

Quick Service (온라인 장애 분석 서비스)

시스템은 점점 늘어나고 복잡해 지는 데....

갑자기 시스템이 느려졌지만
원인을 찾지 못하겠네요.

시스템 운영 상태를
점검하고 향후 운영을 위하여
평가를 받고 싶어요.

운영상에 문제점이
너무 많아서
전문가의 의견이
필요해요.

곧 부하테스트와
시스템 오픈인데
문제가 생길까봐
불안해요.

- 시스템에 대한 모니터링은 잘 되고 있지만 애플리케이션은 블랙박스화 되어 내부 상태를 파악하지 못함
- 장애나 성능 이슈의 90% 이상이 애플리케이션이 원인

웹시스템 진단 서비스
"바로 지금 신청하세요."

Opennaru Service Desk

고객께서 Quick Service를 통해
전달해 주신
화면과 Thread Dump 및
메타 정보를 접수합니다.
자동으로 케이스를 생성합니다.

Opennaru Quick Service 란?



고객사

① 시스템
이슈발생



② 쓰레드 덤프와
모니터링 화면 전달



③ 오픈나루
서비스 데스크 접수



⑤ 장애 분석
보고서 전달



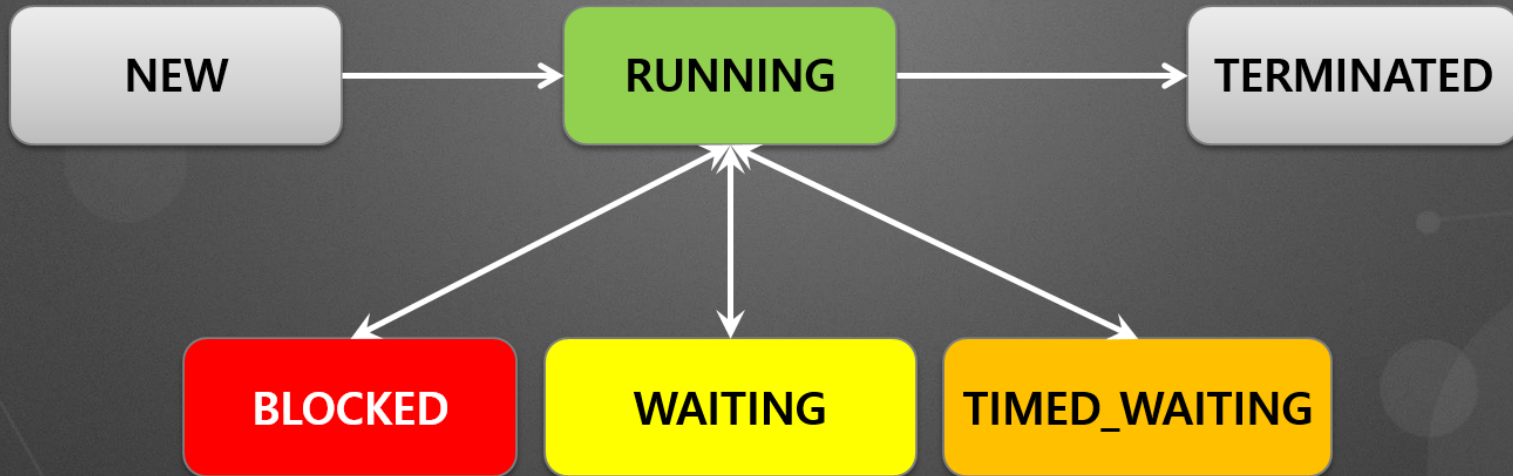
④ 장애 분석



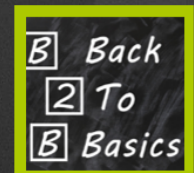
오픈나루

Thread 상태

- `java.lang.Thread.State`

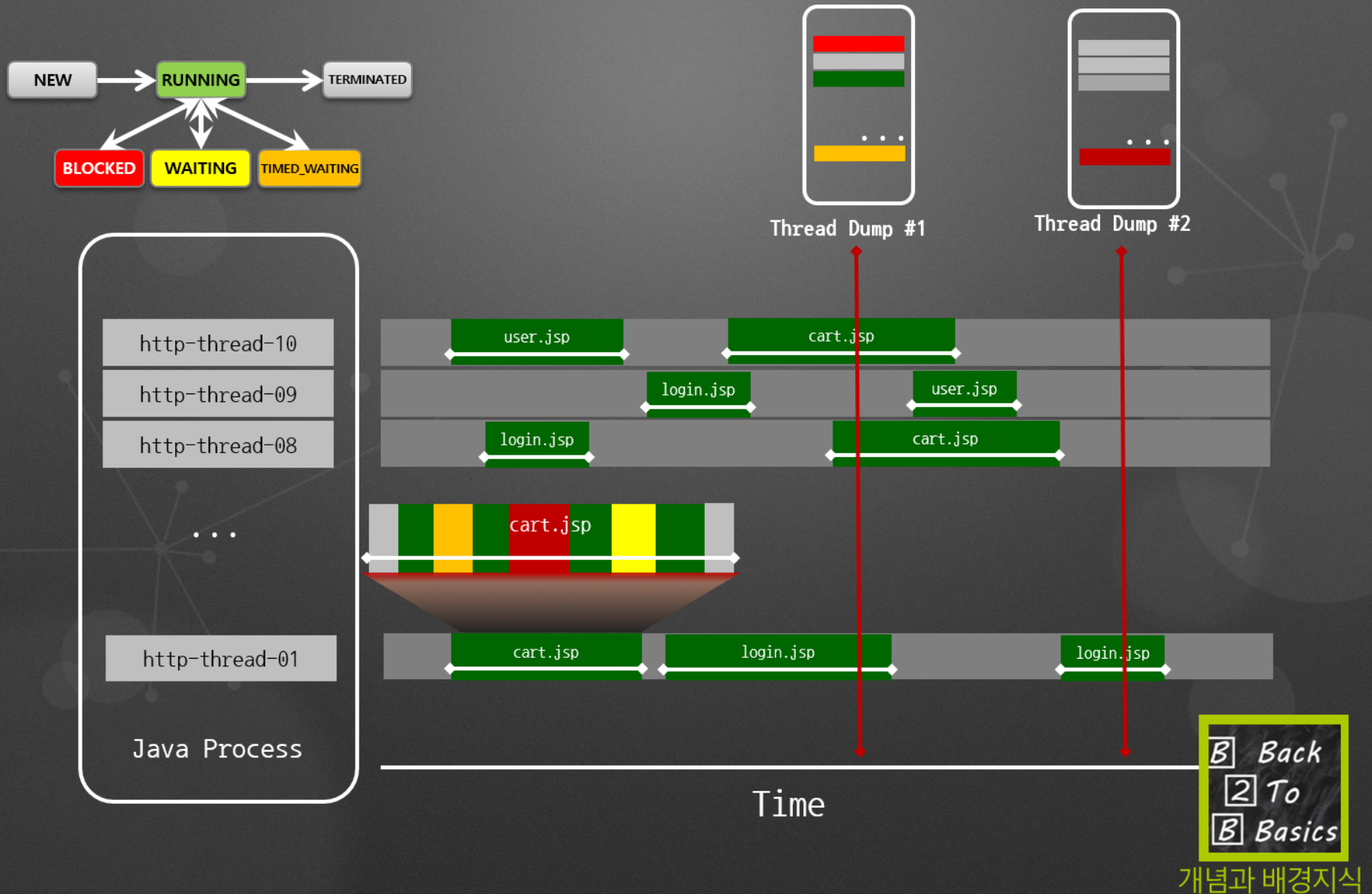


- NEW – 생성되어있지만 실행되지 않은 상태
- RUNNING – 작업 수행 중
- BLOCKED – 다른 스레드의 Lock 해제를 기다리는 중
- WAITING – `wait()`, `join()`, `park()` 메소드로 대기 중
- TIMED_WAITING – WAITING과 같지만, 정해진 시간만 대기
- TERMINATED – 종료됨



