資料庫效能系列

資料庫效能測試與參數調校

Conrad Yang
TmaxSoft Taiwan

資料庫效能系列 - 資料庫效能測試與參數調校

透過資料庫效能壓測工具BenchmarkSQL 及 Tibero Performance Repository (TPR) 的效能分析功能，作為資料庫的資源分配及參數調校的基準。

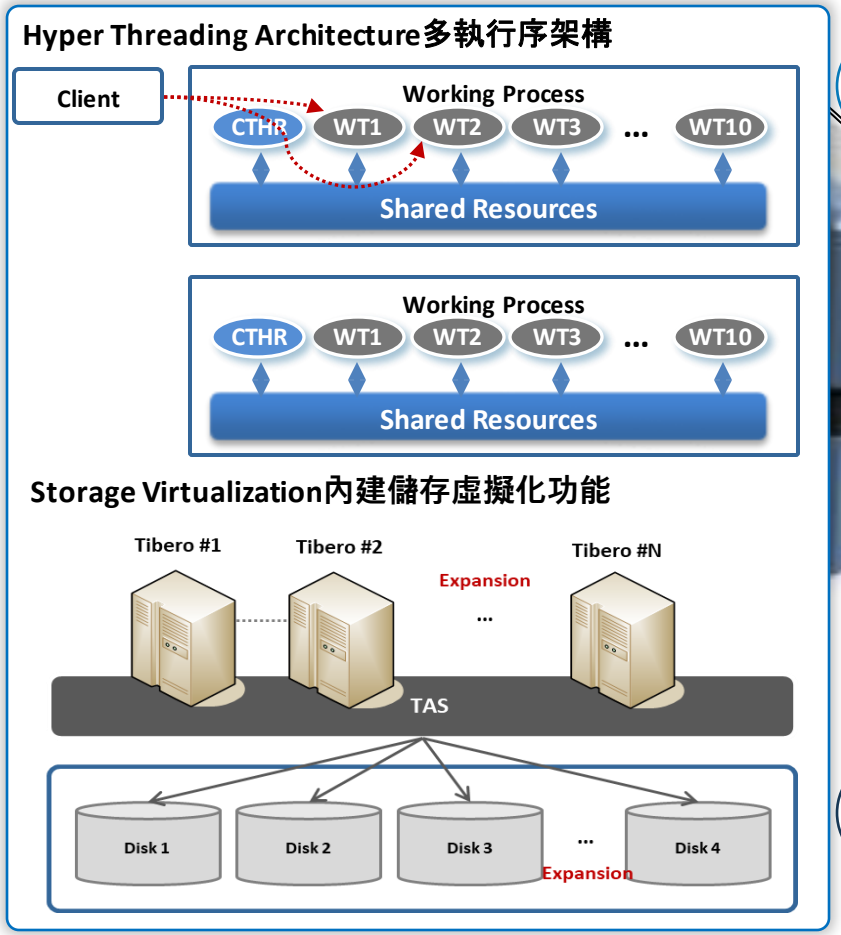
- BenchmarkSQL configuration and running
- TPR Report Analyse and Tibero Parameter Tuning
- Performance Verification

- *效能測試仍然應該以實際上執行的工作為主，使用BenchmarkSQL等壓測工具僅供作為初步的硬體驗證及資料庫基本參數調整*
- *使用額外的硬體資源建立與營運環境相當的測試環境，長期來看會是一個很好的投資*

Multi thread-based RDBMS optimized for large scale transaction
支援大量運算、採用多執行序架構並可橫向擴展的關聯式資料庫平台

TIBERO DataBase

- 高效能並支援橫向擴展
- 彈性的高穩定性及可靠度(共享磁碟雙活資料庫叢集與即時資料庫同步)
- 多執行緒架構，更有效運用硬體資源
- 高資料安全性(資安相關功能為標準功能不須額外選購)
- 相容ANSI SQL標準，並支援主流開發語言介面包含JDBC, ODBC
- 相容Oracle的資料型態、資料字典及PL/SQL、Pro*C
- 支援虛擬化部署(依據實際使用的處理器數目授權)
- 完整的開發及移轉工具，降低移轉人力與風險



