

An analysis of Linux Scalability to Multi Cores

Aaryan Bhagat

Summary

Paper is a critique and an improvement of the current linux kernel of that time with regards to how it scales, they made a combined benchmark of several parallel executed applications of various kinds. The applications range from file indexers to memory heavy applications to applications spending most of the time in the kernel. Overall this is a diverse test!.

Authors have also developed some general techniques of multicore packet scheduling, sloppy counters, lock free comparison which they have applied as patches to the kernel. Their main goal is to check the scalability tests with the current linux kernel design and hardware limitations. After their changes in the kernel all the scalability limitations arise from either the application side or the kernel. We see a considerable difference in timings for each application except metis. It was a 48 core platform and the author suggested that making modest patches to the kernel should work instead of a whole overhaul of the kernel design which many people thought and hence this paper. Also the efficiency in hardware is another approach. They focused on specifically the linux kernel due to it being the most popular, wide usage and support the linux community provides.

Pros

- Diverse set of applications for benchmark, large number of cores, all of these factors really will give thorough testing.
- Solving various linux kernel bottlenecks and even made patches for it.
- Conclusion is important which emphasizes that current usage of linux kernel is fine and no overhaul is required.

Cons

- Should have tried on other kernels as well, then a broader comparison could be done.

Summary

I do not have much else to opine on this paper, it is very straightforward experimentation and conclusion. The results speak for themselves as well as the methods the researchers used are pretty straightforward and logically correct.