

Матеріали XX наукової конференції ТНТУ ім. І. Пулюя, 2017

УДК 004.43

А.М. Луцків, канд.техн.наук, доц., М.Я. Чайковський

Тернопільський національний технічний університет імені Івана Пулюя, Україна

ОСОБЛИВОСТІ ТА ПРОБЛЕМИ МІГРАЦІЇ НА JAVA 9

A.M. Lutskiv, Ph.D., Assoc. Prof., M.Y. Chaikovskyy

FEATURES AND PROBLEMS OF MIGRATION TO JAVA 9

При виконанні науково-дослідних робіт та забезпеченні навчального процесу доволі часто використовується технологія Java. Це зумовлено наявністю великої кількості відкритих та безкоштовних бібліотек, засобів розроблення, готових програмних засобів й багатьма функціональними можливостями, які надає мова. Компанією Oracle, яка є правовласником даної технології анонсовано версію Java 9[1] у липні 2017 року. Загалом, Java 9 є сумісною з попередніми версіями, зокрема, 5, 6, 7 та 8. Водночас, серед понад 90 нововведень є низка таких, які можуть суттєво вплинути на процес розроблення та супроводу програмного забезпечення. Важливо зазначити, що процес випуску версій Java визначається JCP — формалізованим механізмом, який дає змогу впливати зацікавленим особам та організаціям на розробку специфікацій технології. У даному процесі приймають участь комерційні компанії постачальники Java-вирішень (Oracle, IBM, HP, Fujitsu, Red Hat та інші), комерційні компанії споживачі Java-вирішень (зокрема, Goldman Sachs, Credit Suisse та TOTVS), спільноти інженерів-розробників та окремі особи. У даному аспекті компанії Red Hat та IBM виступили проти одного з ключових нововведень[2] запропонованих компанією Oracle (JSR 376) й можуть призупинити випуск дев'ятої версії. Проаналізуємо ключові фактори, які необхідно враховувати при переході на Java 9.

Основна категорія осіб, яких стосуватиметься міграція, це: розробники програмного забезпечення, які створюють нове та забезпечують підтримку наявного програмного забезпечення, системні адміністратори, DevOps. Очевидно, що для кінцевих користувачів Java-вирішень процес міграції буде відносно простим.

На сьогодні підтримка Java 9 забезпечена в практично всіх останніх версіях найпоширеніших IDE: Eclipse (4.7 Oxygene, BETA), Netbeans (v9), IntelliJ Idea (2017.1), а також у найновіших версіях систем автоматичного збирання maven та gradle. Пропоновані особливості Java 9 спрощують процес написання коду програм шляхом введення вдосконалених синтаксичних конструкцій (наприклад, вдосконалений оператор try для роботи з AutoCloseable-ресурсами) та низки очікуваних розробниками конструкцій (наприклад, приватних методів в інтерфейсах).

Водночас є низка нововведень, які розширюють функціональність мови шляхом надання їй функціональних можливостей схожих до мови Scala, фреймворків Akka та Play (нові методи інтерфейсу java.util.Stream: dropWhile та takeWhile; підтримка реактивного програмування при роботі з потоками (Reactive Streams). Також в новій версії є розширені можливості для роботи з процесами виконання операційної системи.

Корисним для розробника є створення оболонки Java (REPL - Read Evaluate Print Loop), яка дає змогу відносно швидко тестувати конструкції мови (класи, інтерфейси, переліки, об'єкти, оголошення тощо). Розширено можливості бібліотек мови та компонентів шляхом забезпечення підтримки клієнтського API для протоколу HTTP2.0, підтримки Unicode 8.0, хеш-функції SHA-3 та цілої низки інших особливостей[4]. Наведені нововведення не порушують сумісності з попередніми версіями Java-програм. Деякі утиліти JDK було видалено (JavaDB, JVisualVm, jhat) з метою зменшення розміру. Відповідні компоненти є доступними з інших ресурсів й можуть бути, за потреби, завантажені з відповідних web-ресурсів розробником.

Водночас, на думку деяких учасників JCP[2] недоцільним є впровадження модульної системи в даній реалізації. Модульна система покликана забезпечити наступні принципи об'єктно-орієнтованого проектування програмних систем:

- строго інкапсуляцію (Strong Encapsulation): обмеження доступу до внутрішнього API JDK або інших компонентів;
- низьку степінь зв'язаності між компонентами (Less Coupling);
- надійну конфігурацію (reliable configuration);
- дотримання принципу єдиної відповідальності (Single Responsibility Principle).

Забезпечення декларованих принципів реалізовано шляхом поділу компонентів платформи Java (модульність JDK та JRE) на окремі модулі, які є відносно незалежними та невеликими. Формуючи дистрибутив програмного забезпечення для користувача, розробник може обирати виключно необхідні для виконання модулі платформи Java. Водночас створюючи програмне забезпечення розробник також створює модулі своєї програми (модульність на рівні вихідного коду). Очевидно, що такий підхід дасть змогу впроваджувати платформу Java в ширше коло пристроїв, зокрема апаратних платформ Internet Of Things. Якщо в Java 8 та попередніх версіях компоненти Java розподілені по різних пакетах і є ціла ієрархія пакетів, то в Java 9 з'являється контейнер вищого рівня — модуль. Модуль може в собі містити пакети, а також інші необхідні ресурси (платформозалежний код, файли локалізації, зображення, конфігураційні файли тощо).

Нова модульна система обов'язково має враховуватись розробниками програмного забезпечення, оскільки, накладає низку правил формування власних програмних модулів й використання наявних, а також передбачає використання додаткових утиліт для лінкування, перевірки сумісності, роботи з jar-файлами. Використання нової модульної системи зумовило переформатування структури ключових jar-контейнерів Java-бібліотек JDK й забезпечило їх більшу компактність та незалежність.

Зміна архітектурних особливостей платформи Java може вплинути перш за все на Java EE і мати наслідки на етапі супроводу програмного забезпечення web-сервісів, яке працює на серверах програм[5], зокрема WildFly, JBoss EAP, IBM WebSphere Application Server та низці інших. Тому впровадження модульності Java 9 передбачає здійснення аудиту сумісності та рефакторингу наявних програмних систем, щоб уникнути небажаних наслідків на серверних платформах. Зазначимо, що один із найпопулярніших фреймворків Spring декларував підтримку Java 9.

Література

1. Java Platform, Standard Edition Oracle JDK 9 Migration Guide / Oracle Help Center // [Електронний ресурс] Режим доступу: URL: <https://docs.oracle.com/javase/9/migrate/toc.htm#JSMIG-GUID-7744EF96-5899-4FB2-B34E-86D49B2E89B6>
2. Krill P. Red Hat and IBM raise objections to Java 9 modularization / Paul Krill // InfoWorld, IDG Communications, Inc. [Електронний ресурс] Режим доступу: URL: <http://www.infoworld.com/article/3193785/java/red-hat-and-ibm-raise-objections-to-java-9-modularization.html>
3. Posa R. Java SE 9: Introduction to Module System / Rambabu Posa // Journal Dev [Електронний ресурс] Режим доступу: URL: <http://www.journaldev.com/13106/javase9-module-system-part1>
4. JDK 9 Features [Електронний ресурс] Режим доступу: URL: <http://openjdk.java.net/projects/jdk9/>
5. Greene J. Jigsaw's Missing Pieces / Jason Greene // [Електронний ресурс] Режим доступу: URL: <http://wildfly.org/news/2016/12/12/Jigsaws-Missing-Pieces/>