High Performance

- High Performance for multi-user access & processing

Stability / Availability

- Active Cluster based on shared storage
- Various Backup & Recovery functions

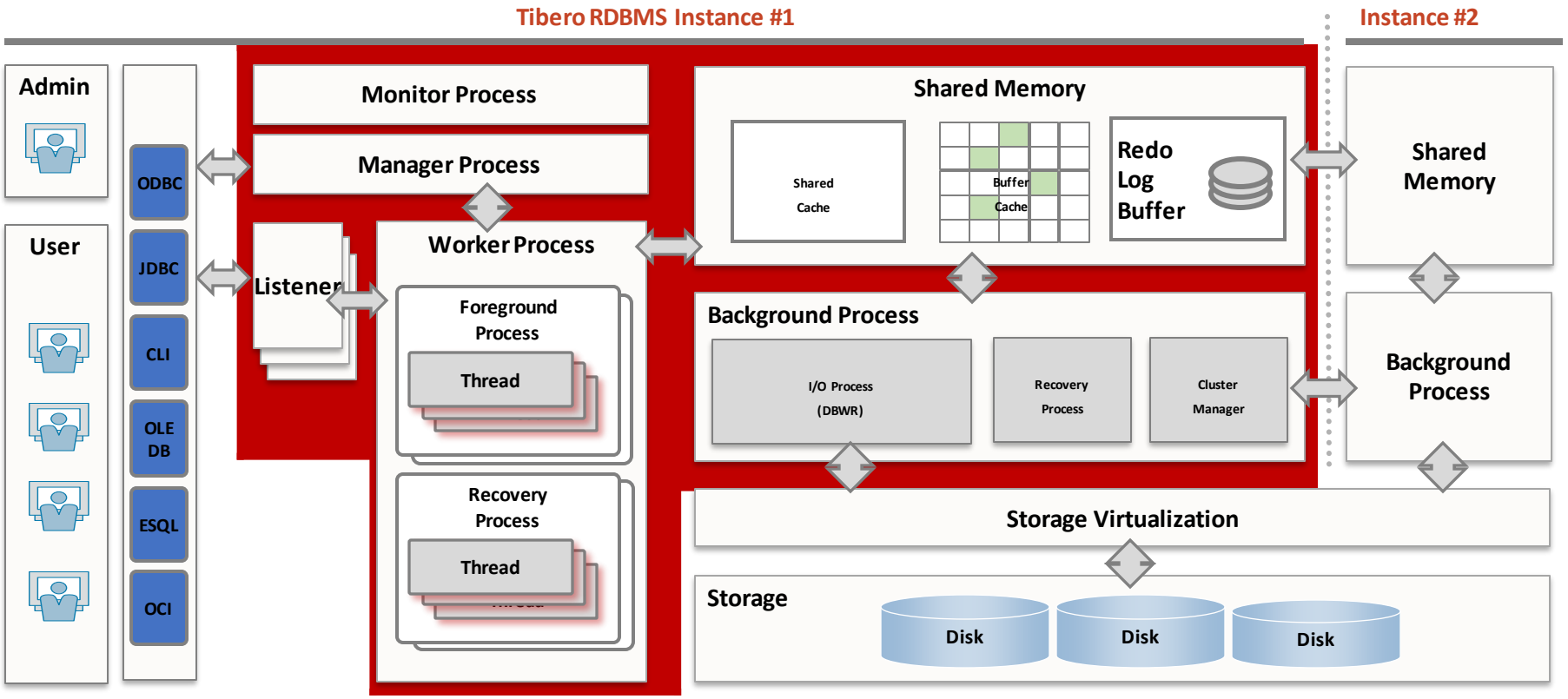
High Compatibility

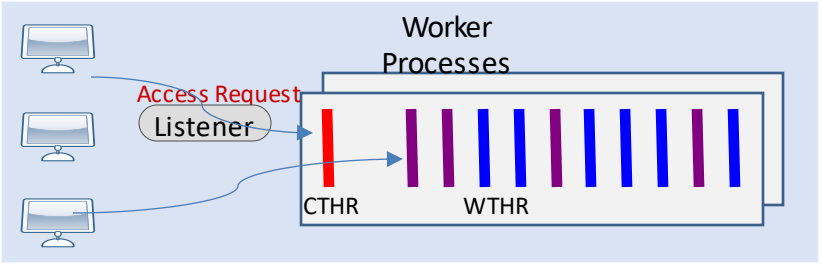
- Standard SQL & Interface supported
- Compatible with **PL/SQL**, Embedded SQL,
- Data Type, Function etc.

DevOps Tools

- Various tools for Admin.

Tibero Architecture – Multi-Thread & Multi-Process





CTHR (Control Thread)

- One CTHR is created per worker process. Configured number of WTHRs are created at server startup.
- Performs signal handling.
- Supports I/O Multiplexing, and relays messages in place of the working thread when needed.

WTHR (Worker Thread)

- Multiple working threads can be created per working process.
- Processes client's messages and returns the results.
- Handles most DBMS jobs, such as SQL parsing, optimization, and execution.

BenchmarkSQL

"An easy to use JDBC benchmark that closely resembles the TPC-C standard for OLTP. RDBMS's presently supported include PostgreSQL, EnterpriseDB and Oracle."

We are running the workload without think time and keying time to stress test the database.

