

Method Execution Report

Date: 06/04/2017

Type: FN / NF

System: evolve

Session overall:

This session aims at gathering impression of the *Method Execution Report* from practitioners. The session is relatively short, only a few minutes long. The session is divided into two phases. The first phase is about yourself, and the other phase is about the evaluation of the report. Note that all your answers are treated anonymously.

Description:

Method Execution Report is a textual and interactive report that summarizes the execution of a particular method for a given software execution. The report provides an overview of the dynamic calls and time consumption.

Code Execution Report
File:///C:/Documents/Publications/2017/kepi17-execution_reports/report_samples/computeCentroids/computeCentroids.html

KMeans.computeCentroids

Summary Method Calls Time Consumption

Summary

Method computeCentroids was called total 4 times. It made total 144 calls to 3 methods. It took 0.28 ms.

Method Calls

Method computeCentroids was called 4 times (100%) by only 1 method (main).

Method computeCentroids made 144 calls (100%) to 3 methods. It made maximum 64 calls (44.4%) each to methods getY and getX.

Time Consumption

Method computeCentroids took 0.28 ms (100%), out of which 0.19 ms (67.9%) were consumed as self time and 0.09 ms (32.1%) by outgoing calls made to 3 methods. Outgoing calls made to method getX took maximum time which was 0.05 ms (17.9%). Please note the measurements are uncertain due to short runtime of method computeCentroids and its outgoing calls.

```
/**
 * Berechnet neue Mittelpunkte anhand der vorgegebenen Cluster
 *
 * @param clusters
 *       Zugehörigkeit der Punkte zu den Clustern
 * @param k
 *       Anzahl der Cluster
 * @return Neue Mittelpunkte der Cluster
 */
private static Point2D[] computeCentroids(int[] clusters, int k) {
    Point2D[] centroids = new Point2D[k];
    for (int j = 0; j < centroids.length; j++) {
        double sumX = 0;
        double sumY = 0;
        int count = 0;
        for (int i = 0; i < clusters.length; i++) {
            if (clusters[i] == j) {
                sumX += points[i].getX();
                sumY += points[i].getY();
                count++;
            }
        }
        centroids[j] = new Point2D.Double(sumX / count, sumY / count);
    }
    return centroids;
}
```

Figure on the left gives an example of the report. The report is structured into three sections, and lists the code of the method.

The summary gives first essential data, including the number of incoming and outgoing calls, and how long the summed executions of the method took.

The second section details the description of the method calls, including the most important caller and callees.

The third part provides further information about the timing.

Phase 1: About yourself

- Are you Female or Male? ~~Female~~ Male

- How many years of experience do you have in programming? Since 2004 (12 years)
- How long have you been programming in Java for? NOT MUCH - ONLY SCHOOL PROJECTS
- Which Java programming environments (IDE) are you familiar with?

ECLIPSE

- While programming, if your application does not behave as you expect, what do you usually do?
How do you usually debug an application?

I usually explore running environment,
debugging, ~~exp~~ inspecting variables.

→ in Java I put a breakpoint and step in/over/out,
also using expression and variables tab.

Sometimes I evaluate expressions in code

- How do you usually do to improve the performance of a particular method?

I have zero experience in Java about performance^{nce}.

I usually use profiler and writes scripts (benchmarks)
to ~~isolate~~ isolate a problem.

But I do not profile regularly. ~~it's~~

It is not a regular issue I need to solve.

Phase 2:

TIME BEGIN: 10:30

QUESTION 1:

What do you think about the content of the textual description?

- I find it easy to understand? (strongly agree, agree, ~~neutral~~, disagree, strongly disagree)

Please, justify

yes, the example is easy to read. Only some grammar issues make it confusing: "called total 4..." → "called ~~in~~ total 4...". But I'm not expert in English.

- I find it useful? (strongly agree, agree, neutral, disagree, strongly disagree)

Please, justify

I have doubts if it is going to be useful in a real scenario. The example shows an isolated method. Maybe the useful information could be aggregated to a standard tree-call-graph-viewer.

QUESTION 2:

What do you think about the interaction and the visual elements offered by the report?

- I find them easy to understand? (strongly agree, agree, neutral, disagree, strongly disagree)

Please, justify

It is easy to understand what elements offer additional information. The first line should also display 304.68 instead of 304.676. → a bit confusing. I do not understand colors at all. 100% - blue for times? green for calls?

- I find them useful? (strongly agree, agree, neutral, disagree, strongly disagree)

Please, justify

I am not convinced that "out of X" additional information is ~~needed~~ important. The called method list is kind of call graph. Looks OK, but source code is missing.

QUESTION 3:

Overall, do you feel that such a report is useful?

- (strongly agree, agree, neutral, disagree, strongly disagree)

Please, justify

I think it would be more useful having the information (x times, y calls, z ms) infused in the source code. Developers need to explore (manipulate) real things (objects) and this add get another degree of information separation.

QUESTION 4:

In what scenarios and for solving which maintenance tasks would developers use Method Execution Reports?

Please, justify

- I have no idea. Without real experience
- I perceive standard profiler more usefull.
- It would be great to add missing information presented on the report, to the profiler.

QUESTION 5:

What tools would you use instead of Method Execution Reports to retrieve the same information?

Please, justify

- I do not know. I think ~~as~~ time ~~and~~ and percentage is present in Profiler. The missing information is number of calls.
- If I suspect there are too many calls in a method, I ~~can~~ could use an extra variable or watcher.

QUESTION 6:

Do you have any suggestion on how to improve the report? Any critic?

Please, justify

well, what I have written before: I have an impression this ~~add~~ add a separation ~~from~~ that is not good. ~~for~~ some simple improvements could be highlighting method calls in the source code. The popup with method list is even more separated from the code. It should be ~~not~~ integrated in ^{the} code. Developers need to know where and how it is call, e.g., for-cycle, after if, --

TIME END:

11:01