**(IDOS) Interlocking Designed Operating System**

by C.D. Jones

In IDOS, programs are treated like puzzle pieces, interconnected to form the operating system. No need for applications, utilities, tools, or even kernels. Each program is unique, interlocking and equal. Each puzzle-piece program is considered to be an equal part of a unique manually constructed system.

Each piece has its own “key” or “service-lock” extension. Which acts analogous to the edges of a puzzle piece. Each piece is unique, each puzzle is is unique because of the swappable piece.

Applications are made up of “bundles” or bunches of documents creating that module (puzzle piece) The OS (PC) acts essentially as a small network or internet.

Part of an operating systems duties are to manage and schedule programs and memory.

It has been postulated that the next generation operating system will be in fact the browser. Sandbox? Cookies? Memory Management?

The operating system has a shell, a program to access its innards such as korn shell, c shell, bash, etc.

Imagine a world where you don’t have to program variables or functions, where the playing field is level, where everything just “fits”. A programming paradigm extremely decoupled. A programming paradigm where one puzzle piece service can pipe to another, but there is no coupling between the two. **A service, when conducting business with another service, neither of the two are dependent upon the others makeup.** It is a flat, universal playing field.

Conducting business between pieces is done through pipes. Memory management and Service scheduling are directed by the only “kernel” or “lower level” layer and that is by the use of the bootstrap service, “init”. In the case where the operating system has to step in, it is done by init.

When the system boots, init launches all the system services, simultaneously, injecting processes into them. Each service has its own process. Processes can operate with multiple threads within their own puzzle piece. The only communication of service to service is through a natural language like language sent via pipes.