Transaction	Description	Min. % of the total number of transactions	Min Mean of Think Time Distribution (seconds)	Min Keying Time (seconds)
New-Order	Initiates a new order.	No minimum	12	18
Payment	Updates the customer's balance and reflects the payment on the district and warehouse sales statistics.	43	12	3
Order-Status	Queries the status of a customer's last order.	4	10	2
Delivery	Processes a batch of 10 new orders, one for each district for a given warehouse.	4	5	2
Stock-Level	Counts the number of items in the last 20 orders in a district that fall below the stock threshold.	4	5	2

Table Name	Cardinality (in rows)	Typical [3] Row Length (in bytes)	Typical [3] Table Size (in 1,000 bytes)
WAREHOUSE	1	89	0.089
DISTRICT	10	95	0.950
CUSTOMER	30k	655	19,650
HISTORY [1]	30k	46	1,380
ORDER [4]	30k	24	720
NEW-ORDER [4]	9k	8	72
ORDER-LINE [4]	300k	54	16,200
STOCK	100k	306	30,600
ITEM [2]	100k	82	8,200

1 Small variations: subject to test execution as rows may be inserted and deleted by transaction activity from test executions.

2 Fixed cardinality: does not scale with number of warehouses.

3 Typical lengths and sizes given here are examples, not requirements, of what could result from an implementation (sizes do not include storage/ access overheads).

4 One percent (1%) variation in row cardinality is allowed to account for the random variation encountered during the initial data loading of the database.

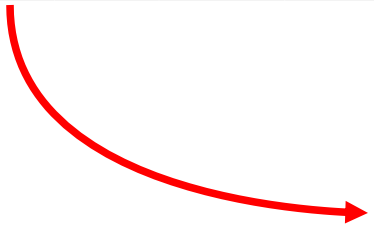
Note: The symbol "k" used in the cardinality column means one thousand

TPC-C OLTP Benchmark

Transaction	Frequency	Selects	Updates	Inserts	Deletes	Non-Unique Selects	Joins
New Order	**44**	**22**	**6**	**12**	**0**	**0**	**0**
Payment	**43**	**7.4**	**1**	**1**	**0**	**0.6**	**0**
Order Status	**5**	**9.6**	**8**	**0**	**0**	**0.6**	**0**
Delivery	**4**	**102**	**33**	**0**	**10**	**0**	**0**
Stock Level	**4**	**0**	**2**	**0**	**0**	**0**	**1**

In article in the Journal of Database Management ["Experimental Study of a Self-Tuning Algorithm for DBMS Buffer Pools" (2005, Vol. 16, pp. 1–20)] provided the workload used in the TPC-C OLTP (Transaction Processing Performance Council's Version C On-Line Transaction Processing) benchmark, which simulates a typical order entry application.

ID	Ware-house	terminals	tpmTOTAL	tpmC	Buffer Cache Hit	DB CACHE SIZE (GB)	TOTAL SHM SIZE (GB)	Huge-Pages	DBWR CNT	New Order	Payment	Order Status	Delivery	Stock Level
run_02	100	256	173,629.05	173,629.05	99.55	32	240	Y	1	43.48	43.48	4.35	4.35	4.35
run_03	100	256	128,414.05	128,414.05	99.44	32	240	N	1	43.48	43.48	4.35	4.35	4.35
run_25	100	600	290,571.51	290,571.51	99.84	8	24	Y	32	43.48	43.48	4.35	4.35	4.35
run_45	10000	384	230,612.50	230,612.50	69.03	8	256	Y	32	1.00	1.00	75.00	1.00	22.00
run_59	**10000**	**1200**	**529,473.47**	**529,473.47**	**99.59**	**512**	**640**	**Y**	**32**	**1.00**	**1.00**	**75.00**	**1.00**	**22.00**
run_60	**10000**	**1200**	**451,494.82**	**451,494.82**	**99.62**	**512**	**640**	**Y**	**32**	**45.00**	**43.00**	**4.00**	**4.00**	**4.00**
run_72	10000	1200	463,026.43	463,026.43	99.66	512	640	Y	64	43.48	43.48	4.35	4.35	4.35

