

Performance Issues Detection by Comparing Nested Execution Traces in Distributed Systems

Maryam Ekhlasi May 16th, 2022

Polytechnique Montreal

DORSAL Laboratory

- Help users understand what is happening when something is different.
 - Performance differences between versions of the application.
 - Performance differences between system configurations.
 - Why some requests are slower than others?



Previous Work

IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 47, NO. 2, FEBRUARY 2021

Fault Analysis and Debugging of Microservice Systems: Industrial Survey, Benchmark System, and Empirical Study

Xiang Zhou[®], Xin Peng[®], Tao Xie, *Fellow, IEEE*, Jun Sun, Chao Ji, Wenhai Li, and Dan Ding

2020 IEEE 20th International Working Conference on Source Code Analysis and Manipulation (SCAM)

DepGraph: Localizing Performance Bottlenecks in Multi-Core Applications Using Waiting Dependency Graphs and Software Tracing

Naser Ezzati-Jivan
Brock UniversityQuentin FournierMichel R. DagenaisAbdelwahab Hamou-LhadjBrock UniversityPolytechnique MontrealPolytechnique MontrealConcordia Universitynezzatijivan@brocku.caquentin.fournier@polymtl.camichel.dagenais@polymtl.cawahab.hamou-lhadj@concordia.ca



DIGITAL ACCESS TO SCHOLARSHIP AT HARVARD DASH.HARVARD.EDU



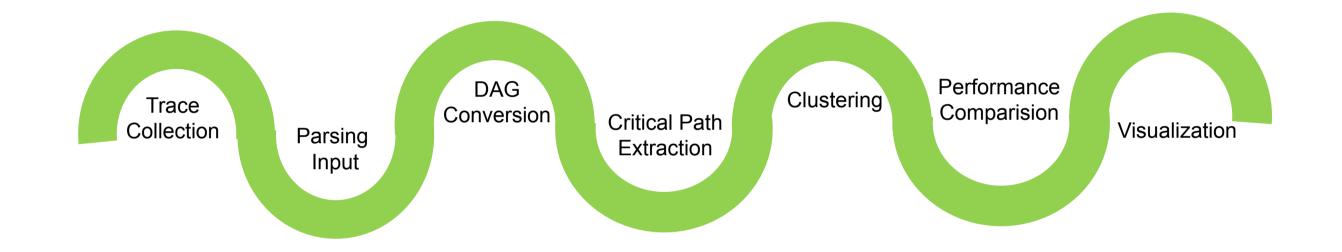
Using Performance Variation for Instrumentation Placement in Distributed Systems



