

## ОСОБЛИВОСТІ І ВІДМІННОСТІ РОБОТИ СУЧАСНИХ ВІРТУАЛЬНИХ МАШИН ДЛЯ ПЛАТФОРМИ ANDROID

Гречмак Д.В.

Науковий керівник – к.т.н., ст. викл. Ткачов В.М.

Харківський національний університет радіоелектроніки  
(61166, Харків, просп. Науки, 14, каф. ЕОМ, тел. (057) 702-13-54)

e-mail: [dec@nure.ua](mailto:dec@nure.ua)

In this article, I'll explore the differences between the Java Virtual Machine (JVM) and the Dalvik Virtual Machine (DVM). Both VM's run applications written in Java, but they use different techniques and processes to compile and run code. Conceptually, there is little difference from an application level between a DVM and a JVM. Architecturally, there is a major difference between the registerbased DVM and the stack-based JVM.

Java Virtual Machine - віртуальна машина, здатна виконувати байт-код Java незалежно від базової платформи. Вона спирається на принцип "Write once, run anywhere". Байт-код Java може бути запущений на будь-якій машині, здатній підтримувати JVM. Компілятор Java перетворює .java файли в class-файли (байт-код). Байт-код передається JVM, який компілює його в машинний код для виконання безпосередньо на CPU.

Особливості JVM це: стекова архітектура: в якості структури даних, куди поміщаються і зберігаються методи, використовується стек. Він працює за схемою LIFO або "Last in - First Out" або "Останнім увійшов, першим вийшов"; Може запускати тільки class-файли.; Використовує JIT-компілятор. Dalvik Virtual Machine (DVM) - віртуальна Java машина, розроблена, як частина мобільної платформи Android. Можна сказати, що Dalvik - це середовище для виконання компонентів операційної системи Android і призначених для користувача додатків. Кожен процес виконується в своєму, ізольованому адресному просторі. Коли користувач запускає додаток (або операційна система запускає один зі своїх компонентів), ядро віртуальної машини Dalvik (Zygote Dalvik VM) створює окремий, захищений процес в загальній пам'яті, в якому безпосередньо розгортається VM, як середовище для запуску програми. Іншими словами, зсередини Android виглядає як набір віртуальних машин Dalvik, в кожній з яких виконується додаток. Особливості DVM це: використання архітектури на основі реєстрів: структура даних, куди поміщаються методи, заснована на реєстрах процесора. За рахунок відсутності операцій POP і PUSH, команди в реєстрової віртуальній машині виконуються швидше аналогічних команд стекової віртуальної машини.; Виконує байт-код власного формату: Android dexer, перетворює class-файли в формат .dex, оптимізовані для

виконання на Dalvik VM. На відміну від class-файлу, dex-файл містить відразу кілька класів.

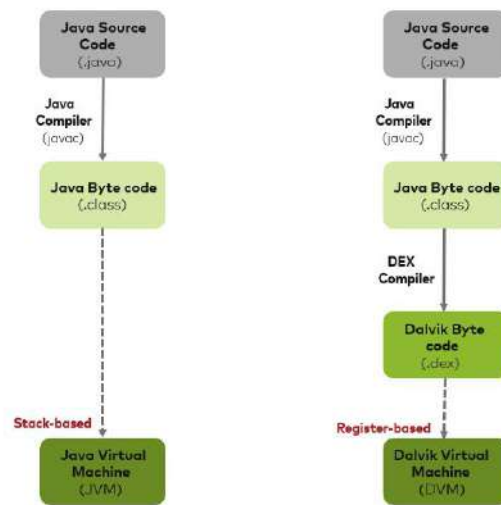


Рисунок 1 – Порівняння структури виконання процесів у JVM та DVM

Різниця в рівні додатків між DVM і JVM незначна. Архітектурно, існує велика різниця між DVM з регістром і JVM на основі стека.

Обидва використовують модель коду VM. Проте, DVM використовує реєстрові коди операцій, які можна порівняти з інструкціями байт-коду на основі регістрів, які вже виконують більшість цільових платформ. Сюди входять такі архітектури, як ARM і MIPS, а також x86-сумісні архітектури від Intel, AMD і VIA Technologies.

Список використаної літератури:

1. Tkachov, V., Bondarenko, M., Ulyanov, O., & Reznichenko, O. (2019, December). Overlay Network Infrastructure for Remote Control of Radio Astronomy Observatory. In 2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT) (pp. 161-165).
2. Tkachov, V., Hunko, M., Volotka, V.: Scenarios for Implementation of Nested Virtualization Technology in Task of Improving Cloud Firewall Fault Tolerance. In 2019 IEEE International Scientific-Practical Conference Problems of Infocommunications, Science and Technology (PIC S&T), pp. 759-763. IEEE (2019).
3. Hunko M.A., Tkachov V.M. Development of a module for sorting the ipaddresses of user nodes in cloud firewall protection of web resources. Дев'ята міжнародна науково-технічна конференція «Сучасні напрями розвитку інформаційнокомунікаційних технологій та засобів управління». 2019. С. 30.