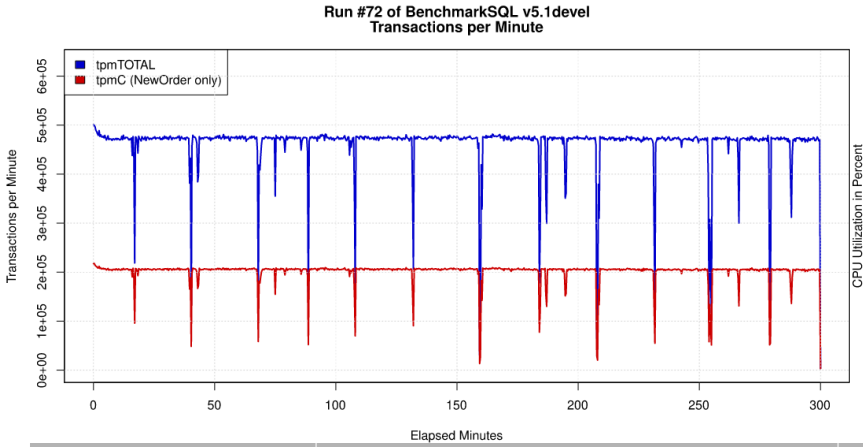


tx_name	tx_count (300 minutes)	tx_percent	tx_90th	tx_avg	tx_max	tx_limit	tx_rbk
NEW_ORDER	60,398,871	43.473%	0.495s	0.273s	47.061s	5	0.999%
PAYMENT	60,414,395	43.484%	0.103s	0.046s	46.317s	5	N/A
ORDER_STATUS	6,036,979	4.345%	0.048s	0.020s	1.976s	5	N/A
STOCK_LEVEL	6,040,779	4.348%	0.189s	0.098s	1.752s	20	N/A
DELIVERY	6,044,100	4.350%	0.000s	0.000s	0.123s	5	N/A
DELIVERY_BG	6,044,100	N/A	0.411s	0.265s	2.072s	80	N/A

BenchmarkSQL

This performance test was done with BenchmarkSQL 5.1, a 3rd Party Implementation of TPC-C test.

Warehouses: 10000, Terminals: 1200, RunTime: 300 minutes
Overall tpmTotal: 463117.08 Overall tpmC: 201329.57

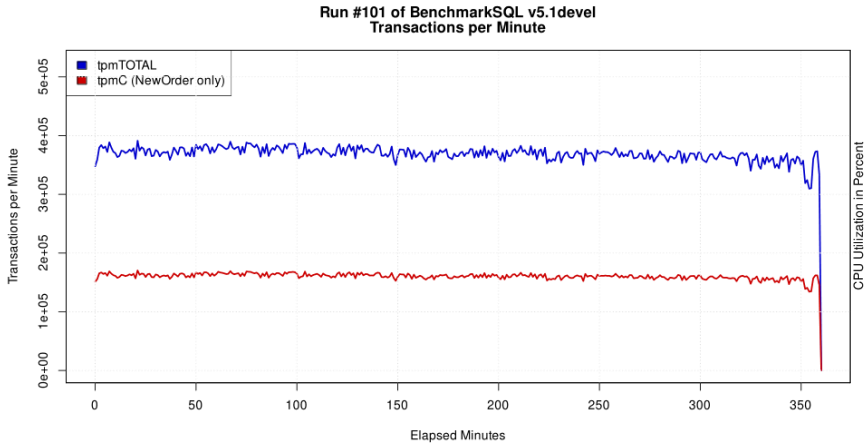


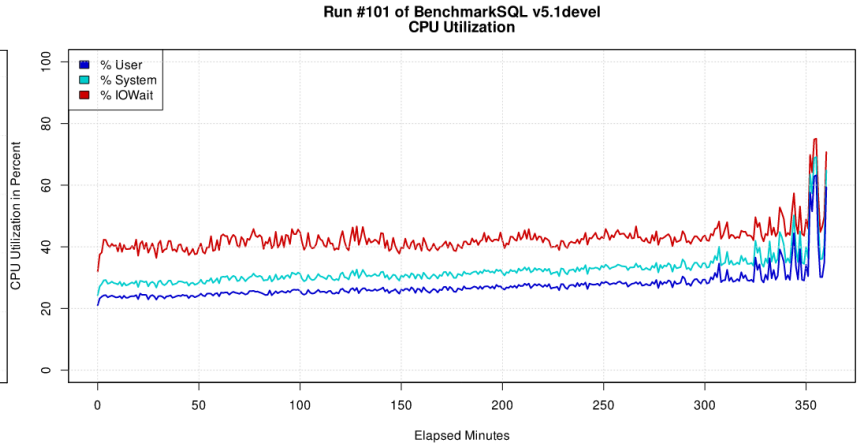
Transaction Type	Latency			Count	Percent	Rollback	Errors	Skipped Deliveries
	90th %	Avg	Max					
NEW_ORDER	0.495s	0.273s	47.061s	60398871	43.473%	0.999%	0	N/A
PAYMENT	0.103s	0.046s	46.317s	60414395	43.484%	N/A	0	N/A
ORDER_STATUS	0.048s	0.020s	1.976s	6036979	4.345%	N/A	0	N/A
STOCK_LEVEL	0.189s	0.098s	1.752s	6040779	4.348%	N/A	0	N/A
DELIVERY	0.000s	0.000s	0.123s	6044100	4.350%	N/A	0	N/A
DELIVERY_BG	0.411s	0.265s	2.072s	6044100	N/A	N/A	0	0

BenchmarkSQL

This performance test was done with BenchmarkSQL 5.1, a 3rd Party Implementation of TPC-C test.

