

March 17, 2021

nvPlayfair

Amanda Chan, Chau Do, Xinzhe Jiang, Kevin Nguyen

Motivation

Use Case: Safety Requirements with Deadlines



Motivation

Deadline Estimations for Autonomous Vehicles

- The Performance Architecture Team (PAT) needs to ensure that safety requirement deadlines will be satisfied.
 - Understand the performance of the software
 - Detect and analyze execution time outliers



Tegra system-on-a-chip (SoC)

Project Goal

To provide an pipeline for a user to benchmark the performance of a workload on a Tegra board and visualize the results.

Pipeline Overview



Live Demo

Control Application

Front-end

- React provided flexibility and modularity
- Served the React app using an Express.js (Node.js) web server managed by PM2

Back-end

- Utilized Django web service
- Contains
 - Benchmarking program
 - Cross compiler
 - MagLev client
 - Automation scripts for the whole pipeline



Benchmarking Framework

- Design Decisions
 - Python-generated C codeMySQL for code storage
- Multi-threading
- Data collection



Benchmarking Framework

read Name: int read_mock_file_descriptor, read_mock_sz; 1 Setup general: read_mock_file_descriptor= open("sample_read.txt", O_WRONLY | 0_CREAT | 0_TRUNC, 0644); Design Decisions if (read_mock_file_descriptor< 0) Setup before benchmark read_fd = open("sample_read.txt", O_RDONLY); if (read_fd< 0) { Python-generated C code perror("[ERROR] Unable to open file for <read>:"): exit(EXIT_FAILURE); • MySQL for code storage Benchmark: read_sz= read(read_fd , read_c, 10); Multi-threading Data collection close(read_fd); Setup after benchmark: free(read_c); Status: int read_ret = remove("sample_read.txt"); printf("[SUCCESS] Benchmarking READ syscall\\n"); printf("\\n");

Benchmarking Framework

- Design Decisions
 - Python-generated C codeMySQL for code storage
- Multi-threading
- Data collection

	Pin Main Workload to Core		Pin Background Workload to Core		
Core 1	Main Workload 0 ×	.	FORK_BOMB ×	MEMORY_HOG ×	
	Pin Main Workload to Core		Pin Background Work	doad to Core	
Core 2	Select workload to pin	•	Select background	workload to pin	
	Pin Main Workload to Core		Pin Background Workload to Core		
Core 3	Select workload to pin	•	Select background	workload to pin	
	Pin Main Workload to Core		Pin Background Workload to Core		
Core 4	Select workload to pin -		Select background workload to pin		
	Pin Main Workload to Core		Pin Background Workload to Core		
Core 5	Select workload to pin		Select background workload to pin		
	Pin Main Workload to Core		Pin Background Workload to Core		
Core 6	Select workload to pin		Select background workload to pin		
	Pin Main Workload to Core		Pin Background Workload to Core		
Core 7	Select workload to pin	•	Select background	workload to pin	
	Pin Main Workload to Core		Pin Background Workload to Core		
Core 8	Select workload to pin		Select background workload to pin		
Scheduling	Configuration				
Scheduling Configur	ation (optional)				
 Core 1 					
Workload Name		Scheduling Policy		Scheduling Priority	
Main Workload 0		SCHED_OTHER		0	
FORK BOMB					
LOUN_DOWN		SCHED_OTHER	•	0	

Data Visualization

- MagLev and Redash
- A query is dynamically tied to a visualization

Data Visualization

Table	Frequency Vs. Execution Time F
-	
bins	frequency
66,000	1
67,000	1
68,500	1
69,000	1
69,500	4
70,000	1
70,500	2
71,000	3
71,500	2
72,000	4
72,500	9
73,000	6
71,000 71,500 72,000 72,500 73,000	3 2 4 9 6

GROUP BY floor(exec_time / {{ Bin Size }}) * {{ Bin Size }} ORDER BY floor(exec_time / {{ Bin Size }}) * {{ Bin Size }}	
GROUP BY floor(exec time / {{ Bin Size }}) * {{ Bin Size }}	
<pre>SELECT floor(exec_time / {{ Bin Size }}) * {{ Bin Size }} AS bins, count(*) AS frequency FROM pat_interns_benchmark.{{Test Name}}</pre>	

Data Visualization



Conclusion

Goal: To build a platform that measures and visualizes system software performance on Tegra



Future Work

- Run benchmark tests on additional platforms (HV+L, QNX)
- Implement a graph to compare between two runs
- Benchmark higher-level workload(s)
- Run benchmark tests in the background and send completion notification emails to users

Special Thanks

- Allen Martin
- Bill Armstrong
- Waqar Ali
- Mitch Luban
- Professor Mark Claypool
- Performance Architecture Team

nvPlayfair

Amanda Chan, Chau Do, Xinzhe Jiang, Kevin Nguyen