3/13 dorsal.polymtl.ca

243

Roadmap

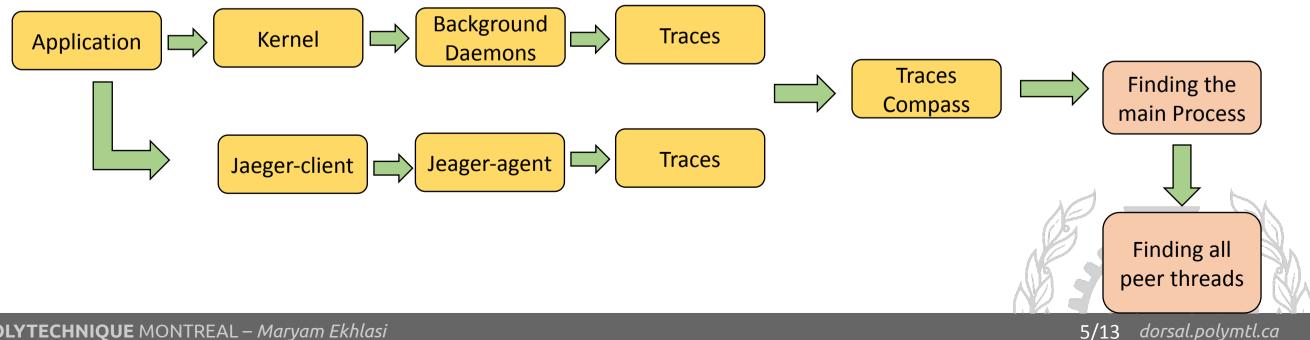


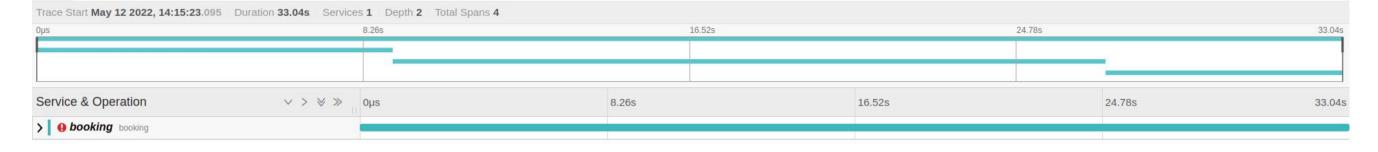


POLYTECHNIQUE MONTREAL – Maryam Ekhlasi

Trace Collection

- Instrumenting Jaeger source code with LTTng.
 - Defining the start and endpoint of each span.
- Converting JSON to DAG.
- Illustrating the communication between the main thread and its peer on the kernel side.





Control Flow ×											🚯 😼 😫 💆 🖗	₽ - ₽ ₽ ₽	10 01 to 01 X	📑 🕶 🖇 🗖 I
rocess	TID	PID	PTID -	Birth time	12:00:39.782	12:00:39.784	12:00:39.786	12:00:39.788	12:00:39.790	12:00:39.792	12:00:39.794	12:00:39.796	12:00:39.798	12:00:39.800
	2170572 2171052 2182112 2182113 t 2182114 t 2182115 2182116	2170572 2171052 2182112 2182112 2182112 2182112 2182112 2182112	2170572 2171052 2182112 2182113 2182113 2182113 2182112	12:00:39.711202711 12:00:39.711582785 12:00:39.731428180 12:00:42.676958408 12:00:42.714719176 12:00:42.748375003 12:00:42.748416858 12:00:42.83874107 12:00:42.879236517					→ Ŭ					■ #

≡ kernel	🗄 finishSpan 🗙	i≡ ust/uid/1000/64-bit	
----------	----------------	------------------------	--