Warehouses: 1000, Terminals: 180, RunTime: 360 minutes.
Overall tpmTotal: 368993.56 Overall tpmC: 160421.13





Transaction Type	Latency			Count	Percent	Rollback	Errors	Skipped Deliveries
	90th %	Avg	Max					
NEW_ORDER	0.038s	0.020s	1.716s	57751607	43.475%	1.001%	0	N/A
PAYMENT	0.054s	0.033s	1.774s	57757789	43.480%	N/A	0	N/A
ORDER_STATUS	0.123s	0.051s	2.584s	5775439	4.348%	N/A	0	N/A
STOCK_LEVEL	0.065s	0.035s	1.699s	5778023	4.350%	N/A	0	N/A
DELIVERY	0.000s	0.000s	0.028s	5774825	4.347%	N/A	0	N/A
DELIVERY_BG	0.090s	0.055s	1.802s	5774825	N/A	N/A	0	0

TIP - Tibero Initialization Parameters

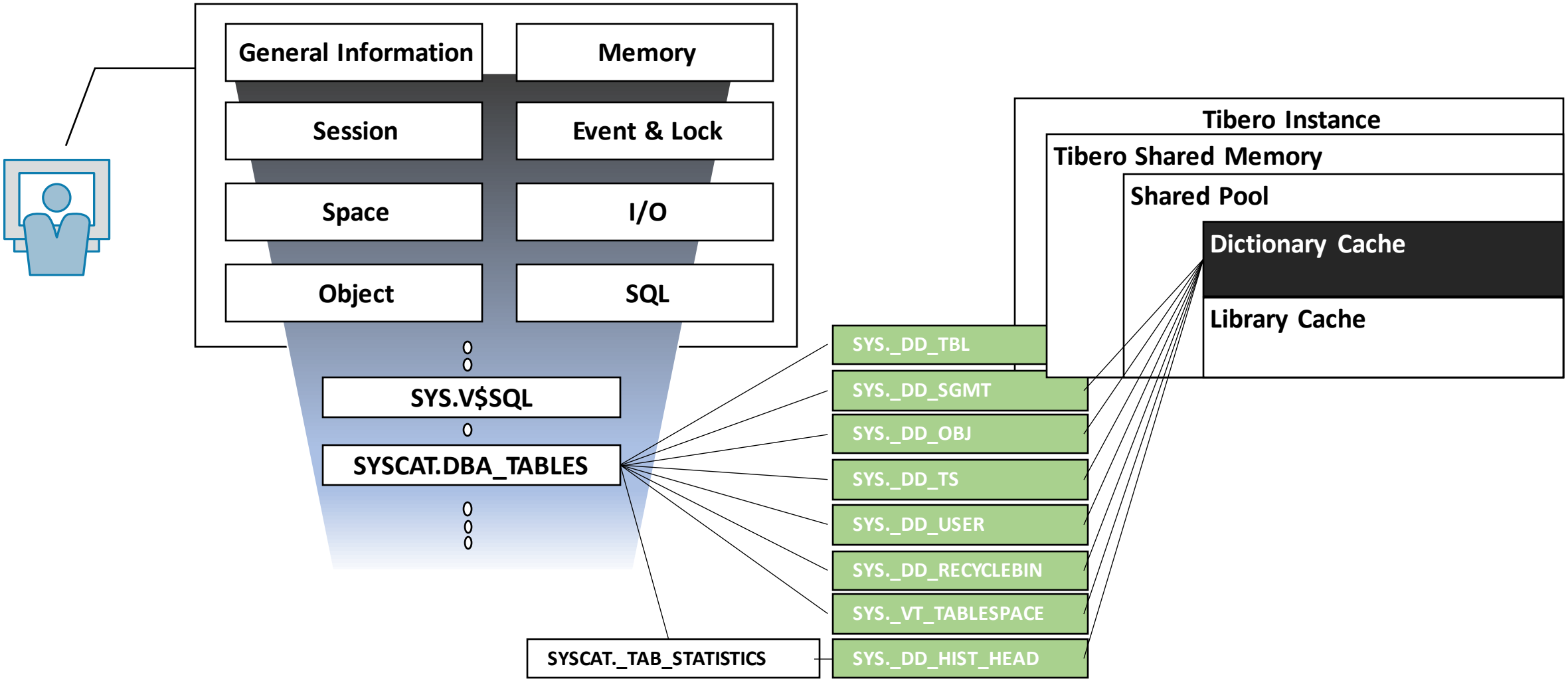
File Location: $TB_HOME/config/$TB_SID.tip

```
SQL> create btip from tip;
SQL> alter system set CURSOR_SHARING=FORCE scope=btip; -- scope=both;
SQL> select * from vt_parameter where name like 'CURSOR_SHARING%';

NAME           TYPE   DFLT_VAL IS_OPTIONAL IS_ADJUSTABLE IS_DYNAMIC  IS_HIDDEN BOUND GLOBAL VALUE STATUS       RANGE
-------------- ------ -------- ----------- ------------- ---------- ---------- ----- ------ ----- ------------ -------------
CURSOR_SHARING STRING EXACT              1             1          1          0 SESS  BOTH   FORCE CMD_MODIFIED EXACT | FORCE
```

