

CHAPTER 7

CONCLUSIONS AND FUTURE ENHANCEMENTS

7.1 Conclusions

This research was mainly focussed on improving the performance of virtual machines in Xen, which is a paravirtualized system. There are many factors by improving which, one can be able to improve the performance of the virtual machines in Xen server. In various kinds of environments and depending upon the various kinds of workloads that are running in the virtual machines, the performance tuning problem can be viewed from different dimensions.

This research, as the topic suggests, focusses on improving the performance of the virtual machines through improvising the original scheduling algorithm present in the xen hypervisor. Regarding the problem, the CPU scheduling problem in xen hypervisor is to schedule VCPUs of domains (Host as well as Guests) to be executed on the PCPUs. The credit scheduler, which is the default CPU scheduler in xen, is a proportional fair share algorithm that provides credit to the VCPUs according the weight and cap defined for the active domains running in the system.

For the initial assessment of the performance of Credit as well as another existing scheduler, SEDF, experiments were conducted, and it was found out that weight of the domains was a crucial factor that affects the performance of the virtual machines in xen server. The credit scheduler always runs with predefined weights of the guest domains, and allocates credits according to those predefined shares. However, it does not account for how much workload does a running virtual machine is running with.

A new function was developed, which calculated the workload of all the running virtual machines, in terms of accumulated execution time, for a given fixed time interval. This workload information was fed to the weight calculating function, which recomputes the weight for each virtual machines or guest domains. Then the original credit scheduler was run for the scheduling, this time, with the updated weight based on the workload of each active domain in the system. Since the new algorithm takes account of the workloads in each virtual machines, the machine with higher workload will get more share of PCPU, as verified by the experimental results. Thus, this new modified credit scheduler with dynamic weight update improves the performance of virtual machines in Xen virtualization system.

The modified algorithm is capable of dynamically changing the weights of the virtual machines according to their workloads. But I/O sensitive applications and network workload applications

that run in the virtual machines require higher time slice for the scheduler for better performance, rather than merely increasing the PCPU share. This algorithm, however, is limited only to the weight parameter so the performance of virtual machines with I/O sensitive workloads and network workloads do not get more optimized performance than it could have been, if timeslice could be dynamically updated according to the workload of the running VMs.

7.2 Future Enhancements

As stated earlier, there are many dimensions through which a performance improvement problem can be viewed upon. As pointed out in many existing literatures as well as noticed during the experiments and observations, it was found that the credit scheduler algorithm is fair and suitable for the conditions where virtual machines are competing for computing needs. However, in case of network input output, as well as while using audio-video applications, the credit scheduler cannot work properly, as these latency sensitive applications fail to perform efficiently, because of the lesser time slice, the credit scheduler uses which is 30ms.

The xen hypervisor 4.6.0 has the capability to increase or decrease the time slice manually, through its system commands. However, as in the case of weight, the nature of the workload for virtual machines cannot be determined initially, so the time slice period also cannot be determined. Further research can be done to find out ways that can dynamically set the time slice period so that the credit scheduler can use bigger time slices for domains running applications with latency sensitive applications.