	Trace	Timestamp	Channel	CPU	Event type	Contents	TID	Prio	PID	Source	Binary Location	FL
R	<srch></srch>	<srch></srch>	<srch></srch>	<srch></srch>	<srch></srch>	<srch></srch>	<srch></srch>	<srch></srch>	<srch></srch>	<srch></srch>	<srch></srch>	<s< th=""></s<>
	kernel	12:00:39.704 534 967	kernelchannel_	2 2	kmem_kmalloc	call_site=0xfffffffc14d95fe, ptr=0xffffa0cf6fb90000, bytes_req=739, bytes_alloc=1024, gfp_flags=3264, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 537 895	kernelchannel_	2 2	kmem_kfree	call_site=0xfffffffc14d9654, ptr=0xffffa0cf6fb90000, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 539 324	kernelchannel_	2 2	kmem_kmalloc	call_site=0xffffffc14d95fe, ptr=0xffffa0ce22dd0a00, bytes_req=312, bytes_alloc=512, gfp_flags=3264, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 540 225	kernelchannel_	2 2	kmem_kfree	call_site=0xfffffffc14d9654, ptr=0xffffa0ce22dd0a00, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 541 026	kernelchannel_	2 2	kmem_kmalloc	call_site=0xfffffffc14d95fe, ptr=0xffffa0ce622d9d30, bytes_req=16, bytes_alloc=16, gfp_flags=3264, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 541 208	kernelchannel_(0 0	power_cpu_idle	state=4294967295, cpu_id=0, context.packet_seq_num=0, context.cpu_id=0, contexttid=0, contextpid=0	0		0			
	kernel	12:00:39.704 541 299	kernelchannel_	2 2	kmem_kfree	call_site=0xfffffffc14d9654, ptr=0xffffa0ce622d9d30, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 541 968	kernelchannel_	2 2	kmem_kmalloc	call_site=0xfffffffc14d95fe, ptr=0xffffa0ce30e72c50, bytes_req=2, bytes_alloc=8, gfp_flags=3264, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, context.pid=857	1050		857			
	kernel	12:00:39.704 542 171	kernelchannel_	2 2	kmem_kfree	call_site=0xfffffffc14d9654, ptr=0xffffa0ce30e72c50, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 542 262	kernelchannel_	0 0	power_cpu_idle	state=1, cpu_id=0, context.packet_seq_num=0, context.cpu_id=0, contexttid=0, contextpid=0	0		0			
	kernel	12:00:39.704 542 525	kernelchannel_	2 2	kmem_kmalloc	call_site=0xfffffffc14d95fe, ptr=0xffffa0ce30e72c50, bytes_req=2, bytes_alloc=8, gfp_flags=3264, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, context.pid=857	1050		857			
	kernel	12:00:39.704 542 697	kernelchannel_	2 2	kmem_kfree	call_site=0xfffffffc14d9654, ptr=0xffffa0ce30e72c50, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 542 849	kernelchannel_	2 2	kmem_kmalloc	call_site=0xfffffffc14d95fe, ptr=0xffffa0ce30e72c50, bytes_req=2, bytes_alloc=8, gfp_flags=3264, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 543 001	kernelchannel_2	2 2	kmem_kfree	call_site=0xfffffffc14d9654, ptr=0xffffa0ce30e72c50, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 543 153	kernelchannel_	2 2	kmem_kmalloc	call_site=0xfffffffc14d95fe, ptr=0xffffa0ce30e72c50, bytes_req=2, bytes_alloc=8, gfp_flags=3264, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, context.pid=857	1050		857			
	kernel	12:00:39.704 543 316	kernelchannel_	2 2	kmem_kfree	call_site=0xfffffffc14d9654, ptr=0xffffa0ce30e72c50, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 543 468	kernelchannel_	2 2	kmem_kmalloc	call_site=0xffffffc14d95fe, ptr=0xffffa0ce30e72c50, bytes_req=2, bytes_alloc=8, gfp_flags=3264, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 543 619	kernelchannel_	2 2	kmem_kfree	call_site=0xfffffffc14d9654, ptr=0xffffa0ce30e72c50, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 543 771	kernelchannel_	2 2	kmem_kmalloc	call_site=0xfffffffc14d95fe, ptr=0xffffa0ce30e72c50, bytes_req=2, bytes_alloc=8, gfp_flags=3264, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 543 934	kernelchannel_	2 2	kmem_kfree	call_site=0xfffffffc14d9654, ptr=0xffffa0ce30e72c50, context.packet_seq_num=0, context.cpu_id=2, contexttid=1050, contextpid=857	1050		857			
	kernel	12:00:39.704 544 086	kernelchannel :	2 2	kmem kmalloc	call site=0xffffffc14d95fe, ptr=0xffffa0ce30e72c50, bytes req=2, bytes alloc=8, gfp flags=3264, context.packet seq num=0, context.cpu id=2, context. tid=1050, context. pid=857	1050		857			

