods of being shelved, culminated in a new, stable operating system called [TempleOS](http://www.templeos.org/Wb/Accts/TS/Wb2/TempleOS.html) (<http://www.templeos.org/Wb/Accts/TS/Wb2/TempleOS.html>). According to the author, Terry Davis, it's "an x86_64, multi-tasking, multi-cored, public domain, open source, (http://en.wikipedia.org/wiki/Ring_%28computer_security%29)ring-0-only (http://en.wikipedia.org/wiki/Ring_%28computer_security%29), single-address-map ([identity-mapped](http://en.wikipedia.org/wiki/Identity_(mathematics)) ([http://en.wikipedia.org/wiki/Identity_\(mathematics\)](http://en.wikipedia.org/wiki/Identity_(mathematics)))), non-networked, PC operating system for recreational programming." It ships with its own compiler, and modified version of C++ ("more than C, less than C++") called "HolyC," and file system "Red Sea," along with FAT32 support. It also has 8-bit ASCII support and a 2D and 3D graphics library, all of which run at 640x480 (VGA) with 16 colors, and outputs all sound through one-voice PC speaker. The entire project is self-hosting, so you can (<http://www.templeos.org/Wb/Demo/MakeDistro.html#1>) (<http://www.templeos.org/Wb/Demo/MakeDistro.html#1>) compile your own distribution inside TempleOS (<http://www.templeos.org/Wb/Demo/MakeDistro.html#1>).



(<http://www.templeos.org>)

At present, all of that, plus extensive documentation and demos, all happen in 121,691 lines of code; demos use 22,242 lines, and the examples for the music organ composing software is 4,292 lines. Ignoring that, TempleOS fits in fewer than 100,000 lines, "as God intended," according to Davis.

The demos are impressive, though as one would expect considering the size of TempleOS and the limitations of VGA graphics, they are rather barebones. A fully-3D first person shooter called [Castle Frankenstein](http://www.templeos.org/Wb/Demo/Games/CastleFrankenstein.html#11) (<http://www.templeos.org/Wb/Demo/Games/CastleFrankenstein.html#11>) exists in 594 lines of code, complete with a villain and a potted plant. (The potted plant is more creative than the [shipping crate trope](http://tvtropes.org/pmwiki/pmwiki.php/Main/CrateExpectations) (<http://tvtropes.org/pmwiki/pmwiki.php/Main/CrateExpectations>),.) Perhaps more impressive is [Eagle Dive](http://www.templeos.org/Wb/Demo/Games/EagleDive.html#11) (<http://www.templeos.org/Wb/Demo/Games/EagleDive.html#11>), a flight simulator of an eagle diving for fish, in full 3D, at 30 frames per second.

Why TempleOS exists

You might be wondering why Davis created the OS. The reason seems to be twofold.

First, his vision for TempleOS is something akin to a modern Commodore 64: It's simple to understand, manipulate, and work within. In essence, it's an educational tool for programming experiments. Drawing a line onscreen in TempleOS and on the Commodore 64 is about as equally complex for the end user. It's barebones by design — you have direct access to the hardware, and you can do what you like with it, within

the limitations of the hardware. This is a laudable goal, and one solution to the complexity of programming modern computers that programmers, hardware designers, and educators are seeking to solve. Most famously, the [Raspberry Pi](https://www.techrepublic.com/pictures/raspberry-pi-inside-the-pi-factory/1/) (<https://www.techrepublic.com/pictures/raspberry-pi-inside-the-pi-factory/1/>) is the modern answer to the lack of cheap educational development hardware.

Second, the motivation is rather personal to Davis. If you haven't noticed the litany of Biblical references like HolyC and Red Sea and the primary persistent resident task Adam, the entire OS is, in essence, a platform for offerings to and communications with God. To that extent, one of the more prominent bundled programs is After Egypt, in which one walks up the mountain (Labeled Mt. Horeb) to talk to God, but must first dodge sheep and trees in order to find the burning bush, which in VGA, is represented by something that resembles a ball of fuzz that randomly redraws and switches colors (all 16 of them).

Once you reach the burning bush, you can use what Davis calls a "high-speed stopwatch," which is linked to a wordlist to generate pseudo-random text, as an "oracle," which Davis defines as being like an Ouija board or Christian tongues (Ouija is a trademark of Hasbro, Inc.). The results are as random as one can expect with a lookup function from a predefined wordlist — the software isn't attempting to assemble syllables or morphemes into a generated language, nor does it attempt to fit English grammatical conventions. As a result, a representative example is as follows:

```
"among consigned penally result perverseness checked stated held sensation reasonings skies
adversity Dakota lip Suffer approached enact displacing feast Canst pearl doing alms comprehendeth
nought"
```

Not a comma, semicolon, period, or (my favorite) (http://en.wikipedia.org/wiki/Dash#Em_dash)em dash (http://en.wikipedia.org/wiki/Dash#Em_dash) to be seen, but the end result is (<http://www.biblegateway.com/passage/?search=1+Corinthians+14&version=NIV>)intended to be incomprehensible (<http://www.biblegateway.com/passage/?search=1+Corinthians+14&version=NIV>). To that end, it appears to be a success.

About Terry Davis

Davis is a very open and very opinionated person. He graduated from Arizona State University with a 3.63 GPA in December 1992 and started his IT career at Ticketmaster in 1990, working on its VAX OS. His SAT score is 1440. His (<http://www.templeos.org/Wb/Accts/TS/Wb2/Rants/TAD/>)rants (<http://www.templeos.org/Wb/Accts/TS/Wb2/Rants/TAD/>) vary from what is on the BBC to politics to users on (http://wiki.osdev.org/Main_Page)OSDev (http://wiki.osdev.org/Main_Page) and more.

He has spent the last 10 years on disability, throughout which he has been working on TempleOS or a previous incarnation of it (the OS was previously known as SparrowOS or LoseThos). According to Davis, he has been diagnosed with schizophrenia, does not live independently, and does not drive a vehicle.

Final thoughts

TempleOS is a testament to the dedication and passion of one man displaying his technological prowess. It doesn't need to be anything more.

What do you think?

What's your first impression of TempleOS? Do you think you'll compile your own distro in TempleOS? Share your thoughts in the discussion.



About James Sanders

James Sanders is a Java programmer specializing in software as a service and thin client design, and virtualizing legacy programs for modern hardware.

[ADD YOUR COMMENT](#)