Group	Class	Description
Static/Dynamic	Static	No dynamic changes
	Dynamic	Dynamic changes
Persistent/Adjustable	Persistent	No changes since DB creation
	Adjustable	Changeable after DB creation
Mandatory/Optional	Mandatory	Start an instance only when defined in the parameter file.
	Optional	Apply the default value unconditionally.
System/Session	System	Same value for all sessions
	Session	Different values per session
Global	LOCAL	Different values for per instance
	GLOBAL	Same value for all instances
	BOTH	Allow both LOCAL and GLOBAL instance settings.

Monitoring Tibero



Monitoring Tibero – Data Dictionary Views

Dynamic Views

View	Description
V$SESSION	Displays session information
V$SESSION_WAIT	Displays the current wait for each session.
V$SESSTAT	Displays session statistics.
V$SESS_TIME_MODEL	Displays session-cumulative times.
V$SYSSTAT	Displays system statistics.
V$SYSTEM_EVENT	Displays information on total wait time for the event.
V$SYS_TIME_MODEL	Displays system-wide cumulative times.
V$PROCESS	Displays server process information.
V$SQLAREA	Displays parent cursor information.
V$SQL	Displays child cursor information.

Monitoring Tibero – Data Dictionary Views

Dynamic Views

View	Description
V$SQL_PLAN	Displays execution plans per cursor.
V$SQL_PLAN_STATISTICS	Displays the statistics of each execution plan.
V$SQL_BIND_CAPTURE	Displays information on bind variables of each cursor.
V$EVENT_NAME	Displays information on wait event.
V$LATCH	Displays cumulative latch statistics.
V$LOCK	Displays locks currently held by Tibero and wait information for the lock.
V$ROWCACHE	Displays cumulative statistics on dictionary cache.
V$LIBRARYCACHE	Displays cumulative statistics on library cache.
V$OSSTAT	Displays system utilization statistics from the OS.
V$BH	Displays buffer header information.

Buffer Hit Ratio

- Shows the hit ratio of the database buffer (buffer cache) accessed by the operations after DB startup.
- The ratio increases when the buffer is accessed by memory while it decreases when the buffer is accessed by a data file. Low buffer hits degrade system performance due to increased disk I/O.
- Improvement Method: Tuned to avoid performing full memory scans by increasing memory allocated to the buffer cache.
- Time: Time when the object is viewed.
- Physical read: Number of times data blocks has been read from the data file.
- Logical read: Number of times data blocks has been read from the memory.
- Hit ratio formula: 1 (Physical reads/ Logical reads) * 100
- Status: Hit > 90, then 'Good'

  70 <= Hit <= 90 then 'Average'

  Hit < 70 'Not Good'

Time	Physical read	Logical read	Hit	Status
2020/05/26 10:10:12	611	25819	97.63	Good

Current Session Info

- Sid, Serial: Session ID. Used to perform a kill session or dump command.
- Username: Connected user name.
- Status:

  READY: The session is connected, but it is idle.

  RUNNING: The session is connected and running an SQL.

  TX_RECOVERING: The transaction is being recovered from an abnormal termination of the database.

  SESS_RECOVERING: The session is being recovered from user's cancellation, along with it's transactions (e.g: CTRL + C or alter system kill session).

  ASSIGNED: WTHR was assigned to connect a session, but no connection has not been established between the WTRH and client. This state is just before or immediately after the session connects to the WTRH. When the connection is established, the state becomes active or running.
- IPaddr: IP to which the user connects.
- Logon_Time: When the session is connected to the database.
- Program: Type of the client program.
- Wlock_Wait: WLOCK type for the session.
- SQL_ID: The latest SQL_ID or CHILD_NUMBER for the session.
- Wthr_Pid: Working thread process to which a session belongs. Same as PID of OS command 'ps'.
- PGA(MB): The usage of a PGA memory area of the session.

Sid,Serial	Username	Status	IPaddr	Logon_Time	Program	Wlock_Wait	SQL_ID	Wthr_Pid	PGA(MB)
64,174	TIBERO	READY	127.0.0.1	20/05/26 10:07:16	tbsql		ch6ffb5v04t1y/67	3761	.051
65,230	TIBERO1	RUNNING	127.0.0.1	20/05/26 10:07:42	tbsql	WLOCK_TX	0tb13yw44xfvg/68	3761	.056
66,284	SCOTT	RUNNING	127.0.0.1	20/05/26 10:08:05	tbsql	WLOCK_TX	0d43knnkgw98p/70	3761	.056

