



Software Redundancy

Tolulope Kupoluyi

Software Redundancy



- Redundancy in software terms differs considerably to its hardware equivalent.
- Simultaneous failure is a very likely possibility in the case of replicated software, hence the need for dissimilar software packages
- Design diversity is a key feature in software redundancy to mitigate the possibility of failure.

Approaches to Software Redundancy

- *Backup Software*: This option is used as a contingency to prevent total system shut down. It is typically associated with reduced functionality and by extension, reduced resource consumption but with key processes accounted for.
- *N-version Programming*: This option employs the use of dissimilar versions of software designed according to a common set of requirements. Expectedly it is the more costly of the two.

Hardware vs. Software Unavailability

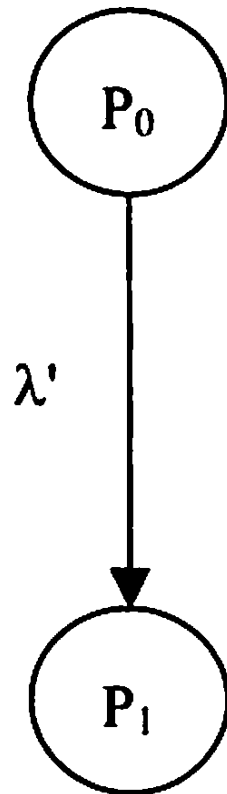
- In reliability modeling and calculations, calculations for hardware reliability can also be extended to software reliability due to equalizing factors when both are considered over a length of time.
- Software unavailability can be considered under two separate conditions: a situation where the software is restarted in which case the failure is overcome but not directly addressed and when actual repair procedures are implemented.

Detection, Isolation & Recovery



- These steps are essential in software redundancy and are the fundamental principles by which primary software and backup software are employed.
- Rollback and recovery are two common approaches of dealing with software failure.

Markov Model Analysis



$$\text{change in } P_0 = \Delta P_0 = -P_0 \times \lambda' \times \Delta t$$

$$\text{change in } P_1 = \Delta P_1 = +P_0 \times \lambda' \times \Delta t$$