개념과 배경지식

Java Thread Dump의 이해



스레드 덤프의 정보 구성

- 스레드 덤프의 각 스레드의 정보 구성



스레드 이름

우선순위

스레드ID

OS 스레드ID

스레드 상태

```

"http-thread-pool-threads - 250" prio=0 tid=0x0000000000000000 nid=0xdbc9 waiting on condition [0x0000000000000000]
java.lang.Thread.State: WAITING (parking)
    at sun.misc.Unsafe.park(Native Method)
    - parking to wait for <0x0000000000000000> (a java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject)
    at java.util.concurrent.locks.LockSupport.park(LockSupport.java:186)
    at java.util.concurrent.locks.AbstractQueuedSynchronizer$ConditionObject.await(AbstractQueuedSynchronizer.java:2043)
    at java.util.concurrent.LinkedBlockingQueue.take(LinkedBlockingQueue.java:442)
    at java.util.concurrent.ThreadPoolExecutor.getTask(ThreadPoolExecutor.java:1068)
    at java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1130)
    at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:615)
    at java.lang.Thread.run(Thread.java:744)
    at org.jboss.threads.JBossThread.run(JBossThread.java:122)

```

스레드 호출 Stack

☐ Back
☒ To
☐ Basics

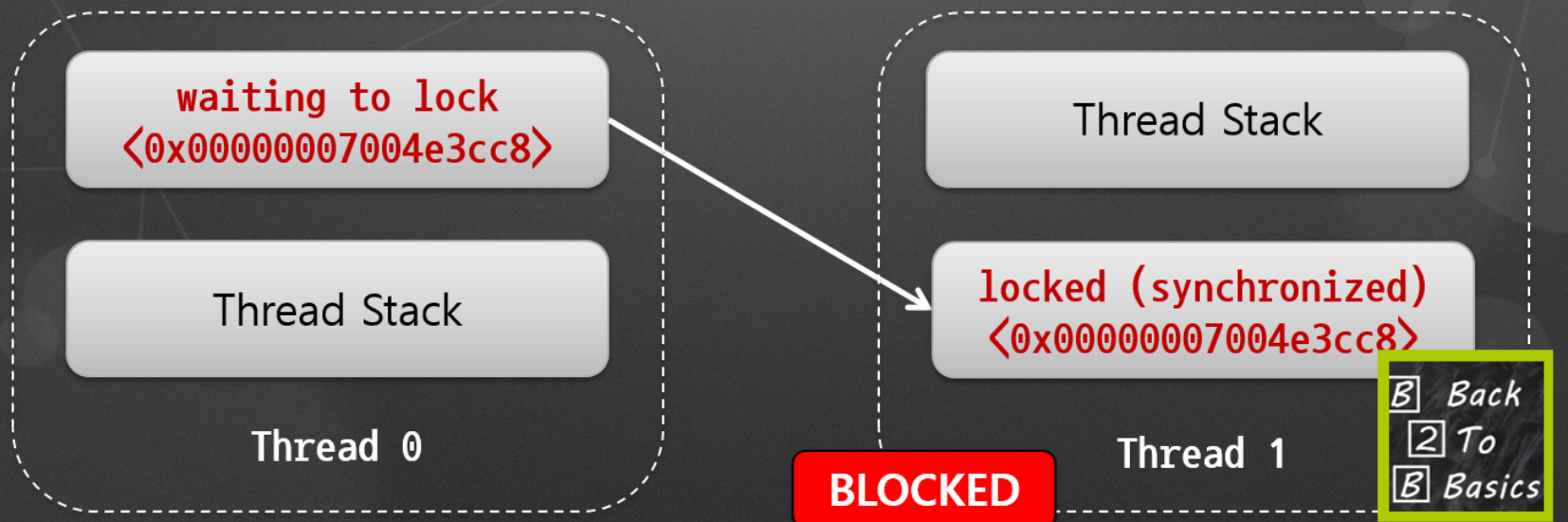
개념과 배경지식

Thread Dump의 Blocked의 의미

- 여러 스레드에서 같은 변수나 메소드를 변경할 경우
- `synchronized`로 다른 스레드의 처리가 끝날 때 까지 제어가 넘어가지 않도록 보호

```
// 특정한 객체에 lock을 걸고자 할 때
synchronized(객체의 참조변수){
    // ...
}

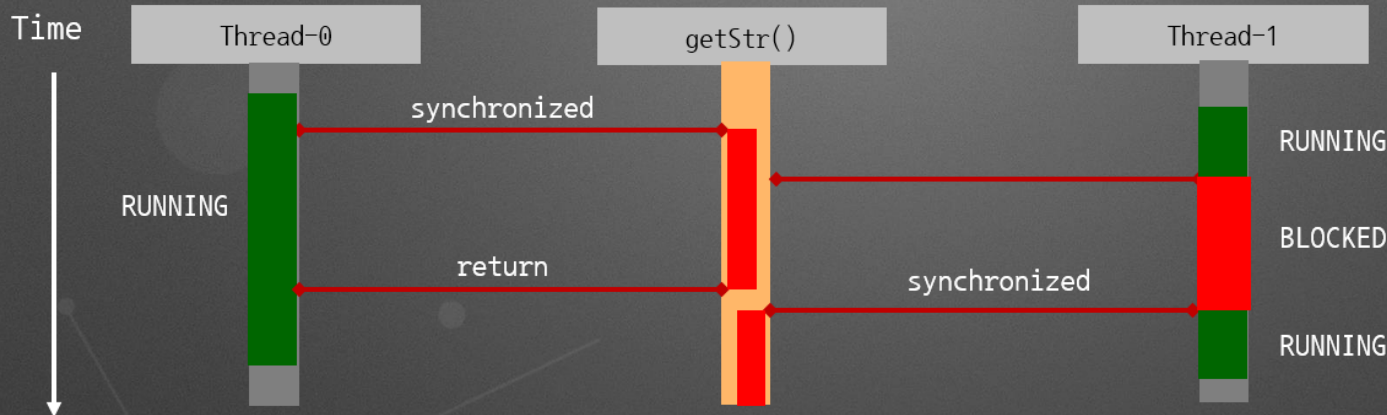
// 메서드에 lock을 걸고자할 때
public synchronized void methodName(){
    // ...
}
```



