Overview

This document serves to characterize the high level design of the CPU scheduler discrete event simulation (DES). A discrete event simulation is one that attempts to codify and categorize the behavior of complex systems as an ordered sequence of well-defined events. The DES system is designed with various goals in mind, prioritized in the following order:

Correctness The simulation should be correct. The mechanics of the simulation should run without error and should logically implement the system description. The statistics, counts, and other analysis-related data should be correctly stored and reported.

Simplicity The DES should be simple and intuitively designed. Explicit code should trump clever code. The design hierarchy should be reproducible and understood from a high level with ease. Code should be well documented and standardized (simple to understand).

Modularity The design of the system should be modular. Definitiions and functionalities of each moving part should be well-defined and not unexpectedly change. The DES will be objected oriented. Classes, data, and functional procedures should be as stand alone as possible; avoid strong coupling between elements of the class hierarchy and general flow of system.

Scalability The system should be scalable. Adding new moving parts should not be impossible or require significant overhaul in order to implement.

Efficiency When possible/sensible, the system should be efficient. Avoid nested looping structures or oppressively iterative computation. Vectorize when possible and carefully weigh each implementation so that speed or storage does not become an issue.

System Description

Sources

- Discrete Event Simulator Description
https://whatis.techtarget.com/definition/discrete-event-simulation-DES