

05β Η ιδεατή μηχανή της Java

Τεχνολογία Λογισμικού

Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών
Εθνικό Μετσόβιο Πολυτεχνείο

Χειμερινό εξάμηνο 2017-18

Δρ. Κώστας Σαΐδης (saiko@di.uoa.gr)

Περιεχόμενα

- Η πλατφόρμα της Java (γενικά)
- Εισαγωγή στο Java Virtual Machine (JVM)

Η πλατφόρμα της Java

- Java Runtime Environment (JRE)
 - Java virtual machine
 - Java class library
- Java Development Kit (JDK)
 - JRE + Εργαλεία ανάπτυξης κώδικα σε γλώσσα Java
 - javac - java compiler
 - javadoc - παραγωγή HTML τεκμηρίωσης από τον πηγαίο κώδικα
 - jdb - debugger
 - κ.ά.

Java Class Library - Τα βασικά APIs

- I/O & Networking (Files, Sockets, κ.ά)
- Threads
- Collections & Streams (v8)
- Security & Encryption
- SQL Databases
- XML
- GUI (swing)

Ανάπτυξη της Java

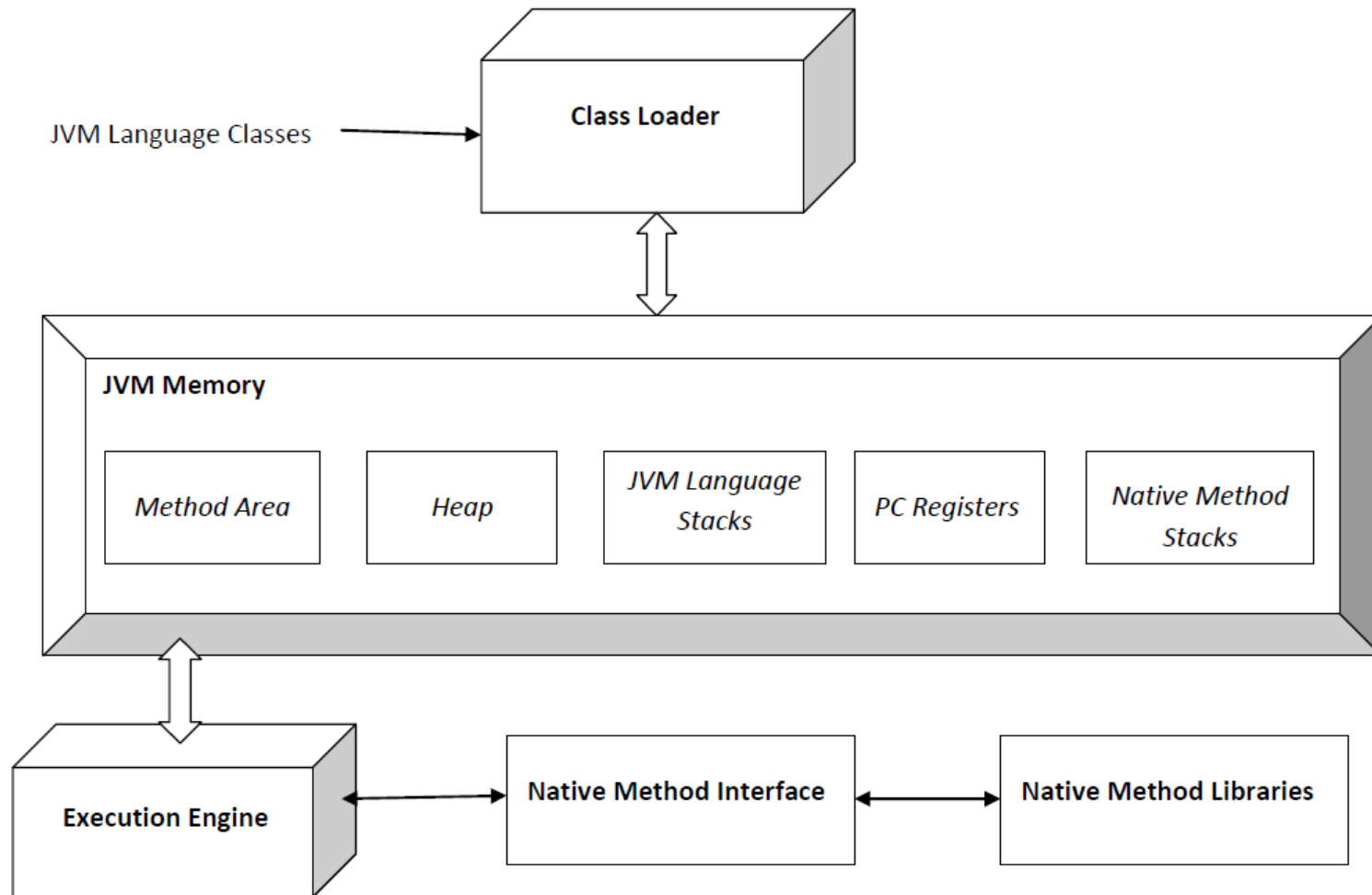
- Java Community Process (JCP)
- Java Specification Requests (JSRs)
- Reference implementations of APIs
- Open JDK
 - GNU GPL v2
 - Reference implementation of Java (JVM + Tools + Class Library)

