CS 306 (Operating Systems) Project

Submitted by :
Pranjul Shukla (2015325) and **Animesh Pandey (2015298)**

Deadlock Resolution Techniques

Description of a deadlock

A deadlock occurs when there is a set of processes waiting for resource held by other processes in the same set. The processes in deadlock wait indefinitely for the resources and never terminate their executions and the resources they hold are not available to any other process. The occurrence of deadlocks should be controlled effectively by their detection and resolution, but may sometimes lead to a serious system failure.

Resolution Techniques

1. Resolution by using Timestamp :

In this approach, a timestamp is allocated to each process as soon as it enters the system. The timestamp of the younger process is greater than the timestamp of older process. According to this approach, the victim is selected on this timestamps, the process with the higher timestamp is aborted, that is the youngest process is selected as the victim and is aborted in order to break the deadlock cycle.

Pros :

The advantage with choosing the youngest process as victim is that the youngest process would have used less resources and less CPU time as compared to older process.

Cons :

It can cause starvation problem because every time a younger process is aborted which can starve the younger process from completion.

2. Resolution by using Burst Time :

Another approach for selecting a victim to break deadlock cycle is considering the burst time of each process. Burst time means the CPU time needed by any process for its execution. This can also be considered as one parameter for selecting a victim.

Pros :

The process with maximum burst time can be aborted in order to break cycle. By doing so the deadlock condition is avoided and one heavy process is terminated.

Cons :

The problem with this technique is that it can abort the process with high burst time which has been in the system for very long i.e. an older process with high burst time can be aborted which is inefficient approach.

3. Resolution by Degree :

In a wait-for-graph for any system, the degree of any vertex denoting a process determines how many resources a process is holding and how many resources a process is requesting. There are two types of degrees in a directed WFG:

- **In-degree**: In-degree means the number of edges coming to any node of WFG and it denotes number of request for resources held by a process.
- **Out-degree**: Out-degree means the number of edges going out of a node in WFG denoting number of request for resources done by the node.

In resolution by degree, degree of each process is calculated and process having highest degree is aborted. Degree of any process can be calculated by taking sum of in-degree and out-degree.

Pros :

Terminating the process using maximum resources results in multiple deadlocks getting resolved by terminating a single process in some cases. This prevents the deadlock condition from occurring that often in the future. This method speeds up the the deadlock resolution process.

Cons :

Detection of degree is difficult to find because it involves making a Wait For Graph (WFG) which is time consuming and complicated.

References

1. "Deadlock Resolution Techniques: An Overview" by Pooja Chahar, Surjeet Dalal, published in International Journal of Scientific and Research Publications, Volume 3, Issue 7, July 2013 1 ISSN 2250-3153

2. "Improved Deadlock Prevention Algorithms in Distributed Systems" by Mahboobeh Abdoos published in International Journal of Engineering and Applied Computer Science (IJEACS) Volume: 02, Issue: 02, February 2017

3. "Analysis for Deadlock Detection and Resolution Techniques in Distributed Database"by Swati Gupta , Meenu Vijarania published in International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 7, July 2013 ISSN: 2277 128X.