

本TIPsは、以下の場面で活用します。

✓ CPU使用率が90%以上の状態が続いた

✓ CPU使用率に関するアラートが発生した



活用TIPs: CPU過負荷(過去)の調査手順(1/3)



Performance Analyzerから、「インスタンス名」、「日付」、「時間」を指定して開きます。「OS指標」 を選択し、CPU使用率が高い時間でダブルクリックします。「プロセス」タブを選択し「CPU」順に ソートし、上位プロセスがDB関連プロセスかどうかを切り分けします。Oracle以外のプロセスの 場合はサーバ管理者に問い合わせします。Oracleプロセスの場合はPIDを確認します。



MaxGauge活用TIPs



活用TIPs: CPU過負荷(過去)の調査手順(2/3)



「セッション」タブを選択し、「CPU」でソートします。プロセスの「PID」とセッションの「SPID」が同一のものがあることを確認し、「SQL Text」を特定します。

	MaxGauge for O	Pracle	
	Detabase Perfor	mance Maximizer ファイル(E) ツール(E) オブション(Q) ウィンドヴ) ツリーの非表示(N) ヘルブ(H)
Image: 100 mm	IL) TIPS 10-17 22:00~23:00		_ 8 ×
Image: Solution of the solution		 	■ 新規ウィンドウ
Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: State (SD) Mark Care Course Image: SD) Course Im	■ 1 インスタンス	CPU アクティブ 論理読取 物理読取 SQL REDO ラッチ ロック フル・スキャン 索引スキャン L/C 総合待機	22:20:00 🗰 🕅 📢 🕪 🖬 📥
220000 220000 220000 220000 220000 220000 220000 220000 200000 200000 200000 200000 200000 200000 200000 200000 200000		80 - 60 - 40 - 20 -	€ CPU
1/2018/41/t: Image: State of Water State Sta		22:00:00 性能指揮 各級指揮 指煙リスト 比率指煙 OS指煙 ロック・ツリー ロック・リスト POツリー	23:00:00
12018/81/h: ••••••••••••••••••••••••••••••••••••			IO Wait
0700849:ft: 0 <td< td=""><td></td><td></td><td>User CPU</td></td<>			User CPU
100018397£ Image: File (x6) Wax Guyest Image: CitibusersWinyamamotoWDocumentsWing Image: CitibusersWinyamamotoWDocumentsWing Image: CitibusersWing Image: CitibusersWing Image: CitibusersWing Image: CititititibusersWing			Sys CPU
1ンスタンス名 050日付 0xcL 012-10-17 012-10-18 012-10-17 10 012 10 10 10 10 10 10 10 10 10 15 10 10 11 15 11 10 12 10 10 10 1	ログの格納先 ・・・ ・ C:¥Users¥hyamamoto¥Documents¥M ・ C:¥Program Files (x86)¥MaxGauge3¥	60 - 40 - 20 -	Free Memory(MB)
ORCL D1220210-17 D1220210-18 ビノバックグラウンド・セッション除外 間給 2012-10-17 22:00 は Imme SQL Text LReads PReads Block Executions Hard Parse Total Parse 「協給 2012-10-17 22:00 は Imme SQL Text LReads PReads Stot Secial //Sec Changes /Sec Count/Sec 「区間検索 1 100 05 100 15 200 25 300 35 140 145 50 155 59 N N N	インスタンス名 ログの日付	99- 98- 97-	Used Memory(%)
The second s	TIPS 2012-10-17	129932 7022	☑ バックグラウンド・セッション除外
	開始 2012-10-17 ▼ 22:00 ま 終了 2012-10-17 ▼ 23:00 ま 区間検索 性能トレンド ま 開入 比較分析 (+ -	Program Module User Name SPID CPU Elapsed Time SQL Text LReads PReads SID Serial LReads PReads File sqlplus@e SQL*Plus TEST 9545 43 1,057 SELECT NAME, CATEGORY, D 49,783,051 49,572,816 22 11 0 0 0 sqlplus@e SQL*Plus TEST 9546 46 1,056 SELECT NAME, CATEGORY, D 49,557,718 49,348,214 24 9 0 0 0	Lutions Hard Parse Total Parse Sec Count/Sec Count/Sec 0 0 0 0
	開く (1) 20121017 77		•





活用TIPs: CPU過負荷(過去)の調査手順(3/3)