개념과 배경지식

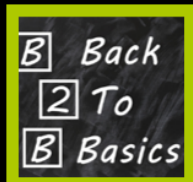
Java Thread Dump의 BLOCKED의 의미

- waiting for monitor entry
 - 오브젝트에 대한 락의 해제를 기다리고 있는 상태



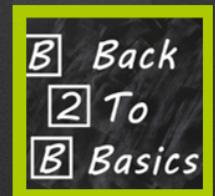
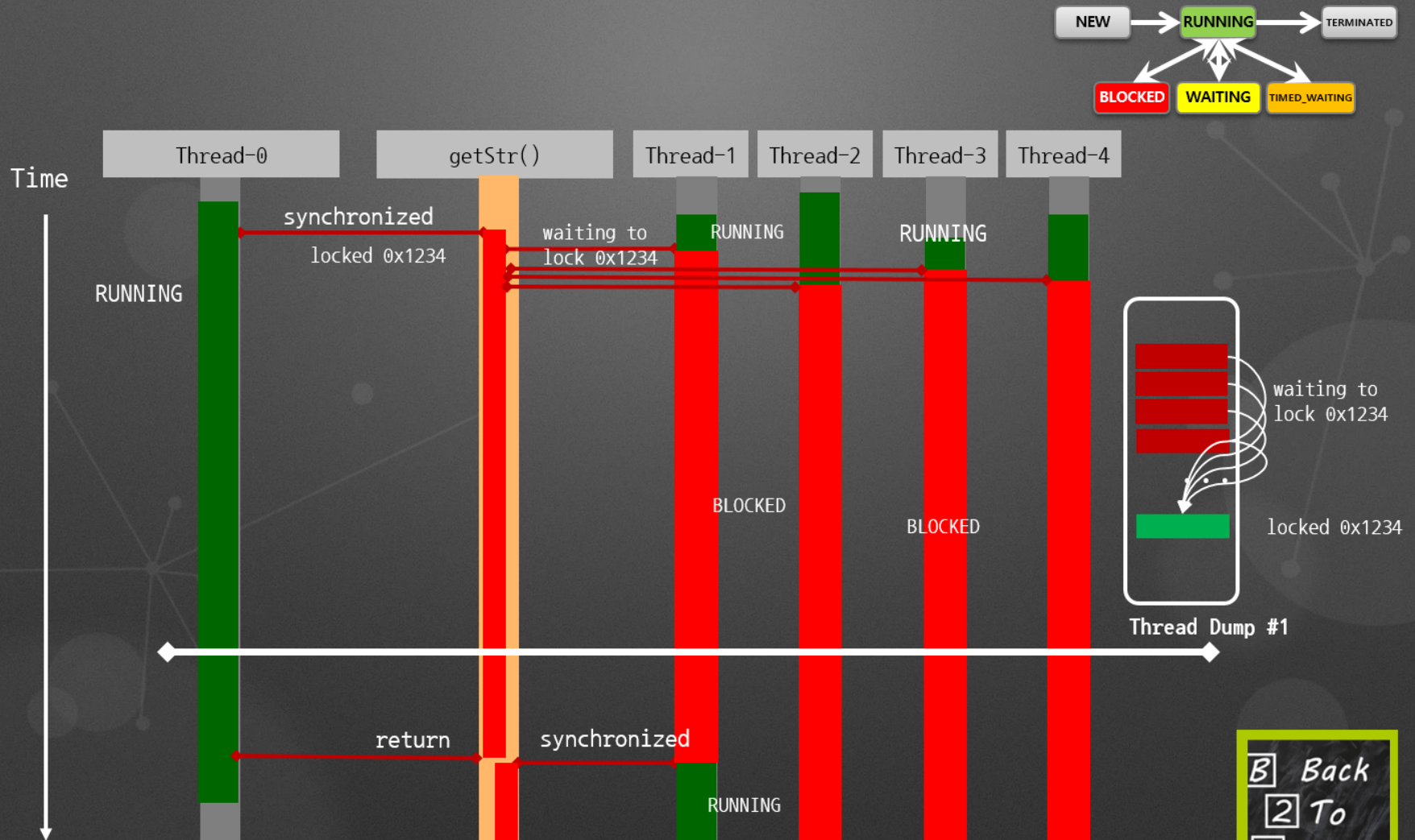
```

"Thread-1" prio=10 tid=0x00007f69fc0b8800 nid=0x7358 waiting for monitor entry [0x00007f6a0050a000]
  java.lang.Thread.State: BLOCKED (on object monitor)
    at sample.SampleMain.getStr(SampleMain.java:23)
    - waiting to lock <0x000000077acacf60> (a java.lang.Class for sample.SampleMain)
    at sample.SampleMain.run(SampleMain.java:17)
    at java.lang.Thread.run(Thread.java:679)