Major Wait Events

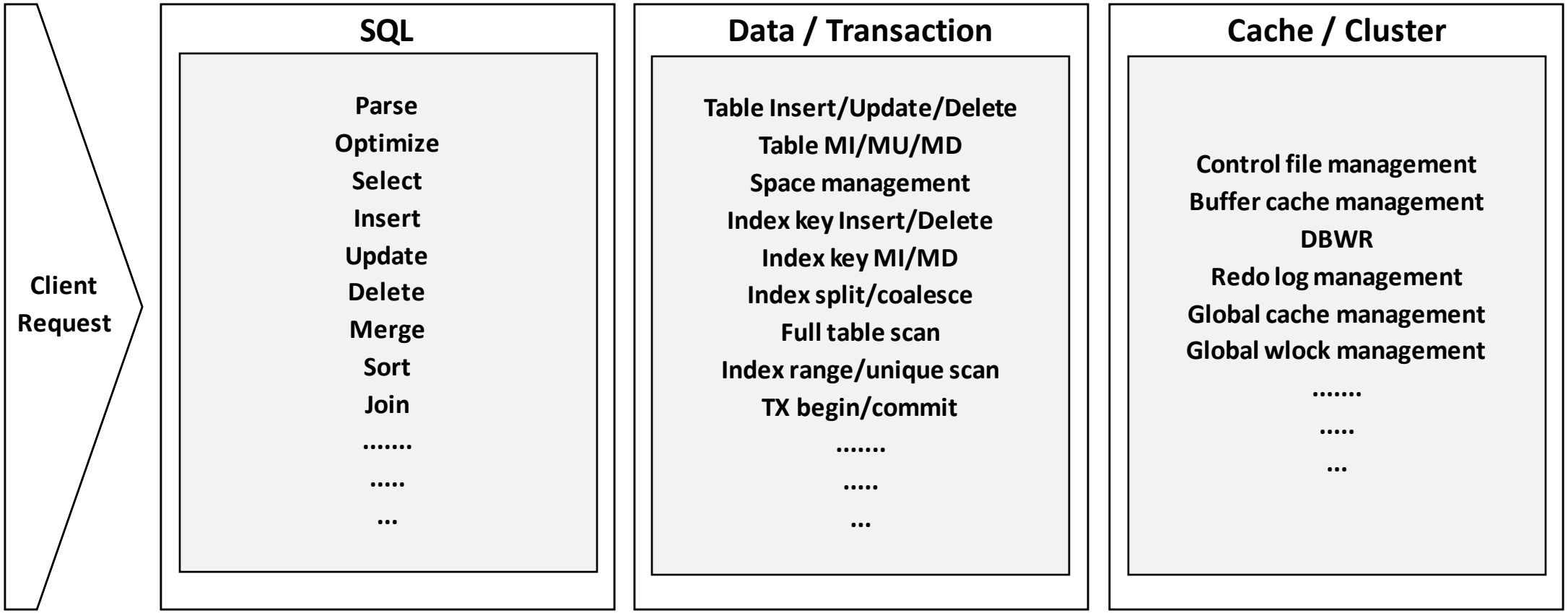
- **WE_LOG_FLUSH_COMMIT - log flush wait for commit**
  - **Occurs when LGWR writes all log buffer data to log files for user's request of COMMIT.**
  - **▷ Disk I/O performance is low in the volume including redo log files**
  - **Use a volume with the best performance only for redo log files.**
  - **▷ The redo log buffer size is too large**
  - **If a redo log buffer size is too large in a system where a large volume of redo log occurs, this event can occur. Decrease the buffer size if the "log flush wait for space" event does not occur.**
  - **▷ Unnecessary commit occurs frequently**
  - **Analyze transactions to decrease commit count. If sequences are used, use a cache option because a data dictionary is committed unless the option is used.**

Major Wait Events

- WE_SPIN_TOTAL_WAIT (spinlock total wait)
  - Waiting time to get a spinlock
- WE_BUF_WAIT (buf pin wait)
  - Ranked high when multiple sessions try to handle one block.
- WE_BUF_FREE (wait for free buf)
  - Cache is too small, or DB writer does not write dirty blocks to a disk fast.
- WE_SPIN_BUF_BUCKET (spinlock: cache buffers chains)
  - This event occurs when multiple sessions compete for one spinlock.
- WE_SPIN_BUF_WS (spinlock: cache buffers lru chain)
  - This event occurs when multiple sessions compete for one spinlock.

