View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by UMP Institutional Repository

A MODEL FO POWER EFFICIENCY OF MOBILE DEVICES THROUGH LIGHTWEIGHT METHOD LEVEL COMPUTATIONAL OFFLOADING

MUSHTAQ ALI

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Doctor of Philosophy

Faculty of Computer Systems & Software Engineering

UNIVERSITI MALAYSIA PAHANG

FEBRUARY 2018

ABSTRAK

Peranti mudah alih telah menjadi satu bahagian penting dalam kehidupan seharian kita. Walaubagaimanapun, masa yang terhad pada bateri mengurangkan masa operasinya. Untuk menangani masa yang terhad pada bateri, pemunggahan pengiraan (computational offloading) digunakan untuk melepaskan tugas intensif daripada peranti mudah alih ke pelayan jauh bagi melaksanakan tugas itu dari jauh dan menjimatkan hayat bateri. Rangka kerja pemunggahan pengiraan berdasarkan penghijrahan Virtual Machine (VM), aplikasi keseluruhan penghijrahan atau tahap kaedah (method level) tradisional pemunggahan adalah sumber yang intensif dan memakan masa. Pembahagian yang dinamik pada aplikasi, pelaksanaan tugas pada pelayan awan, panggilan perkhidmatan oleh SOAP dan tiada penentu mekanisme untuk parameter yang telah ditetapkan, mengubahkan rangka kerja tahap kaedah pengiraan (method level computational) menjadi tidak cekap untuk penjimatan tenaga. Dalam usaha untuk menangani kekurangan rangka kerja tahap kaedah pengiraan pemunggahan (method level computational offloading), rangka kerja tahap kaedah yang ringan (lightweight method) telah dicadangkan. Empat komponen yang berbeza digunakan dalam rangka kerja yang dicadangkan bagi menghapuskan kelemahan rangka kerja yang dibangunkan sebelum ini. REST digunakan untuk panggilan perkhidmatan yang berasaskan JSON dan ia menghapuskan SOAP yang berasaskan XML. Oleh sebab itu, REST adalah satu pendekatan ringan. REST juga mengurangkan saiz data komunikasi sehingga 100% berbanding dengan panggilan perkhidmatan SAOP. Pelayan tumpang (Surrogate server) dikonfigurasikan pada jarak hop tunggal yang mengurangkan RTT dan seterusnya mengurangkan penggunaan kuasa. Aplikasi itu dibahagikan di peringkat tahap kaedah (method level) yang sama dengan rangka kerja tahap kaedah sebelumnya tetapi pembahagian berlaku di peringkat kod sumber statik. Satu mekanisme khusus untuk pemilihan parameter yang telah ditetapkan adalah penting untuk dipertimbangkan sebelum setiap offload. Parameter yang telah ditetapkan terdiri daripada tahap bateri, jenis rangkaian dan masa pelaksanaan telah mengesahkan penjimatan tenaga semasa pemunggahan. Rangka kerja yang dicadangkan telah dilaksanakan dalam persekitaran pengkomputeran awan mudah alih yang sebenar. Masa Pelaksanaan dan Penggunaan Tenaga oleh Pelaksanaan Tempatan dan Pemunggahan Secara Tradisional ditanda aras untuk menyiasat dan mengesahkan pelaksanaan rangka kerja tahap kaedah ringan (lightweight method level) yang dicadangkan. Prototaip dibangunkan dengan tiga komponen REST-Offload, Pelaksanaan Tempatan dan Traditional-Offload serta ia diuji dalam persekitaran awan mudah alih sebenar untuk Masa Pelaksanaan dan Penggunaan Tenaga. Hasilnya telah menunjukkan bahawa penyelesaian yang dicadangkan telah mengurangkan penggunaan sumber pada peranti mudah alih. REST-Offload adalah amat berguna jika dibandingkan dengan kedua-dua Pelaksanaan Tempatan dan kaedah Pemunggahan Tradisional. Ia mengurangkan kira-kira 50% Masa Pelaksanaan dan kira-kira 38% Penggunaan Tenaga.

ABSTRACT

Mobile devices have become an integral part of our daily lives. However, the restricted battery timing curtails longer operational hours. To tackle the limited battery timing issue, a technique, computational offloading is used. In computational offloading, the intensive tasks are offloaded from mobile devices to remote server in order to execute the task remotely and save battery life. Computational offloading frameworks/models based on VM migration, whole application migration, or traditional method level offloading are resources intensive and time consuming. The dynamic partitioning of application, execution of task at cloud server, service call by Simple Object Access Protocol (SOAP) and no defined mechanism for predefined parameters, make the previous method level computational frameworks/models inefficient for energy saving. In order to address the inefficiencies of previous method level computational offloading frameworks/models, a lightweight method level computational offloading model is proposed. Four distinct components are deployed in the proposed model which eliminates the shortcomings of previously developed frameworks/models. A Representational State Transfer (REST) based technique developed for calling the remote services which is based on JSON instead of XML, and hence is lightweight. REST also reduces the size of communication data at approximately 100% as compared to SAOP service call. Surrogate server is configured at a single hop distance which reduces the RTT and ultimately reduces the power consumption. The application is partitioned at method level by a novel dynamic technique in source code, which counters the inefficiencies of existing partitioning techniques. A mechanism for selection of predefined parameters is defined. These parameters are important to consider before each offload. The predefined parameters consist of battery level, network type, and execution time which affirms the energy saving during offloading. The proposed framework is implemented in the real mobile cloud computing environment. Execution time and energy consumption of both local execution and traditional offloading are benchmarked in order to investigate and validate the performance of the proposed lightweight method level model. The prototype is developed with three components which are REST-Offload, Local Execution and Traditional-Offload and then tested in real mobile cloud environment for Execution Time and Energy Consumption. The result of this research indicates that the proposed solution diminishes resources utilization. The REST-Offload is significantly useful compared to both Local Execution and Traditional Offloading methods. It reduces about 50% Execution Time and approximately 38% Energy Consumption.