"Thread-0" prio=10 tid=0x00007f69fc0b6800 nid=0x7357 waiting on condition [0x00007f6a0060b000]
  java.lang.Thread.State: TIMED_WAITING (sleeping)
    at java.lang.Thread.sleep(Native Method)
    at sample.SampleMain.getStr(SampleMain.java:23)
    - locked <0x000000077acacf60> (a java.lang.Class for sample.SampleMain)
    at sample.SampleMain.run(SampleMain.java:17)
    at java.lang.Thread.run(Thread.java:679)
  
```



개념과 배경지식

Thread BLOCKED 상태가 길어지면 발생하는 상황



개념과 배경지식



Thread Dump 분석 방법

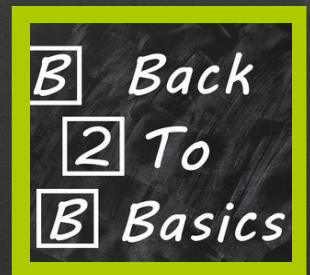
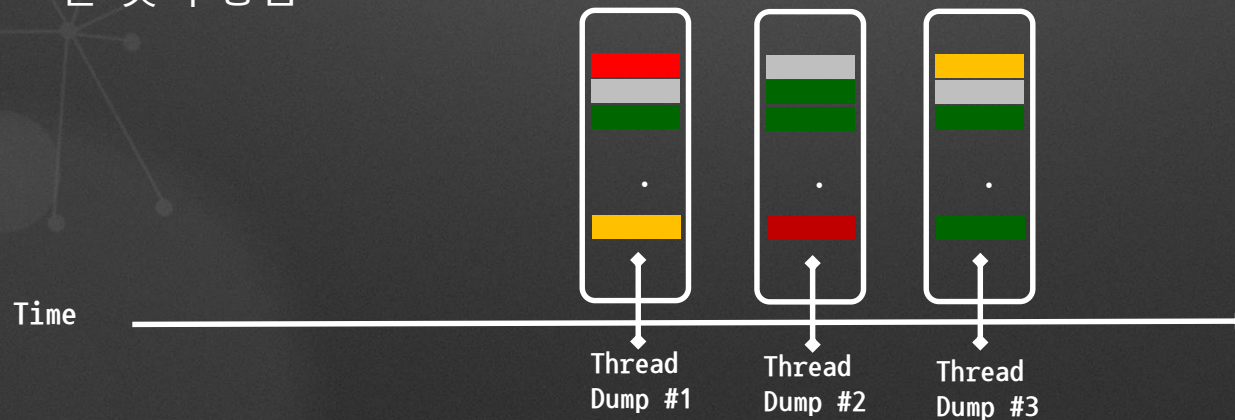
스레드 덤프

