

# Sherpa Oracle v5

셀파오라클 v5



# AGENDA

## I. 회사개요

## II. 셀파 특징점

1. 아키텍처의 우수성
2. 성능의 우수성
3. 주요기능
4. WAS-DB 통합 모니터링

## III. 비교자료

## IV. 주요 원백 사례

## V. 수집서버 권장 사양





# I . 회사개요

# I. 회사개요

“ 고객의 성공적인 비즈니스를 위해  
최고의 데이터베이스 성능 모니터링 솔루션을 개발, 공급합니다 ”



9 가지

다양한 DBMS  
성능관리 솔루션  
제공

Sherpa SAP HANA  
Sherpa Oracle  
Sherpa SQL Server  
Sherpa MySQL  
Sherpa MariaDB  
Sherpa PostgreSQL  
Sherpa TIBERO  
Sherpa AltiBase  
Sherpa Cloud

200 여  
고객사에서  
안정성과 기술력을  
인정받은 성공사례  
보유

SEC(Sherpa SAP HANA)  
LG Group  
HKMC Group etc.

10,000 여개 DB  
시스템에서  
성능 모니터링 중

ERP, MES, SCM, HR, SVC,  
Banking Core,  
Telco Core,  
Air etc.

업력 15 년  
이상  
데이터베이스  
전문기업

Founded 2006

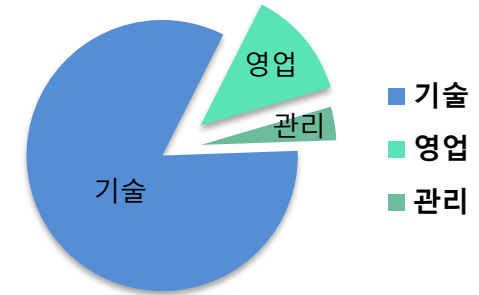
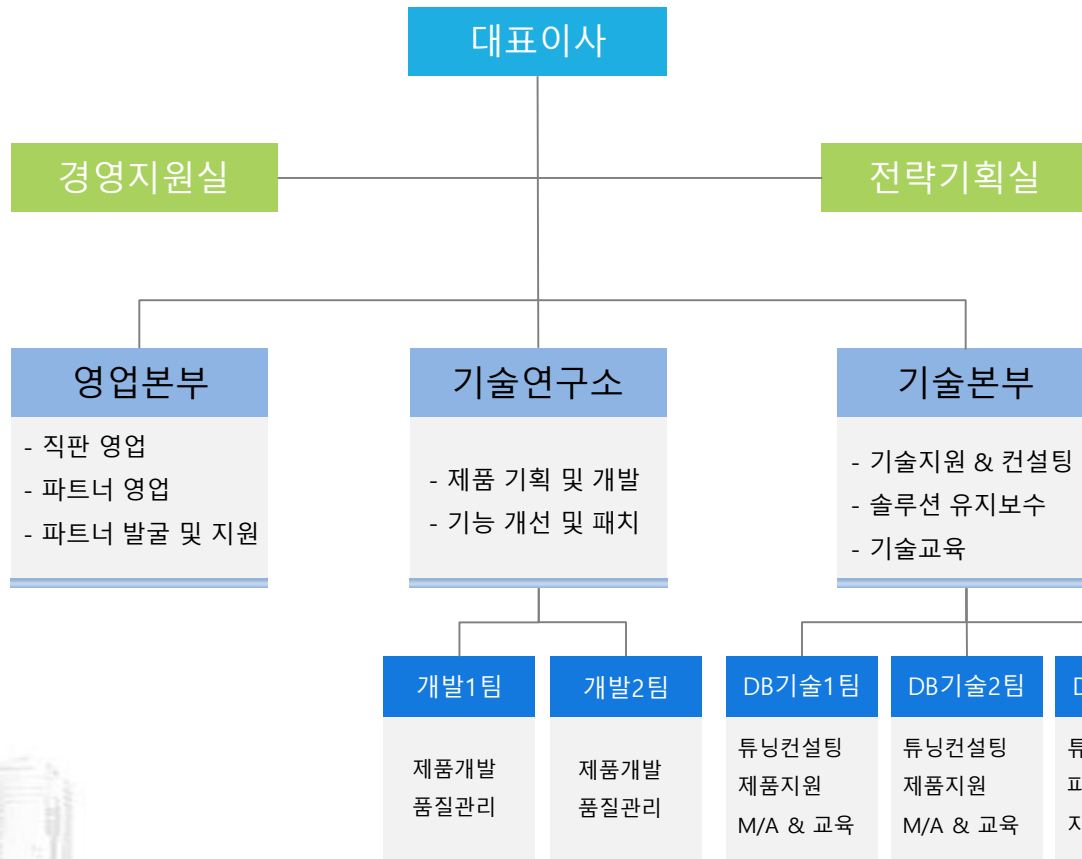
History

