

DAFTAR PUSTAKA

- Abhaay, S., Shashank, M., Shreyas, B., & Kalambur, S. (2021). Effect of Garbage Collection on Streaming Big Data Workloads. *2021 2nd Global Conference for Advancement in Technology (GCAT)*, 1–6. <https://doi.org/10.1109/GCAT52182.2021.9587808>
- About the Java Technology (The Java™ Tutorials > Getting Started > The Java Technology Phenomenon). (n.d.). Retrieved December 5, 2022, from <https://docs.oracle.com/javase/tutorial/getStarted/intro/definition.html>
- Chapter 1 Introduction to the Java Programming Environment (JDK 1.1 for Solaris Developer's Guide). (n.d.). Retrieved December 5, 2022, from <https://docs.oracle.com/cd/E19455-01/806-3461/6jck06gqb/index.html>
- Evans, B. J., Gough, J., & Newland, C. (2018). Optimizing java: Practical techniques for improving JVM application performance. O'ReillyMedia.
- Flood, C., & Kennke, R. (n.d.). Shenandoah: Theory and Practice.
- Flood, C. H., Kennke, R., Dinn, A., Haley, A., & Westrelin, R. (2016). Shenandoah: An open-source concurrent compacting garbage collector for OpenJDK. *Proceedings of the 13th International Conference on Principles and Practices of Programming on the Java Platform: Virtual Machines, Languages, and Tools*, 1–9. <https://doi.org/10.1145/2972206.2972210>
- Garbage-First Garbage Collector. (n.d.). [Topic]. Oracle Help Center; September 2019. Retrieved July 16, 2023, from <https://docs.oracle.com/en/java/javase/13/gctuning/garbage-first-garbage-collector.html#GUID-3A99AE6C-F80A-4565-A27C-B4AEDF5CDF71>
- Getting Started with the G1 Garbage Collector. (n.d.). Retrieved December 5, 2022, from <https://www.oracle.com/technetwork/tutorials/tutorials-1876574.html>
- Grgic, H., Mihaljević, B., & Radovan, A. (2018). Comparison of garbage collectors in Java programming language. *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 1539–1544. <https://doi.org/10.23919/MIPRO.2018.8400277>
- Grigorev, A. (2022, March 31). Data Engineering Zoomcamp. GitHub. https://github.com/DataTalksClub/data-engineering-zoomcamp/raw/main/week_5_batch_processing/code/06_spark_sql.py
- Jagelid, N. (2020). Performance evaluation of Java garbage collectors for large heap transaction based applications (Dissertation). Retrieved from <https://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-280823>

- Java Garbage Collection Basics. (n.d.). Retrieved December 5, 2022, from <https://www.oracle.com/webfolder/technetwork/tutorials/obe/java/gc01/index.html>
- Li, H., & Wu, M. (2018). DwarfGC: A Space-Efficient and Crash-Consistent Garbage Collector in NVM for Cloud Computing. *2018 IEEE Symposium on Service-Oriented System Engineering (SOSE)*, 192–197. <https://doi.org/10.1109/SOSE.2018.00032>
- Li, H., Wu, M., & Chen, H. (2018). Analysis and Optimizations of Java Full Garbage Collection. *Proceedings of the 9th Asia-Pacific Workshop on Systems*, 1–7. <https://doi.org/10.1145/3265723.3265735>
- Lidén, P. (2018). *Scalable Low-Latency GC in JDK 1*.
- Lindholm, T., Yellin, F., Bracha, G., Buckley, A., & Smith, D. (n.d.). The Java® Virtual Machine Specification. 624.
- Main—Main—OpenJDK Wiki. (n.d.). Retrieved July 16, 2023, from <https://wiki.openjdk.org/display/shenandoah/Main>
- OpenJDK 17 Using JDK Flight Recorder with OpenJDK. (n.d.). Retrieved June 4, 2023 from https://access.redhat.com/documentation/en-us/openjdk/17/pdf/using_jdk_flight_recorder_with_openjdk/openjdk-17-using_jdk_flight_recorder_with_openjdk-en-us.pdf
- Part 1: Introduction to the G1 Garbage Collector. (n.d.). Retrieved July 16, 2023, from <https://www.redhat.com/en/blog/part-1-introduction-g1-garbage-collector>
- Sahin, S., Cao, W., Zhang, Q., & Liu, L. (2016). JVM Configuration Management and Its Performance Impact for Big Data Applications. *2016 IEEE International Congress on Big Data (BigData Congress)*, 410–417. <https://doi.org/10.1109/BigDataCongress.2016.64>
- Shenandoah GC—Main—OpenJDK Wiki. (n.d.-a). Retrieved December 5, 2022, from <https://wiki.openjdk.org/display/shenandoah>
- Sriram, A., Nair, A., Simon, A., Kalambur, S., & Sitaram, D. (2020). A Study on the Causes of Garbage Collection in Java for Big Data Workloads. *2020 IEEE International Conference on Big Data (Big Data)*, 5831–5833. <https://doi.org/10.1109/BigData50022.2020.9378113>
- Tama, C.G.N, 2017, Sistem Operasi untuk Pemrosesan Big Data dengan berbasis Centos 7, Skripsi, Jurusan Ilmu Komputer FMIPA UGM, Yogyakarta.
- The Z Garbage Collector—Main—OpenJDK Wiki. (n.d.-b). Retrieved December 5, 2022, from <https://wiki.openjdk.org/display/zgc>

Universal JVM GC analyzer—Java Garbage collection log analysis made easy.
(n.d.). Retrieved October 23, 2022, from <https://gceasy.io/gc-index.jsp>

Venners, B. (1999). Inside the Java Virtual Machine. McGraw-Hill.

VisualVM: Home. (n.d.). Retrieved October 23, 2022, from
<https://visualvm.github.io/>

Xu, L., Guo, T., Dou, W., Wang, W., & Wei, J. (2019). An experimental evaluation of garbage collectors on big data applications. *Proceedings of the VLDB Endowment*, 12(5), 570–583.
<https://doi.org/10.14778/3303753.3303762>

Zhang, J. (2021). Performance Comparative Analysis on Garbage First Garbage Collector and Z Garbage Collector. *2021 IEEE 3rd International Conference on Frontiers Technology of Information and Computer (ICFTIC)*, 733–740.
<https://doi.org/10.1109/ICFTIC54370.2021.9647167>