• 스레드 덤프란?

- JVM 내부에서 현재 수행중인 스레드의 SnapShot을 텍스트로 출력
- 서버 Hang, 서버 Slowdown등 서버에서 발생하는 문제를 해결하는 가장 유용한 도구
- 약 3~5초간격으로 3~5회 연속해서 스레드 덤프를 받아야 분석이 가능하다

• 언제 받아야 하나?

- 운영 중인 서버에서도 수행가능(많은 시간이 걸리지 않는 작업임)
- 서버 Hang, Slowdown시, 서버가 Crash되거나 OutOfMemory가 발생한 직후에 받는 것이 좋음



개념과 배경지식

스레드 덤프를 분석

Take at least 3 times, with 3 to 5 seconds interval



진단 및 모니터링

Thread 1
Thread 2
Thread 3
Thread 4
Thread 5
Thread 6
Thread 7
Thread 8
Thread 9

스레드 덤프 받는 방법은?

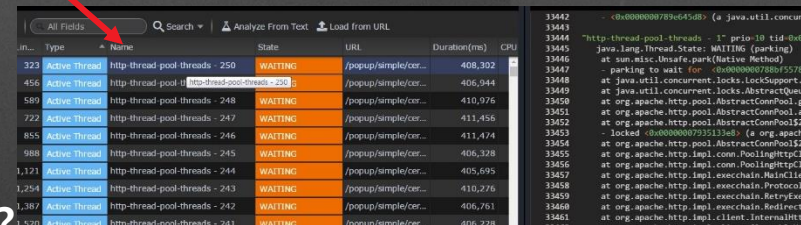
- Linux / UNIX 시스템에서 스레드 덤프
 - kill -3 <PID>
 - 일반적을 nohup 명령을 이용하여 시작하기 때문에 nohup.out파일에 stdout 로그가 남게 됨
 - 윈도우에서는 Ctrl + Break
 - KHAN [apm] 화면에서 버튼 클릭(장애시 자동 생성)



- Docker와 Openshift 같은 환경
 - 스레드 덤프를 어떻게 받나?



- docker exec -it c4343003c1 /bin/bash
~# kill -3 <PID>
- 내용을 어떻게 복사? nohup 로그는 남기나?
 - docker cp c4343003c1:/file/path/nohup.out /host/path/target
- KHAN [apm]은 Openshift, Docker를 지원 → 버튼 클릭 & 분석 UI 제공

