UNIVERSITI PUTRA MALAYSIA

RUNTIME PLUGGABLE CPU SCHEDULER FOR LINUX OPERATING SYSTEM

NASR ADDIN AHMED SALEM AL-MAWERI

RUNTIME PLUGGABLE CPU SCHEDULER FOR LINUX OPERATING SYSTEM

By

NASR ADDIN AHMED SALEM AL-MAWERI

Thesis Submitted to the School of Graduate Studies, Universiti Putra Malaysia, in Fulfillment of the Requirements for the Degree of Master of Science

November 2010

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfillment of the requirement for the degree of Master of Science.

RUNTIME PLUGGABLE CPU SCHEDULER FOR LINUX OPERATING SYSTEM

By

NASR ADDIN AHMED SALEM AL-MAWERI

November 2010

Chair: Khairulmizam Samsudin, PhD

Faculty: Engineering

The demand for a flexible operating system have increased with the variation of working environments i.e. interactive systems, real-time systems, batch systems, distributed systems or embedded systems. This variation of working environments has generated different applications with different user requirements.

Process scheduler, or CPU scheduler is the component of the operating system that plays the role of processes coordination and synchronization. The CPU scheduler uses different scheduling policies (algorithms) to manage such processes. Each scheduling policy has its own properties and suits specific applications or a specific environment.

In this thesis a Linux based framework, Runtime Pluggable Scheduling Policies framework, has been proposed to allow switching between the scheduling policies on the CPU scheduler during run-time. The framework purpose is to enable the operating system to use suitable scheduling policy for particular applications and user requirements.

This framework allows Linux users to plug and unplug a preferred scheduling policy available in the CPU scheduler while the system is running without the need to reboot or recompile the system.

It also provides a flexible interface for kernel developers which facilitates the evaluation and testing for their newly developed scheduling algorithms. Kernel programmers could add and remove a scheduling policy to the kernel then set the CPU scheduler to run on a specific policy at run-time as well.

This framework has been developed on top of Linux 2.6.25 kernel, and supports future kernel releases. It added more flexibility to the Linux CPU scheduler. It has been implemented in three layers, kernel, modules, and user interface layer, to simplify the framework development and maintenance as well as to maintain performance.

Testing and evaluating the framework have proved that it works properly in a reliable way. Various metrics have been measured with an accurate benchmarks. Benchmarking the framework has assured that it has not introduced a deterioration to the overall CPU scheduler performance.

The kernel build time of vanilla kernel and the modified kernel were extremely close to each other regardless of the small percentage difference in some cases which is not higher than 0.2%. CPU throughput was almost similar in the modified kernel and the vanilla. CPU utilization was 95.5%, and 95.7% in vanilla kernel and RPSP respectively. The total memory was utilized by the framework was almost negligible at 396 KB. RPSP achieves the process communication in all the cases with 3.5% better than the vanilla kernel. Executing RPSP had some impact on the performance of communication. However, the difference percentage does not exceed 1%. The latency difference was not significant, where the maximum value of difference was 1.6 ms only in few cases. This deterioration in response time does not exceed the noticeable value of 7 ms. Scalability of RPSP were close to the vanilla with a maximum difference of 5% higher than vanilla. However, this percentage difference is not noticeable since it is in a fraction of microsecond.

Abstrak tesis yang dikemukakan kepada Senat Universiti Putra Malaysia sebagai memenuhi keperluan untuk ijazah Master sains.

**RANGKA KERJA POLISI PENJADUALAN MASA JALANAN BOLEH PALAM
UNTUK SISTEM OPERASI LINUX**

Oleh

NASR ADDIN AHMED SALEM AL-MAWERI

November 2010

Pengerusi: Khairulmizam Samsudin, PhD

Fakulti: Kejuruteraan

Permintaan untuk sistem operasi yang fleksibel telah meningkat dengan variasi persekitaran tugas contohnya sistem interaktif, sistem masa nyata, sistem kelompok, sistem teragih atau sistem terbenam. Variasi dalam persekitaran tugas ini telah menjana pelbagai aplikasi dengan keperluan pengguna yang berbeza.

Penjadual proses, atau penjadual CPU ialah komponen sistem operasi yang memainkan peranan penyelarasan dan penyegerakan proses-proses ini. Penjadual CPU menggunakan polisi penjadualan (algoritma) yang berbeza untuk menguruskan proses-proses sedemikian. Setiap polisi penjadualan mempunyai ciri-ciri tersendiri dan bersesuaian dengan aplikasi atau persekitaran tertentu.

Dalam tesis ini rangka kerja berasaskan Linux, rangka kerja Polisi Penjadualan Masa Jalanan Boleh Palam (Runtime Pluggable Scheduling Policies), telah dicadangkan untuk membenarkan penukaran polisi penjadualan pada penjadual CPU sewaktu masa jalanan. Tujuan rangka kerja ini adalah bagi membolehkan sistem operasi menggunakan polisi penjadual yang sesuai untuk keperluan sesuatu aplikasi dan pengguna.

Rangka kerja ini membolehkan pengguna Linux untuk menukar polisi penjadualan pilihan yang tersedia di penjadual CPU semasa sistem itu berjalan tanpa perlu memasang semula atau mengumpul semula sistem tersebut.