POLYTECHNIQUE MONTREAL – Maryam Ekhlasi

6/13 dorsal.polymtl.ca

- -

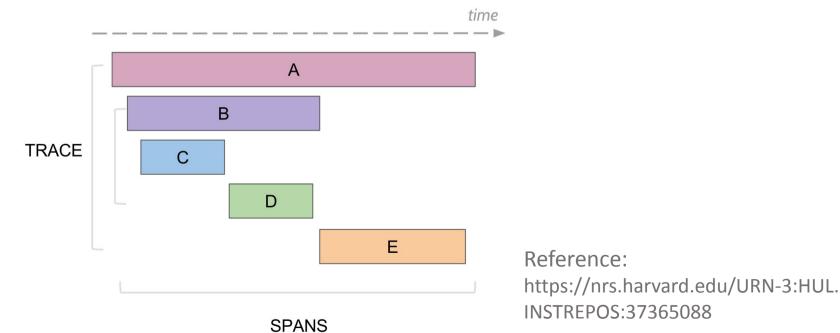
- Local problem
 - Bad code
 - ✔ Infinite loop.
- System problem
 - System misconfiguration.
 - Workload on the CPU/ Memory/ Network.
 - ✓ Using benchmark tools for creating HW load.

WARNING:num-threads is deprecated, usethreads instead sysbench 1.0.18 (using system LuaJIT 2.1.0-beta3)	1 [
Running the test with following options:	4 [111111111111111111111111111111111111
Number of threads: 16	Mem[]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]
Initializing random number generator from current time	Swp[0K/2.00G] Load average: 3.80 1.27 0.73
	Uptime: 4 days, 05:58:48
Prime numbers limit: 10000	PID USER PRI NI VIRT RES SHR S CPUX MEM% TIME+ Command
	460523 maryam 20 0 34456 9460 8044 S 1587 0.0 1:31.73 sysbench cpu runnum-threads=16
Initializing worker threads	460537 maryam 20 0 34456 9460 8044 R 100. 0.0 0:05.76 sysbench cpu runnum-threads=16
	460532 maryam 20 0 34456 9460 8044 R 100. 0.0 0:05.76 sysbench cpu runnum-threads=16
Threads started!	460535 maryam 20 0 34456 9460 8044 R 100. 0.0 0:05.76 sysbench cpu runnum-threads=16
	460525 maryam 20 0 34456 9460 8044 R 100. 0.0 0:05.75 sysbench cpu runnum-threads=16
CPU speed:	460533 maryam 20 0 34456 9460 8044 R 100. 0.0 0:05.75 sysbench cpu runnum-threads=16
events per second: 45867.53	460536 maryam 20 0 34456 9660 8044 R 100. 0.0 0:05.75 sysbench cpu runnum-threads=16
	460527 maryam 20 0 34456 9460 8044 R 100. 0.0 0:05.74 sysbench cpu runnum-threads=16
General statistics:	460534 maryam 20 0 34456 9460 8044 R 100. 0.0 0:05.74 sysbench cpu runnum-threads=16 460526 maryam 20 0 34456 9460 8044 R 99.4 0.0 0:05.74 sysbench cpu runnum-threads=16
total time: 10.0003s total number of events: 458710	460526 maryam 20 0 34456 9460 8044 R 99.4 0.0 0:05.74 sysbench cpu runnum-threads=16 460528 maryam 20 0 34456 9460 8044 R 99.4 0.0 0:05.74 sysbench cpu runnum-threads=16
	460526 Maryam 20 0 34456 9460 8044 R 99.4 0.0 0:05.74 Sysbench cpu runnum-threads=16
Latency (ms):	460539 maryam 20 0 34456 9460 8044 R 99.4 0.0 0:05.68 sysbench cpu run -num-threads=16
min: 0.19	460531 maryam 20 0 34455 9460 8044 R 98.7 0.0 0:05.06 system cpu run -num-threads-10
avg: 0.35	460531 maryam 20 0 34456 9460 8044 R 98.1 0.0 0:05.70 system cpu run -num-threads=16
max: 24.35	460524 marvam 20 0 34456 9460 8044 R 96.8 0.0 0:05.71 system con court in -num-threads-16
95th percentile: 0.35	46052 maryum 20 0 34455 9460 8044 R 96.2 0.0 0:05.65 sysbench cpu ruh -num threads-16
sum: 159949.73	17277 maryam 20 0 6835M 376M 123M 5 7.1 1.2 23:35.48 /usr/bin/gnome-shell
	387848 marvam 20 0 13804 7204 3644 F 1.9 0.0 32:01.25 htop
Threads fairness.	17088 marvam 20 0 1945M 220M 142M 51 3 0 7 25:52 19 /usr/lib/xorg/Xorg vt2 -displayfd 3 -auth /rup/user/1000/odm/Xauthori