Σύντομη εισαγωγή στο JVM

Βασικά χαρακτηριστικά

- Διαλειτουργικότητα - Portability (Write once, run everywhere)
- Αυτόματη διαχείριση μνήμης (Garbage collection)
- Υποστήριξη για πολλαπλά νήματα (threads)
- Just-In-Time Μεταγλώττιση
- Γρήγορο, στιβαρό, αξιόπιστο και διεθνώς διαδεδομένο περιβάλλον εκτέλεσης (σε servers, desktops, κινητά, άλλες συσκευές)

Η δομή του JVM



By Michelle Ridomi - Own work, CC BY-SA 4.0,

<https://commons.wikimedia.org/w/index.php?curid=35963523>

Η δομή του JVM

- Method area: αποθήκευση των κλάσεων και των σχετικών τους πληροφοριών (μέθοδοι, πεδία, υπερ-κλάσεις, κτλ.).
- Heap area: αποθήκευση όλων των αντικειμένων / στιγμιοτύπων.
- Language stacks: κάθε thread έχει τη δική του στοίβα, όπου αποθηκεύονται τοπικές μεταβλητές, ενδιάμεσα αποτελέσματα, κτλ σε μορφή stack frames.
- PC registers: Program counter register που περιέχει τη διεύθυνση του υπό εκτέλεση bytecode instruction.
- Native method stacks: αποθηκεύονται όλες οι native μέθοδοι της εφαρμογής.

Class Loader

- Μηχανισμός που επιτρέπει την παραμετροποιήσιμη προγραμματιστικά φόρτωση των κλάσεων στο JVM (.class files, bytecode)
- Οι class loaders είναι "κανονικά" Java αντικείμενα (τύπου `java.lang.ClassLoader`)
- Μια εφαρμογή Java μπορεί να έχει πολλαπλούς class loaders

Ορισμός κλάσεων

java.lang.Class

- Κατά την εκτέλεση του προγράμματος, οι κλάσεις αναπαρίστανται από "κανονικά" Java αντικείμενα (reflection)

Προσοχή

- Για την ίδια κλάση, όταν αυτή φορτώνεται από διαφορετικούς class loaders, παράγονται διαφορετικά Class αντικείμενα.

Βήματα

1. Χρήση του class loader για φόρτωση της κλάσης από το δίσκο, το δίκτυο ή από κάπου αλλού (π.χ. δυναμική επιτόπου παραγωγή της, on-the-fly)
2. Σύνδεση της κλάσης (linking)
 - Επιβεβαίωση bytecode
 - Εντοπισμός και αρχικοποίηση εξαρτήσεων
3. Αρχικοποίηση της κλάσης
 - `static` πεδία, blocks και μέθοδοι

Αρχικοποίηση κλάσεων

- Οκνηρή / lazy (ακριβώς πριν την πρώτη χρήση τους).
- Που είναι πότε;
 - Πριν στιγμιοτυπυπιδεί ένα αντικείμενο της κλάσης.
 - Πριν κληθεί μια `static` μέθοδος της κλάσης.
 - Πριν προσπελαστεί ένα `static` πεδίο της κλάσης (είτε για ανάγνωση, είτε για εγγραφή της τιμής του)
- Η αρχικοποίηση μιας κλάσης προϋποθέτει την αρχικοποίηση της υπερ-κλάσης της (super class).
- Η αρχικοποίηση μιας κλάσης γίνεται μόνο μία φορά ανά class loader.