Ia juga menyediakan suatu antara muka yang fleksibel untuk pembangun kernel dan bagi memudahkan penilaian dan pengujian algoritma penjadualan yang baru dibangunkan. Pengaturcara kernel boleh menambah dan membuang polisi penjadualan pada kernel dan kemudiannya sekaligus menetapkan penjadual CPU untuk menjalankan satu polisi tertentu pada masa jalanan.

Rangka kerja ini telah dibangunkan di atas kernel Linux 2.6.25, dan menyokong kernel yang akan datang. Ia telah menambahkan kelenturan kepada penjadual CPU Linux. Ia telah dilaksanakan dalam tiga lapisan, kernel, modul dan lapisan aplikasi pengguna, untuk memudahkan pembangunan dan penyenggaraan rangka kerja serta untuk mengekalkan prestasi.

Ujian dan penilaian rangka kerja telah membuktikan yang ia berfungsi dengan baik dan boleh diharap. Pengujian tanda aras rangka kerja juga telah menjamin yang ia tidak menyebabkan kemerosotan pada prestasi keseluruhan penjadual CPU.

Masa bangun kernel bagi kernel vanilla dan kernel yang telah diubahsuai hampir sama tanpa mengambil kira perbezaan peratusan yang sangat kecil dalam beberapa kes yang tidak melebihi 0.2%. Daya pemprosesan CPU hampir sama di dalam kernel yang telah diubah dan kernel vanilla. Penggunaan CPU adalah 95.5% di kernel vanilla dan 95.7% di RPSP. Jumlah memori yang digunakan oleh rangka kerja itu hampir boleh diabaikan pada 396 KB. RPSP mencapai proses komunikasi dalam semua kes dengan 3.5% lebih baik dari kernel vanila. Pelaksanaan RPSP memberikan beberapa kesan terhadap prestasi komunikasi. Walau demikian, peratusan perbezaan tidak melebihi 1%. Perbezaan penangguhan tidak ketara,dimana nilai maksimum perbezaan adalah 1.6 ms hanya dalam beberapa kes. Kemerosotan dalam waktu respon ini tidak melebihi nilai ketara pada 7 ms. Kebolehskalaan RPSP hampir sama kepada vanila dengan perbezaan

maksimum pada 5% lebih tinggi daripada vanila. Bagaimanapun, perbezaan peratusan ini tidak ketara kerana ia sebahagian dari mikrosaat.

ACKNOWLEDGEMENTS

In the name of Allah the Most Beneficent the Most Merciful. It is with the deepest sense of gratitude to Allah who has given me the strength, patience and the ability to achieve this thesis as it is today.

I would like to express my deepest appreciation and gratitude to my supervisor Dr. Khairulmizam Samsudin for his continuous spirit of help and assistance. Without his guidance and persist help this thesis would not have been possible.

I would like to thank my co-supervisor Dr. Fakhrul Zaman Rokhani for his valuable help and recommendations.

I would like to thank my fantastic family who tolerated my absence for this long. Thanks to my great parents who always pray for my success in this life. Thanks for my brothers and sisters for their encouragements.

Finally, many thanks to all my friends for their help, advices and emotionally support.

I certify that a Thesis Examination Committee has met on **8th November 2010** to conduct the final examination of Nasr addin Ahmed Salem Al-Maweri on his thesis entitled "**Runtime Pluggable CPU Scheduler for Linux Operating System**" in accordance with the Universities and University Colleges Act 1971 and the Constitution of the Universiti Putra Malaysia [P.U.(A) 106] 15 March 1998. The Committee recommends that the student be awarded the Master of Science.

Members of the Thesis Examination Committee were as follows:

M. Iqbal Bin Saripan, PhD
Lecturer
Faculty of Engineering
Universiti Putra Malaysia
(Chairman)

Abdul Rahman b. Ramli, PhD
Associate Professor
Faculty of Engineering
Universiti Putra Malaysia
(Internal Examiner)

Nur Izura Udzir, PhD
Lecturer
Faculty of Computer Science and Information Technology
Universiti Putra Malaysia
(Internal Examiner)

Mohamed Khalil Hani, PhD
Professor
Faculty of Graduate Studies
Universiti Teknologi Malaysia
Malaysia
(External Examiner)

_____

BUJANG KIM HUAT, PhD
Professor and Deputy Dean
School of Graduate Studies
Universiti Putra Malaysia

Date:

This thesis was submitted to the Senate of Universiti Putra Malaysia and has been accepted as fulfillment of the requirement for the **Master of Science**. The members of the Supervisory Committee were as follows:

Khairulmizam Samsudin, PhD
Lecturer
Faculty of Engineering
Universiti Putra Malaysia
(Chairman)

Fakhrul Zaman Rokhani, PhD
Lecturer
Faculty of Engineering
Universiti Putra Malaysia
(Member)

_____

HASANAH MOHD GHAZALI, PhD
Professor and Dean
School of Graduate Studies
Universiti Putra Malaysia

Date:

DECLARATION

I declare that the thesis is my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously, and is not concurrently, submitted for any other degree at Universiti Putra Malaysia or at any other institutions.

_____

NASR ADDIN AHMED SALEM AL-MAWERI

Date: 08/November/2010

TABLE OF CONTENTS