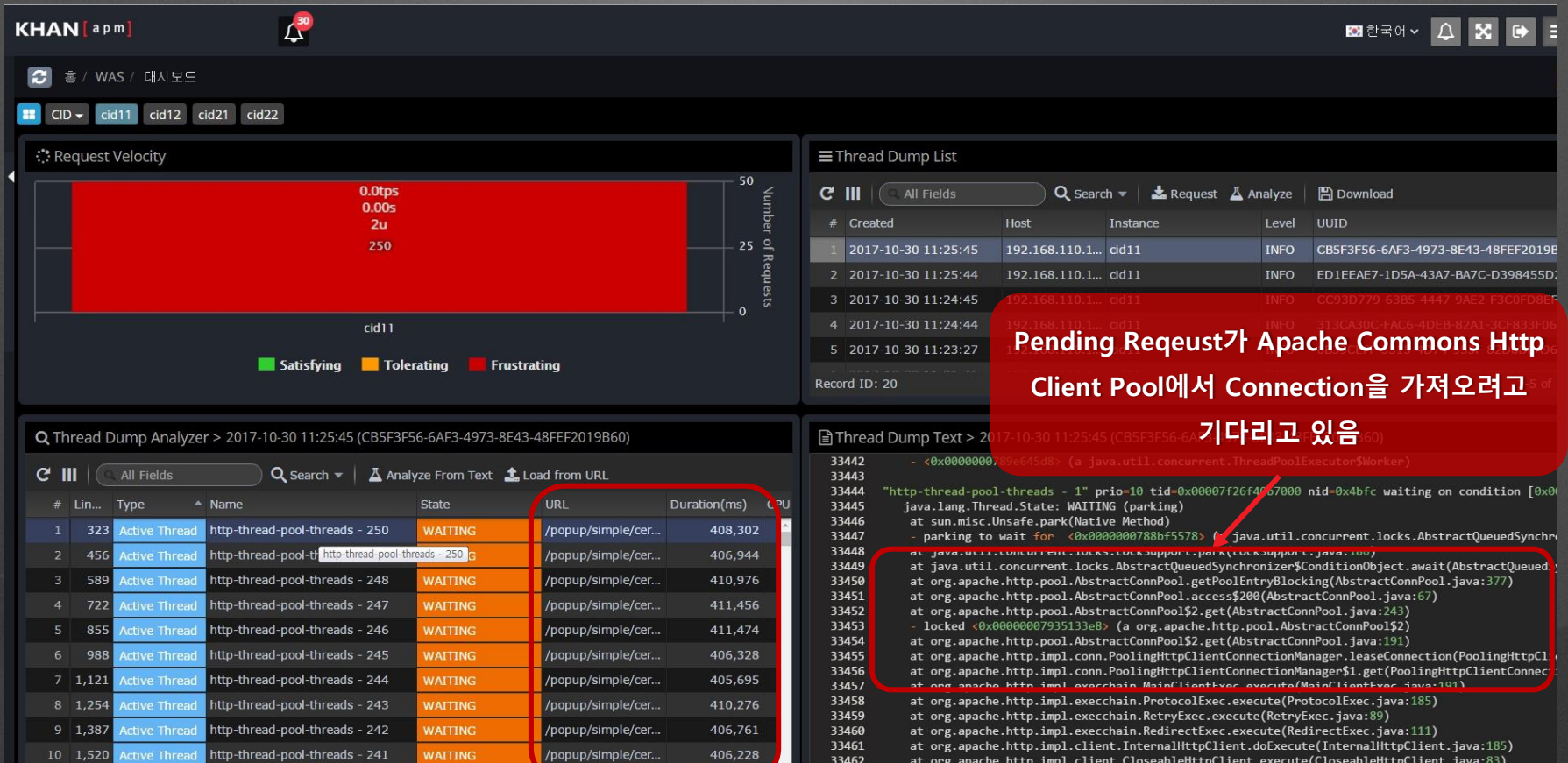


ID	Type	Name	State	URL	Duration(ms)	CPU
3223	Active Thread	http-thread-pool-threads - 250	WAITING	/popup/sample/cor...	408,302	
456	Active Thread	http-thread-pool-threads - 248	WAITING	/popup/sample/cor...	406,944	
569	Active Thread	http-thread-pool-threads - 248	WAITING	/popup/sample/cor...	410,976	
722	Active Thread	http-thread-pool-threads - 247	WAITING	/popup/sample/cor...	411,456	
855	Active Thread	http-thread-pool-threads - 246	WAITING	/popup/sample/cor...	411,474	
988	Active Thread	http-thread-pool-threads - 245	WAITING	/popup/sample/cor...	406,328	
1,121	Active Thread	http-thread-pool-threads - 244	WAITING	/popup/sample/cor...	405,695	
1,254	Active Thread	http-thread-pool-threads - 243	WAITING	/popup/sample/cor...	410,276	
1,387	Active Thread	http-thread-pool-threads - 242	WAITING	/popup/sample/cor...	406,761	
1,520	Active Thread	http-thread-pool-threads - 241	WAITING	/popup/sample/cor...	406,228	