Διάφανη χρήση των class loaders από το JVM

- Το JVM διαθέτει έναν Bootstrap class loader για τη φόρτωση των class libraries.
- Το JVM χρησιμοποιεί έναν γενικής χρήσης class loader για την φόρτωση των κλάσεων της εφαρμογής (συνήθως τύπου `java.net.URLClassLoader`).

Δεν είναι σύνηθες να απαιτείται η συγγραφή class loader από τον προγραμματιστή.

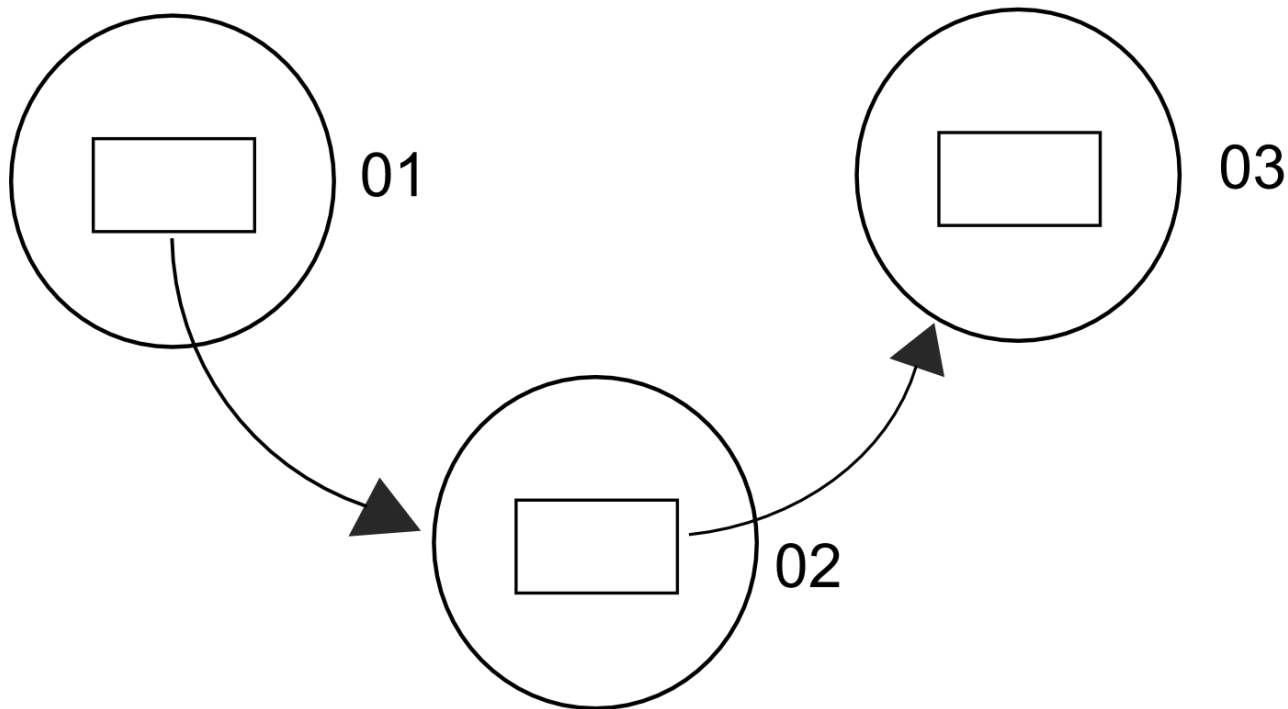
Η μηχανή εκτέλεσης του JVM

Εκτέλεση του bytecode.

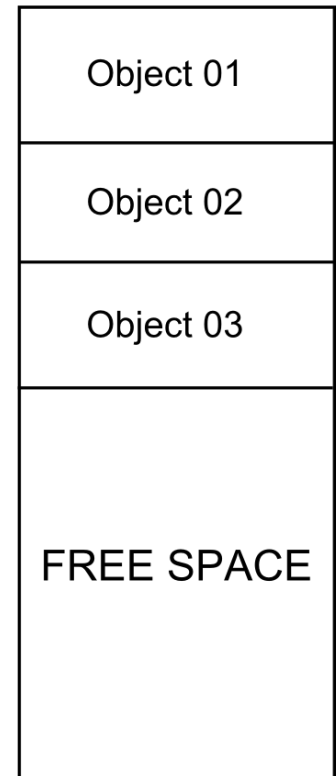
- Διερμηνέας / interpreter. Εκτέλεση γραμμή - γραμμή.
- Just-in-time μεταγλωττιστής / compiler. Μεταγλώττιση του bytecode σε native κώδικα ανάλογα με τη χρήση. Δραστική βελτίωση στην απόδοση.
- Garbage collector. Εκτελείται στο παρασκήνιο και απελευθερώνει/διαχειρίζεται τη μνήμη της εφαρμογής (heap).