「SQLリスト」タブを選択し検索後に「CPU(SUM)」順でソートします。 前項の「セッション」タブで確認されたSQLが、上位に存在することを確認します。

Max	Gauge for (Dracle																+	
	Database Per	formance Maximizer									ファイル(E)	ツール	(1) オフ	¹ ション(ロ)	ウィンドウ	- ツリーの非	表示(№)	ヘルプ(出	
🗐 (L) TIPS 10-17 2	2:00~23:00						1											_ 8	X
		■ 性能トレ	ンド 💷 トップsq	🔳 hyt	クセッション	■ 指標の	kh 🔳 squij	リスト 🔳 セッショ	シリスト	e /	ラメータ 🔳	チェンジ	・パラメータ	- 🖪 7	アラート			新規ウィント	50 F
■ 1ンスタンス		Ero	m · 22.00			TD -													
		共		riax <u>c</u>	cou T	. 10 :										* T			
		基本	lo: 23:00		SQL I	ext:					3					: 世 冬末言		検索	
		From Time	SQL Text	Elapsed (SUM)	Elapsed (AVG)	LReads	PReads	Block Scan	Row Fetch	Sort Rows	CPU ⊽ (SUM) ⊽	CPU (AVG)	Waiting (SUM)	Waiting (AVG)	Executions	Program	Module	Machir	
		22:30:00	SELECT NAME	1,199.2	79.947	65,547,257	65,272,260	1,055,639,303	0	337	965.3	64.353	233.9	15.593	15	sqlplus@e	. SQL*	em1.j.	
		22:20:00	SELECT NAME	1,197.3	59.865	67,868,077	67,583,292	1,349,783,048	0	345	958.2	4/.91	239.1	11.955	20	sqipius@e	. SQL*	em1.j.	
		22:10:00	SELECT NAME	909.25	69.942	34,747,053	34,602,441	0	1	173	459.45	35,342	449.8	34.6	13	sqipius@e	SQL	em1.j.	
		22:40:00	SELECT NAME	565.7	56.57	31.697.681	31,564,054	1,111,737,091	0	161	453.95	45.395	111.75	11.175	10	salplus@e	. SOL*	em1.i.	
		22:00:00	SELECT T.CLI	13.35	13.35	202	5	0	75	151	6.7	4.7	6.65	6.65	1	oracle@e	MMO	em1.j.	
		22:00:00	SELECT ATTR	4.35	4.35	174	0	0	58	0	3.6	3.6	75	0.75	1	orade@e	MMO	em1.j.	
		22:00:00	SELECT COU	4.1	4.1	92	0	0	8	46	3.05	3.05		-	1	orade@e	MMO	em1.j.	
		22:00:00	SELECT COU	4.15	4.15	75	2	0	13	1,163	2.95	_							
		23:00:00	begin dbsnmp	3.3	1.1	801	30	0	446	114	2.2	- 上	からち	;番目	までに	CPU時	間が	集中し	ており
ロガのお納牛	ÆE	22:00:00	INSERT /*+ A	4.95	4.95	508	5	0	95	75	2	- Ei	roade		roadel	± 乞い	- 64	ミンナ	います
C:¥Users¥hvama	moto¥Documents¥M	22:00:00	WITH MWS A	2.03	3.03	20	0	67 108 864	113	14	1.95	_ ''	reaus		reaus	530		·)] /]·	949
C:¥Program Files	(x86)¥MaxGauge3¥l	22:00:00	INSERT INTO	1.95	1.95	4.618	9	07,100,004	44	0	1.8	1.8	0.15	0.15	1	oracle@e	MMO	em1.i.	
		22:00:00	INSERT INTO	1.85	1.85	0	0	0	0	1,400	1.5	1.5	0.35	0.35	1	oracle@e	MMO	em1.j.	
		22:00:00	insert into wr	6.75	6.75	867	29	0	298	36	1.4	1.4	5.35	5.35	1	orade@e	MMO	em1.j.	
インスタンス名	ログの日付	22:00:00	insert into wr	1.55	1.55	211	4	0	33	25	1.4	1.4	0.15	0.15	1	orade@e	MMO	em1.j.	
ORCL	2012-10-17	22:10:00	DECLARE job	1.4	0.467	0	0	0	0	0	1.3	0.433	0.1	0.033	3	orade@e		em1.j. 🖣	
TIPS	2012-10-18	•	·					-										Þ	
		εοι τ						↓ 1		. .									
		SELEC	T NAME C	ATEGOR	RY DE	SCN FR	OM PROD	ULGT WHER	= 1.01	NE R(NAME)	= : B1	1						
問始 2012-10-17	22:00 :	02220																-	
28 Z 2012 10 13																			
■ 区明绘表	23:00																		
区間使案	I BBZ																		
注意ドレノト	・ (第1)																		
比較分析	÷ =																	7	
	開く	•																	-
				_	_	_					_				_	_	_		
<u>IUJAN</u>	(L) 20121017_	TIPS	1/204																



活用TIPs: CPU過負荷時の改善例



問題になっているSQL(以下)は、関数を使っていたため「NAME」カラムの索引が活用されなかった事例です。改善策としてファンクション索引を追加した結果、CPU使用率が正常になりました。 SELECT NAME, CATEGORY, DESCN FROM PRODUCT WHERE LOWER(NAME) = :B1





活用TIPs: CPU過負荷(過去)の調査手順



今回のTIPからは、以下の事を確認しました。

- ✓ OS観点のCPU負荷状況
- ✓ プロセス観点のCPU負荷状況
- ✓ 各SQLのCPU負荷状況
- ✓ ファンクション索引追加による改善結果



http://www.ex-em.co.jp/exem_labo/exemlabo_maxgauge_tips_index.html

<お問い合わせ>

日本エクセム株式会社

TEL:03-4530-9598

e-mail : service@ex-em.co.jp

DATABASE ARTIST GROUP http://www.ex-em.co.jp



