Characterizing Concurrency of Java Programs

Chenwei Wang, Eric Aubanel, David Bremner, Michael Dawson
University of New Brunswick, IBM Canada
Faculty of Computer Science
{cwang3|aubanel|bremner}@unb.ca, Michael_Dawson@ca.ibm.com

Abstract

We created a tool (implemented on IBM's J9 JVM) to characterize concurrency for Java programs using a set of concurrency related metrics. These metrics can help us better understand Java programs' concurrency behavior, such as how many threads contribute significantly and concurrently and how threads use shared memory.

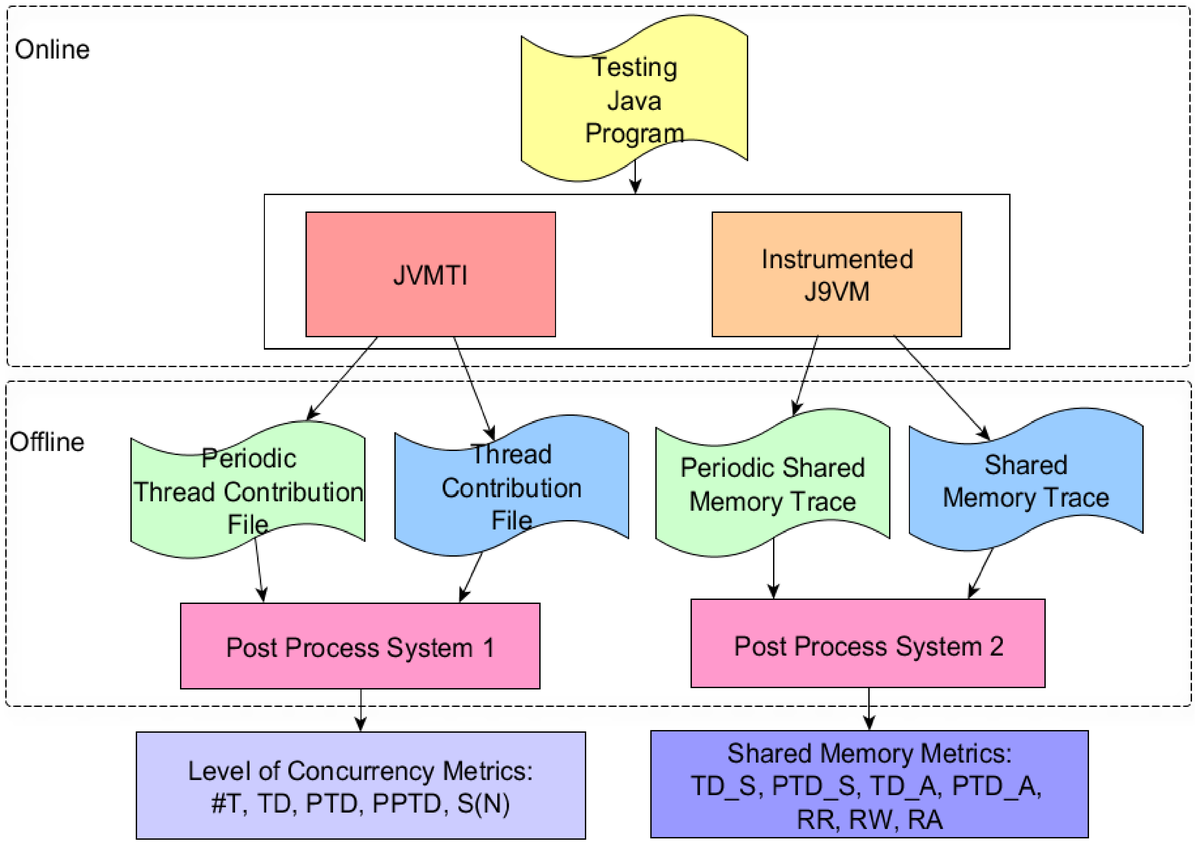
Metrics

Type	Metric Name	Abbr.	Explanation
Level of Concurrency	# of Spawned User Threads	#T	# of spawned user threads, excluding JVM service threads.
	Thread Density	TD	How many threads contribute to the workload significantly (95%). Contribution: CPU time
	Periodic TD	PTD	How many threads contribute to the workload concurrently. median (TD in each period)
	Parallel PTD	PPTD	How many threads contribute to the workload concurrently in the parallel periods.
	Predicted Speedup	S(N)	Predict the maximum speedup according to Amdahl's law. Counting periods with TD<=1 to get the sequential proportion.
Shared Memory	TD for Shared Objects	TD_S	How many threads contribute to shared objects significantly (95%). Contribution: shared read/write operations.
	PTD for Shared Objects	PTD_S	How many threads contribute to shared objects concurrently. median (TD_S in each period)
	TD for Alternating Operations	TD_A	How many threads contribute to alternating operations significantly (95%). Contribution: alternating operations
	PTD for Alternating Operations	PTD_A	How many threads contribute to alternating operations concurrently. median (TD_A in each period)
	Shared Read rate	RR	Rate for shared read operations.
	Shared Write rate	RW	Rate for shared read operations.
	Alternating Operation Rate	RA	Rate for Alternating operations.

Metrics in red: our new contributions

Metrics Tool Overview

The tool consists of an on-line data-collection part and an off-line post processing part.



DaCapo Benchmark Suite Measurement

Level of concurrency metrics 1

benchmark	#T	TD	PTD	PPTD	S/S(32)
sunflow	12	4	4	4	19.5/28.1
xalan	5	4	4	4	23.1/30.2
lusearch	5	5	4	4	10.6/11.2
tomcat	52	10	9	9	7.6/ 8.2

Level of concurrency metrics 2

benchmark	#T	TD	PTD	PPTD
eclipse	851	23	1	2
luindex	2	2	1	0
pmd	34	31	1	29
batik	9	2	1	0
jython	3	1	1	0
fop	2	1	1	0

sunflow, xalan, lusearch and *tomcat* are multi-threaded (TD>1) and parallel (PTD>1), *eclipse, luindex, pmd*, and *batik* are multi-threaded (TD>1) but non-parallel (PTD=1), *jython* and *fop* are sequential programs (TD=1, PTD=1).

Shared memory metrics

benchmark	TD_S	PTD_S	TD_A	PTD_A	RR	RW	RA
sunflow	4	4	4	1	30,528.1	0.6	0.6
xalan	4	4	4	3	6,790.5	104.9	466.3
lusearch	4	4	4	1	449.1	0.1	0.1
tomcat	9	9	10	6	3,984.9	477.8	482.4

In the 4 parallel benchmarks, multiple threads contribute to shared memory significantly and concurrently. *Sunflow* and *lusearch* are candidates for the embarrassingly parallel pattern (PTD_A=1).