Server가 느려 지는 상황에서 Dashboard



KHAN [apm] 스레드 덤프 분석 예시



Pending Request가 Apache Commons Http Client Pool에서 Connection을 가져오려고 기다리고 있음

스레드에서 어떤 URL이 얼마동안, CPU 자원을 사용하며 실행중인지 표시함

opennaru

2	924	Active Thre	default task-238	BLOCKED	/cnsl/front/exame...	33,986	693.4	org.apache	0x000...
3	1,061	Active Thre	default task-236	BLOCKED	/cnsl/front/exame...	33,058	628.9	org.apache	0x000...
4	1,181	Active Thre	default task-235	BLOCKED	/cnsl/front/exame...	39,877	745.3	org.apache	0x000...
5	1,588	Active Thre	default task-215	BLOCKED	/cnsl/front/exame...	38,246	745.2	org.apache	0x000...
6	1,737	Active Thre	default task-212	BLOCKED	/cnsl/front/exame...	32,150	637.7	org.apache	0x000...
7	1,983	Active Thre	default task-204	BLOCKED	/cnsl/front/exame...	30,869	627.5	org.apache	0x000...
8	2,117	Active Thre	default task-202	BLOCKED	/cnsl/front/exame...	38,711	757.2	org.apache	0x000...
9	2,314	Active Thre	default task-196	BLOCKED	/cnsl/front/exame...	30,347	608.5	org.apache	0x000...
10	2,498	Active Thre	default task-193	BLOCKED	/cnsl/front/exame...	38,861	757.6	org.apache	0x000...
11	2,620	Active Thre	default task-191	BLOCKED	/cnsl/front/exame...	37,997	699.2	org.apache	0x000...
12	2,767	Active Thre	default task-188	BLOCKED	/cnsl/front/exame...	37,980	699.2	org.apache	0x000...
13	2,911	Active Thre	default task-186	BLOCKED	/cnsl/front/exame...	37,980	699.2	org.apache	0x000...
14	3,261	Active Thre	default task-174	BLOCKED	/cnsl/front/exame...	35,674	638.1	org.apache	0x000...
15	3,419	Active Thre	default task-172	BLOCKED	/cnsl/front/exame...	35,674	638.1	org.apache	0x000...
16	3,633	Active Thre	default task-168	BLOCKED	/cnsl/front/exame...	35,674	638.1	org.apache	0x000...
17	4,007	Active Thre	default task-160	BLOCKED	/cnsl/front/exame...	29,606	589.9	org.apache	0x000...
18	4,144	Active Thre	default task-158	BLOCKED	/cnsl/front/exame...	29,606	589.9	org.apache	0x000...
19	4,278	Active Thre	default task-156	BLOCKED	/cnsl/front/exame...	29,606	589.9	org.apache	0x000...
20	4,400	Active Thre	default task-146	BLOCKED	/cnsl/front/memb...	1,145	11.3	org.apache	0x000...
21	4,577	Active Thre	default task-143	BLOCKED	/cnsl/front/exame...	37,405	694.4	org.apache	0x000...
22	5,179	Active Thre	default task-110	BLOCKED	/cnsl/front/exame...	28,692	618.0	org.apache	0x000...
23	5,302	Active Thre	default task-109	BLOCKED	/cnsl/front/exame...	12,069	282.6	org.apache	0x000...
24	5,589	Active Thre	default task-98	BLOCKED	/cnsl/front/exame...	31,622	630.8	org.apache	0x000...
25	5,738	Active Thre	default task-95	BLOCKED	/cnsl/front/exame...	38,526	743.0	org.apache	0x000...
26	5,917	Active Thre	default task-90	BLOCKED	/cnsl/front/exame...	28,350	612.5	org.apache	0x000...
27	6,265	Active Thre	default task-74	BLOCKED	/cnsl/front/exame...	37,662	743.6	org.apache	0x000...
28	6,496	Active Thre	default task-66	BLOCKED	/cnsl/front/exame...	27,470	557.1	org.apache	0x000...

[illegible]



**" IF YOU CAN'T MEASURE IT
YOU CAN'T MANAGE IT. "**

- Peter Drucker



감사합니다.



제품이나 서비스에 관한 문의

콜 센터 : 02-469-5426 (휴대폰 : 010-2243-3394)

전자 메일 : sales@opennaru.com