To JVM Heap

Όλα τα αντικείμενα αποθηκεύονται στο Heap.



Heap



Όλα μα όλα τα αντικείμενα

- `java.lang.OutOfMemoryError` (συνήθως λόγω μη διαθέσιμου ελεύθερου χώρου στο Heap)
- Ειδικές παράμετροι στην εκκίνηση του JVM
 - Ελάχιστο heap size: `-Xms512M`
 - Μέγιστο heap size: `-Xmx2048M`

Το JVM διαχειρίζεται το Heap

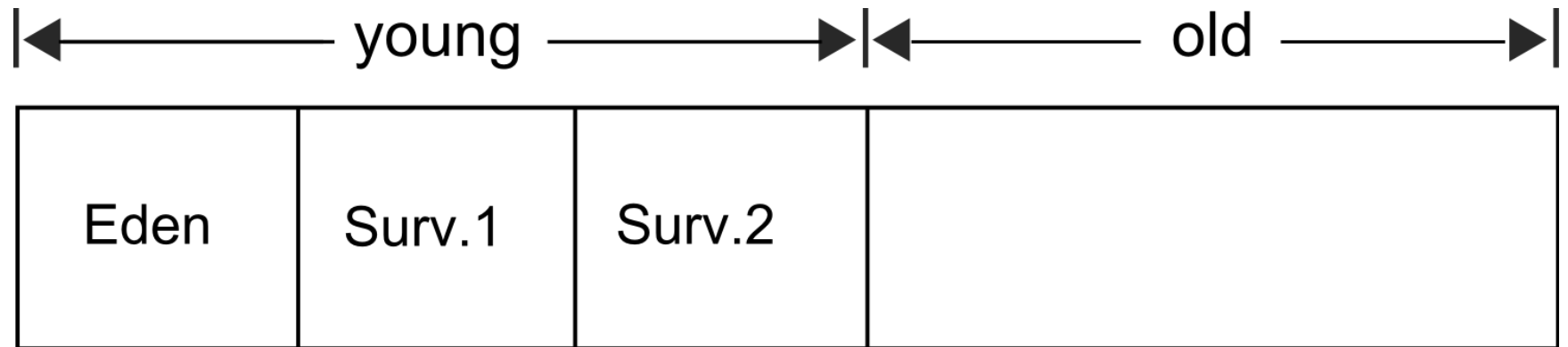
- Με διάφανο για τον προγραμματιστή τρόπο.
- Αυτόματη δέσμευση και αποδέσμευση μνήμης (~~pointers~~).
- Αυτόματοι έλεγχοι εύρους (~~buffer overflows & underflows~~).
- Garbage collection: αυτόματη απελευθέρωση μνήμης των αντικειμένων που δε χρησιμοποιούνται πλέον από το πρόγραμμα.

Garbage collection

- Θεμελιώδης έννοια: reachability of references/objects.
- Διάφοροι αλγόριθμοι: καταμέτρησης αναφορών, σημείωσης και εκκαθάρισης, σειριακοί, παράλληλοι, με ολική παύση του προγράμματος (stop-the-world), με συμπύκνωση της θρυμματισμένης μνήμης, με έμφαση στην ταχύτητα δέσμευσης/αποδέσμευσης.