7/13 dorsal.polymtl.ca

POLYTECHNIQUE MONTREAL – Maryam Ekhlasi

- Span or swim lane showing call graph relationship
- DAG also shows call graph and the task's relationship
- and the task's relationship
 - Concurrent behavior
 - Synchronization points
 - Asynchronous Behavior



Concurrent behavior: Concurrent behavior: Asynchronous Synchronous Request Request start start Request Synch Asynchronous stop point branch Request stop

POLYTECHNIQUE MONTREAL – Maryam Ekhlasi

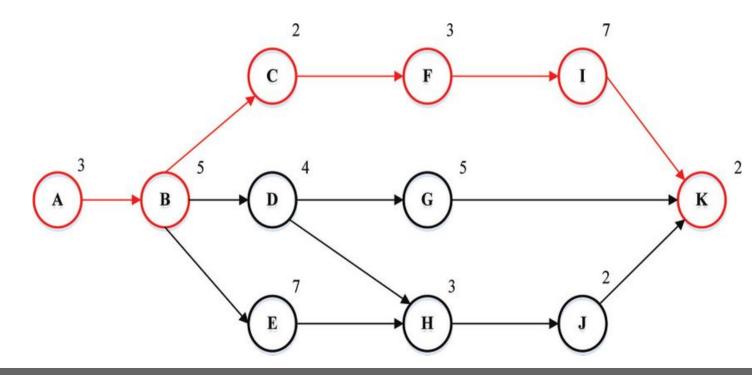


- Extracting critical path
 - Concurrency

Finding the path between the start and endpoint of the execution while we are not waiting for something else.

No concurrency

The workflow of the application.



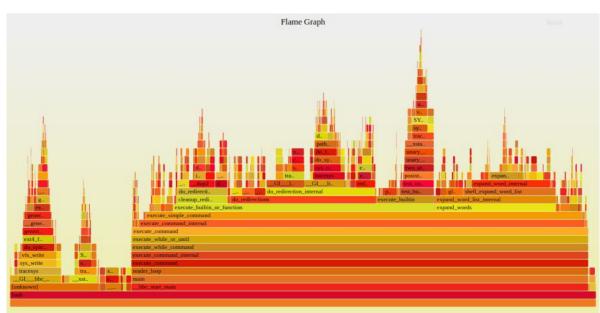


POLYTECHNIQUE MONTREAL – Maryam Ekhlasi

9/13 dorsal.polymtl.ca

- Categorization
 - Thresholding
 Thresholding
 Number of page fault
 Memory usage
 Execution time
 ...
 - Clustering