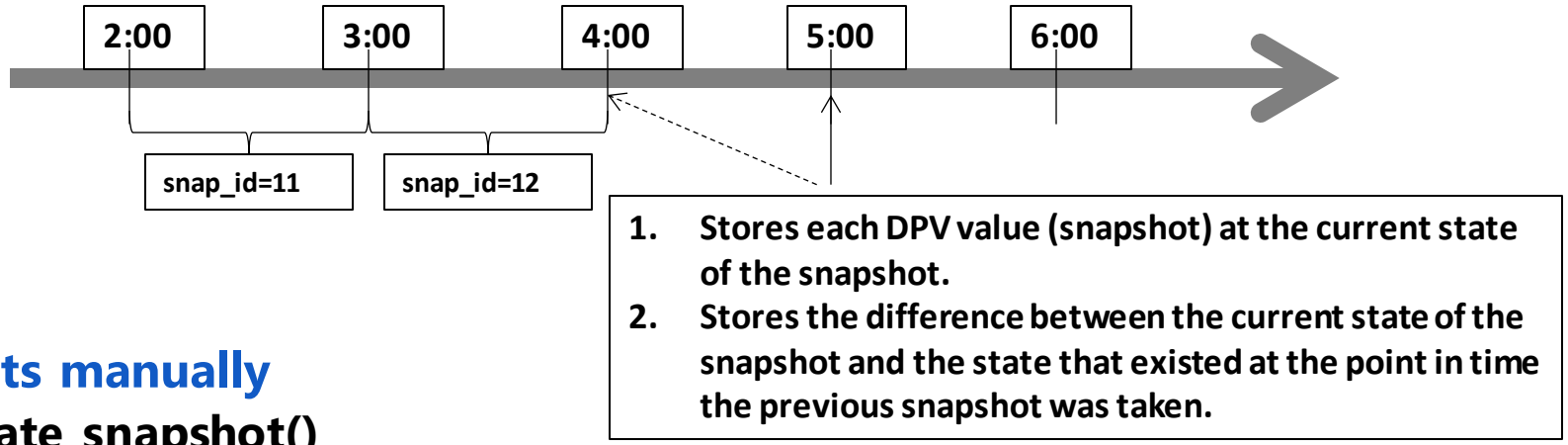
TPR - Tibero Performance Repository

Data Processing Procedure



TPR - Tibero Performance Repository

- **Stores snapshot of system statistics every hour.**



- **Creating snapshots manually**
  - **dbms_tpr.create_snapshot()**

- **Creating a performance analysis report**
  - e.g.) Analyze the load condition of the past day.
    dbms_tpr.report_text (sysdate-1, sysdate)

  - Report is generated with the name '$TB_HOME/instance/$TB_SID/tpr_report -'.

TPR - Tibero Performance Repository

Section	Object	Metrics
Overview Part	System Overview	• CPU Usage • Memory Usage
	Workload Overview	• Workload Summary • Workload Stats
	Instance Overview	• Instance Efficiency • TAC Statistics Overview (Cluster Cache Activity, Cluster Buffer Cache, Cluster Cache and Wait Lock Statistics) • Top 5 Wait Events by Wait Time • I/O Overview
	SQL Overview	• PGA Work Area Statistics • Top 3 SQLs Ordered by Elapsed Time • Top 3 SQLs Ordered by Executions • Top 3 SQLs Ordered by Gets

TPR - Tibero Performance Repository

Section	Object	Metrics
Detail Part	System Details	• OS Statistics • Shared Pool Statistics • Physical Plan Cache Statistics • Data Dictionary Cache Statistics • PGA Statistics
	Workload Details	• Workload Stats (Time-based) • Workload Stats (Number-based) • Workload Stats (Size-based)
	Instance Details	• Buffer Cache Statistics • Wait Event Summary (by Class) • Wait Events by Wait Time • Session Status with Wait Event • Blocking Session Status with Wait Event • Wlock Statistics • Spinlock (Latch) Statistics • Spinlock (Latch) Sleep Statistics • Tablespace I/O Statistics • File I/O Statistics • Temp Segment Usage Statistics

TPR - Tibero Performance Repository

Section	Object	Metrics
Detail Part	Instance Details	• Segments Ordered by Physical Reads ("TPR_SEGMENT_STATISTICS=Y") • Segments Ordered by Logical Reads ("TPR_SEGMENT_STATISTICS=Y") • Segments Ordered by ITL Waits ("TPR_SEGMENT_STATISTICS=Y") • Segments Ordered by Buffer Busy Waits ("TPR_SEGMENT_STATISTICS=Y") • Segments Ordered by Row Lock Waits ("TPR_SEGMENT_STATISTICS=Y") • Undo Statistics • Wait Statistics ("_DB_BLOCK_PIN_WAIT_USE_STAT=Y")
	SQL Details	• PGA Summary • PGA Work Area Histogram • SQLs Ordered by Elapsed Time (with Physical Plan) • SQLs Ordered by Elapsed Time/Execution (with Physical Plan) • SQLs Ordered by Executions (with Physical Plan) • SQLs Ordered by Gets (with Physical Plan) • SQLs Ordered by Reads (with Physical Plan) • SQLs Ordered by Extra I/O (with Physical Plan) • SQLs Ordered by CPU (with Physical Plan)
	Etc.	• Tibero Init. Parameters (.tip) • Modified Parameters

TmaxData

Thank You