

ボトルネック調査のやり方について、自分のわかる範囲内でまとめておく。

負荷の見極め

一般的な負荷は大きく2つに分類されるため、ここをまず調査する。

- CPU負荷
- I/O負荷

これらを見極めるために、まずロードアベレージを見ていく。

ロードアベレージ観測

ロードアベレージを見るにはいくつか方法があるが、筆者は `top` を見ることが多い。

\$ top

top - 10:56:23 up 23 days, 23:03, 2 users, load average: 0.13, 0.08, 0.03

Tasks: 327 total, 1 running, 105 sleeping, 0 stopped, 151 zombie

%Cpu(s): 0.3 us, 0.4 sy, 0.0 ni, 99.3 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st

KiB Mem : 8069384 total, 1033512 free, 1240868 used, 5795004 buff/cache

KiB Swap: 1048572 total, 1047632 free, 940 used. 6661788 avail Mem

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
18199	ygnmhdtt	20	0	407608	28708	7484	S	1.0	0.4	11:22.06	docker-c
30897	root	20	0	728456	27980	13556	S	1.0	0.3	57:39.16	docker-coi
30889	root	20	0	1324280	60788	35364	S	0.7	0.8	33:01.55	dockerd

一番上の `load average` の3つの数字(上記では0.13, 0.08, 0.03) がロードアベレージである。

ロードアベレージとは

そもそもロードアベレージとは、単位時間あたりのCPUによる実行を待っているタスクの数を示している。

基本的には上記のように0.01 ~ 0.2くらいに収まり、感覚では1を超えていると「おっ」と思う。

また、障害が起きてるノードだと、10を超えるのを見たことがある。(こうなると完全にどこかがおかしい)

左から、1分、5分、15分間の平均値を示している。

また、ロードアベレージが示す待ち状態のタスクとは、CPUが専有されていて実行されないタスクだけでなく、

ディスクIOを待っているタスクも含んでいる。

そのため、ロードアベレージが高いだけではCPUネックか、IOネックかは判断できない。

CPU、I/Oのいずれかがボトルネックかを探る

以下、それぞれの基本的な流れを説明する。

ロードアベレージが低い場合

まず、ロードアベレージは低い(感覚的には2 ~ 3以下くらい)のにシステムのスループットが上がらない場合は、サーバには負荷がかかっていない(おそらく)ため、

ソフトウェアの設定や不具合、ネットワーク(TCPコネクション数が増えていないか)、リモートホスト側に原因がないかなどを調査する。

ロードアベレージが高い場合

ロードアベレージが高い場合、ここから、CPU、I/Oのいずれがボトルネックかを調査する。

CPU使用率を確認し、CPU使用率が高ければCPUネックとなる。
筆者はsarコマンドを使用することが多い。

sarはsysstatパッケージに入っているコマンドであり、AmazonLinuxとかはデフォルトでは(たぶん)入ってないので、インストールする必要がある。
使い方としては、

ちなみに、swapがあるのかどうかは `swapon` で確認する。


```
$ swapon -v
```

```
NAME    TYPE    SIZE USED PRIO  
/dev/sda5 partition 1024M 940K -2
```

/dev/sda5がスワップになっている。

CPU負荷が高い場合

CPU負荷が高い場合、以下の流れで探っていく。

- psで見えるプロセスの状態やCPU使用時間などを見ながら、原因となっているプロセスを特定する。
- プロセスを特定したら、straceでシステムコールをトレース、oprofileでプロファイリングをして、ボトルネック箇所を絞り込む。

プログラムに不具合があるケースは不具合を取り除く必要がある。

IO負荷が高い場合

CPU負荷が高くない場合はIO負荷を疑う。

IO不可の場合、プログラムからの入出力が多いか、スワップが発生してディスクアクセスが発生しているかのいずれかである場合がほとんど。

上記の `sar -W` でスワップの発生状況を確認する。

スワップが発生している場合:

- psで極端にメモリを食っているプロセスを特定。
- プログラムの不具合であればプログラムを改善。
- 搭載メモリが不足している場合はメモリ増設で対応。メモリ増設できない場合は分散。

ちなみに、スワップが発生してディスクを読んでいる場合、メモリよりも数

桁遅いらしい。注意。

スワップが発生していない場合:

キャッシュに必要なメモリが不足しているケースが考えられる。

- メモリが増やせる場合は増設。
- それ以外では、分散やキャッシュサーバの導入などを検討。また、プログラム改善でI/O頻度を軽減する。

まとめ

こういうのは、慣れてないと無理。障害を起こす訓練とかやりたい。

その他

以下のようなコマンドもよく使う。

- vmstat

```
$ vmstat 1 1000
```

```
procs -----memory----- ---swap-- -----io---- -system-- -----cpu-----
```

```
 r b swpd free buff cache si so bi bo in cs us sy id wa st
```

```
 1 0  940 1033456 942644 4852684  0  0  5 19  1  2 2 0 98 0 0
```

```
 0 0  940 1033628 942644 4852688  0  0  0  0 278 644 1 0 99 0 C
```

```
 0 0  940 1033596 942644 4852688  0  0  0  0 267 635 1 0 99 0 C
```

- iftop … ネットワークのI/Oプロファイリング
- iotop … プロセスごとのDISK I/Oが見られる