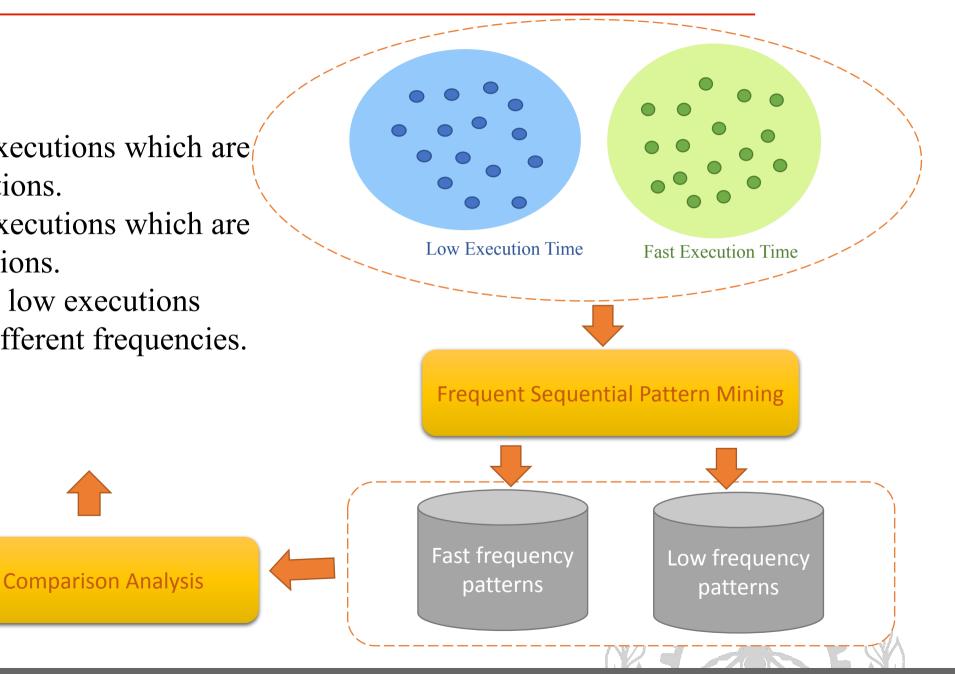
Combination of performance metrics





POLYTECHNIQUE MONTREAL – Maryam Ekhlasi

- Comparison:
- Frequent patterns of fast executions which are not available in low executions.
- Frequent patterns of low executions which are not available in fast executions.
- Common patterns between low executions and fast executions with different frequencies.



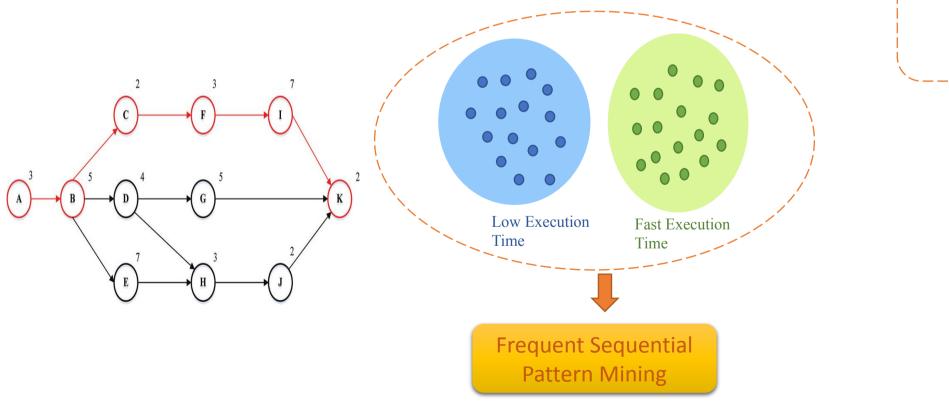
dorsal.polymtl.ca

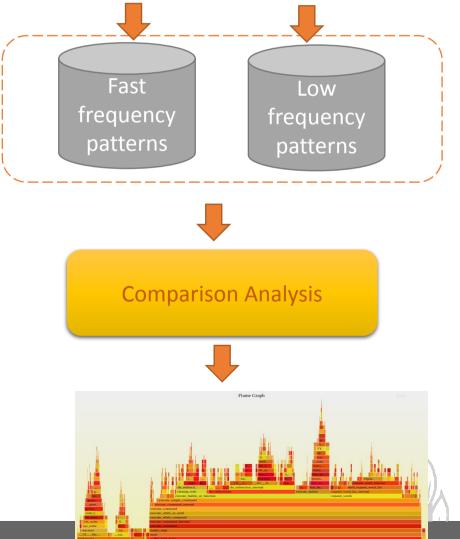
11/13

POLYTECHNIQUE MONTREAL – Maryam Ekhlasi

Future Goal

- Extracting critical path based on the DAG.
- Categorizing executions.
- Localizing performance problems.







Email: maryam.ekhlasi@polymtl.ca

Source Code: https://github.com/maryamekhlasi/jaeger-client-python.git



POLYTECHNIQUE MONTREAL – *<Maryam Ekhlasi>*