TempleOS: A Technical Report

By Jesse Arstein, Alexis Tingey, Benjamin Barnett and Kevin Kleiman

Introduction

- Developed by Terry A. Davis.
- 64 bit operating system.
- Major religious themes.

History

- Developed by Terry A. Davis in the early 2000s
- Brilliant but troubled engineer
- Diagnosed with schizophrenia
- TempleOS heavily influenced by his illness
- Originally named J operating system
- Commodore 64
- God’s Third Temple on earth

Technical Analysis Process

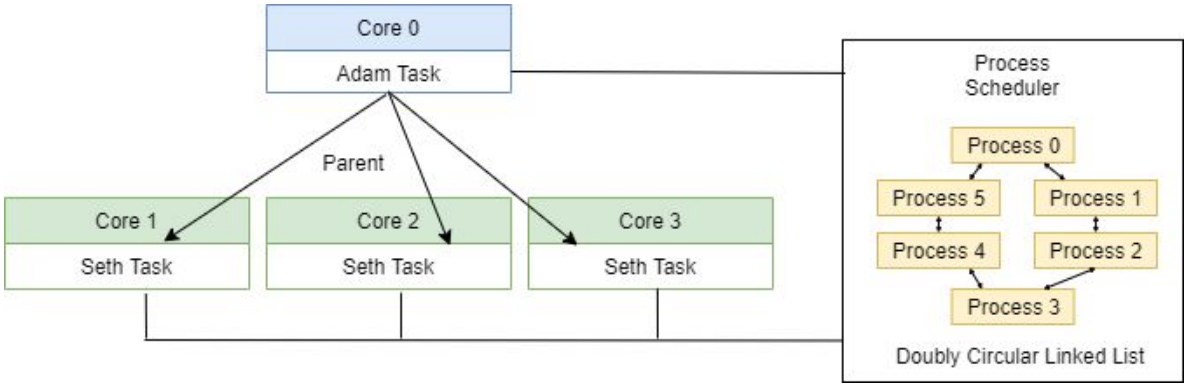
- Analysis is mainly done through a combination of reading technical documentation and from reading the source code.
- Broken into two main components
  - Processes and process scheduling
  - Memory management

Processes and Process Scheduling

- Threads don’t necessarily exist. When Forks are executed, the OS treats the child as a whole new program/process.
  - Shared resources and resource locks
- All processes follow a hierarchical parent-child relationship.
- All processes are descendants of the “Adam Task.”
- Multi core processing is handled in a master-slave design pattern.
  - Seth Tasks
    - Adam’s Son

Processes and Process Scheduling Continued

- Process scheduling is done through a circular doubly linked list.
  - Seth Task is the head of every linked list.
- Simple locks for race condition handling.
- All processes run in ring 0.
  - Complete access to entire system.



Memory Management

- Paging
  - Is simplicity always the answer?
- User v Kernel memory space
  - Is there a difference?
  - The Adam Task
- Memory allocation
  - On a need basis
  - Does TempleOS leak memory?
  - What is a user problem and what is a system problem?

Usage

Built for small local projects

- Davis built the OS for recreational programming. He compared it to a motorcycle, where windows was more like a car.
- No networking or simultaneous processes limits it to this

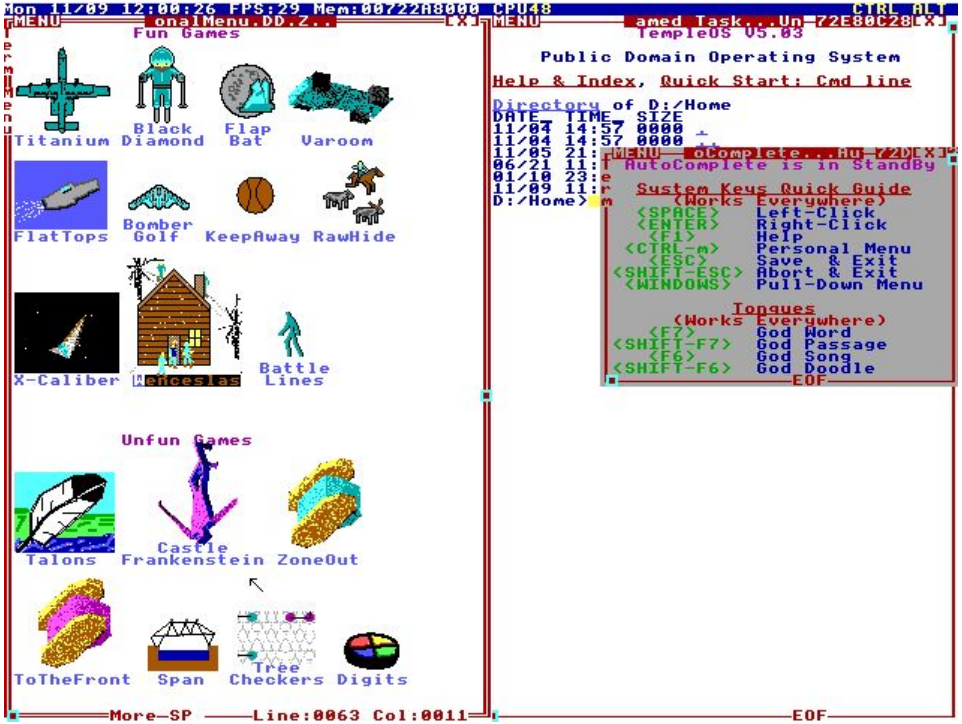
Simple to use

- Code straight from the command line
- Write in Holy C (similar to C and C++)
- Can crash if you do something wrong, but will reboot quickly

Usage Continued

Other Features

- Keyboard and mouse compatible
- Has ~30 preloaded programs (mostly games)
- Compatible with a mouse



Conclusion

- TempleOS is an impressive testament to the dedication of one individual.
- Operating systems are complex systems that require a lot of work and meticulous design choices.
- TempleOS followed the philosophy of giving power to the user.
    - Linux - Power to use without sacrificing usability and security.
    - MacOS- Less power to the user but more usability.
    - Windows - Attempts to be balanced with a focus on backwards compatibility.

Bibliography

Davis, Terry A. “TempleOS Documentation.” *tempos.holyc.xyz*, Terry A. Davis, 30 March 2018, https://templeos.holyc.xyz/Wb/Doc/. Accessed 1 November 2020.

Mitton, Richard. “A Constructive Look At TempleOS.” *Coders Notes*, Coders Notes, 8 June 2015, http://www.codersnotes.com/notes/a-constructive-look-at-templeos/. Accessed 1 November 2020.