# Understanding Heap Dumps and Thread Dumps in Spring Boot Java Applications

This document explains Heap Dumps and Thread Dumps, what information they contain, how to capture them, and how they help diagnose issues in Spring Boot applications. It also covers Grafana integration possibilities.

## 6. Capturing Heap Dump at Runtime and Grafana Integration

Heap dumps can be captured at runtime while the JVM is active, making it possible to analyze issues without stopping the service. This is critical in production when memory usage spikes unexpectedly.

### Capturing Heap Dump at Runtime

To obtain a heap dump during runtime:  
  
1. \*\*Using jcmd (Recommended):\*\*  
 jcmd <PID> GC.heap\_dump /tmp/runtime\_heapdump.hprof  
  
2. \*\*Using jmap:\*\*  
 jmap -dump:live,format=b,file=/tmp/runtime\_heapdump.hprof <PID>  
  
3. \*\*Using VisualVM/JConsole:\*\* Connect and trigger a dump via GUI.  
  
4. \*\*Spring Boot Actuator Endpoint (Optional):\*\*  
 management.endpoints.web.exposure.include=heapdump  
 Access it via: http://localhost:8080/actuator/heapdump

### Can We Get Heap Dump in Grafana?

Grafana cannot display or process heap dump (.hprof) files because they are binary snapshots of JVM memory, not numerical metrics. Grafana and Prometheus handle time-series data — metrics that update continuously, not static dumps.

#### Why Heap Dumps Are Not Available in Grafana

- Heap dumps are large binary files (hundreds of MBs).  
- They contain sensitive data and require offline analysis tools.  
- Grafana visualizes metrics (numbers, trends) rather than static memory states.  
- Processing dumps inside Grafana would pose performance and security risks.

### How Grafana Helps Instead

Grafana can monitor JVM memory metrics via Prometheus and Micrometer, helping detect problems before they cause crashes. By observing heap usage trends, you can identify leaks early and trigger actions automatically.

Key metrics to monitor include:  
- jvm\_memory\_used\_bytes{area="heap"}  
- jvm\_memory\_max\_bytes  
- jvm\_gc\_pause\_seconds\_count  
Alerts can be configured for sustained high heap usage (>90%) to trigger heap dump scripts automatically.

### Automating Heap Dump Triggering from Grafana Alerts

You can connect Grafana alerts to Alertmanager or webhook scripts that execute heap dump commands automatically. For example:  
  
 jcmd <PID> GC.heap\_dump /tmp/auto\_heapdump\_$(date +%F-%H%M).hprof  
  
This workflow ensures heap dumps are captured during critical incidents while Grafana continues to provide live trend visibility.