Στα σύγχρονα JVM

Generational Garbage Collector

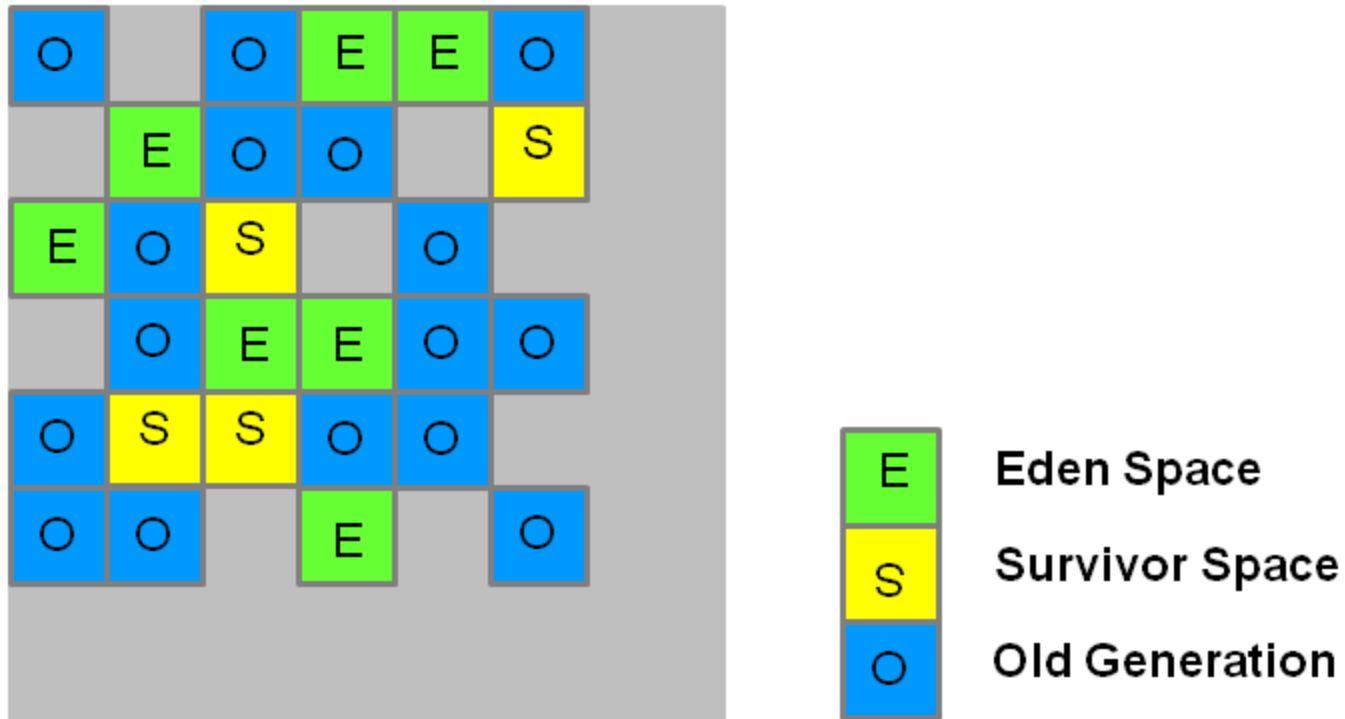


Βασική ιδέα

- Κατηγοριοποίηση των αντικειμένων με βάση τη "γενιά" τους.
 - Τα περισσότερα αντικείμενα "πεθαίνουν νέα".
 - Κάποια, λίγα, αντικείμενα "ζουν" για πολύ μεγάλα διαστήματα.
 - Σπάνια τα "παλιά" αντικείμενα αναφέρονται σε "νέα".
- Μετακίνησή των αντικειμένων από τις νεότερες στις παλαιότερες γενιές ανάλογα με το πόσες "συλλογές/εκτελέσεις" του GC έχουν καταφέρει να επιβιώσουν.
- Ταχύτερη συλλογή χώρου από τις νέες γενιές: μικρότερες παύσεις.

Garbage-First Garbage Collector (Java 9)

G1 Heap Allocation



Εκτέλεση του JVM

```
$ java -cp CLASSPATH package.of.MainClass  
$ java -jar app.jar
```

- Classpath: λίστα από τοποθεσίες που περιέχουν .class αρχεία (συνήθως φάκελοι ή jar αρχεία)
- MainClass: το πλήρες όνομα της κλάσης που περιέχει την main μέθοδο
- Jar file: αρχείο zip με κλάσεις (.class) και λοιπά στοιχεία λειτουργίας (manifests, resources, κ.ά)

Έννοιες

Language abstractions

- Classes, Interfaces, Types
- Packages (= namespaces of the above)
- Modules (Java 9, = strongly encapsulated namespaces of packages)

Application abstractions

- Components (logical)
- Layers (logical)
- Tiers (physical)
- Subsystems (logical or physical)

Java packages και δομή καταλόγων

```
package com.foo.bar;  
class Main {  
    ...  
}
```

- src
 - com
 - foo
 - bar
 - Main.java

Θα επανέλθουμε σε επόμενη διάλεξη

Java 9 Modules

```
module com.foo.bar {  
    requires com.foo.baz;  
    exports com.foo.bar.alpha;  
    exports com.foo.bar.beta;  
}
```

Δεν θα μας απασχολήσουν στο μάθημα