Product

Customer

Installed System

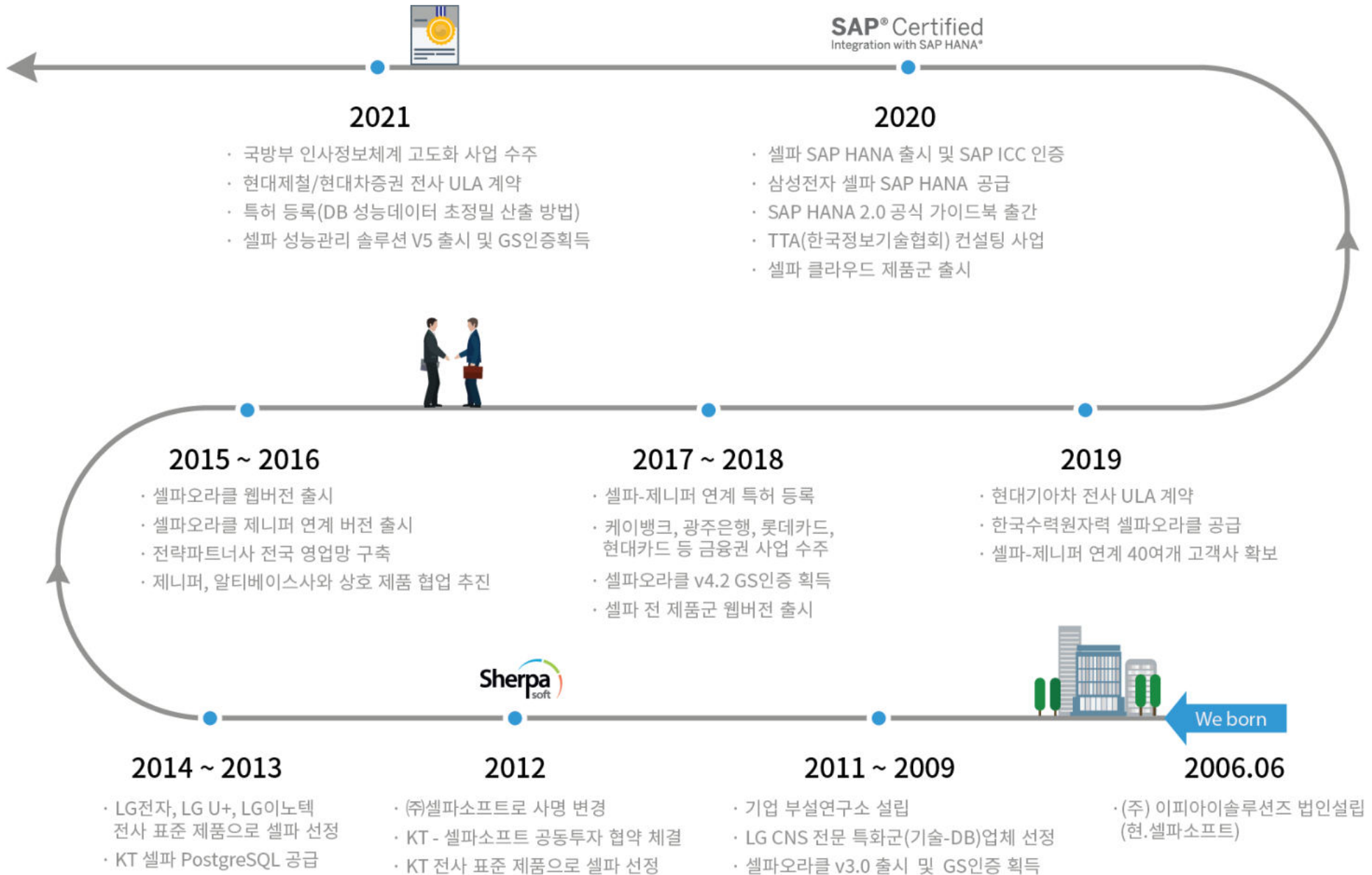
## 2. 조직현황



구분	특급	고급	중급	초급	계
기술연구소	3	3	3	12	20
기술본부	4	2	4	9	19
영업 / 지원	4	1	1	1	8
계	11	6	8	20	47



# 3. 연혁



“데이터베이스 성능관리 노하우를 바탕으로  
최고의 솔루션과 새로운 서비스를 고객에게 제공합니다”



## 5. 셸파 제품 포트폴리오

“셸파”는 **오라클, SAP HANA, SQL Server, Tiberio, Altibase, PostgreSQL, MariaDB, MySQL** 등 다양한 상용 DBMS의 성능 모니터링 솔루션으로, 귀사의 IT인프라를 pro-active하게 관리 합니다.





# 6. 고객사 (금융, 제조, 통신)

2022.06월 기준

금융



제조 / 통신



# 6. 고객사 (공공)

2022.06월 기준

10  
F10

# 6. 고객사 (대학, 병원, 서비스, 유통, 기타)

2022.06월 기준

대학 / 병원



서비스 / 유통 / 기타



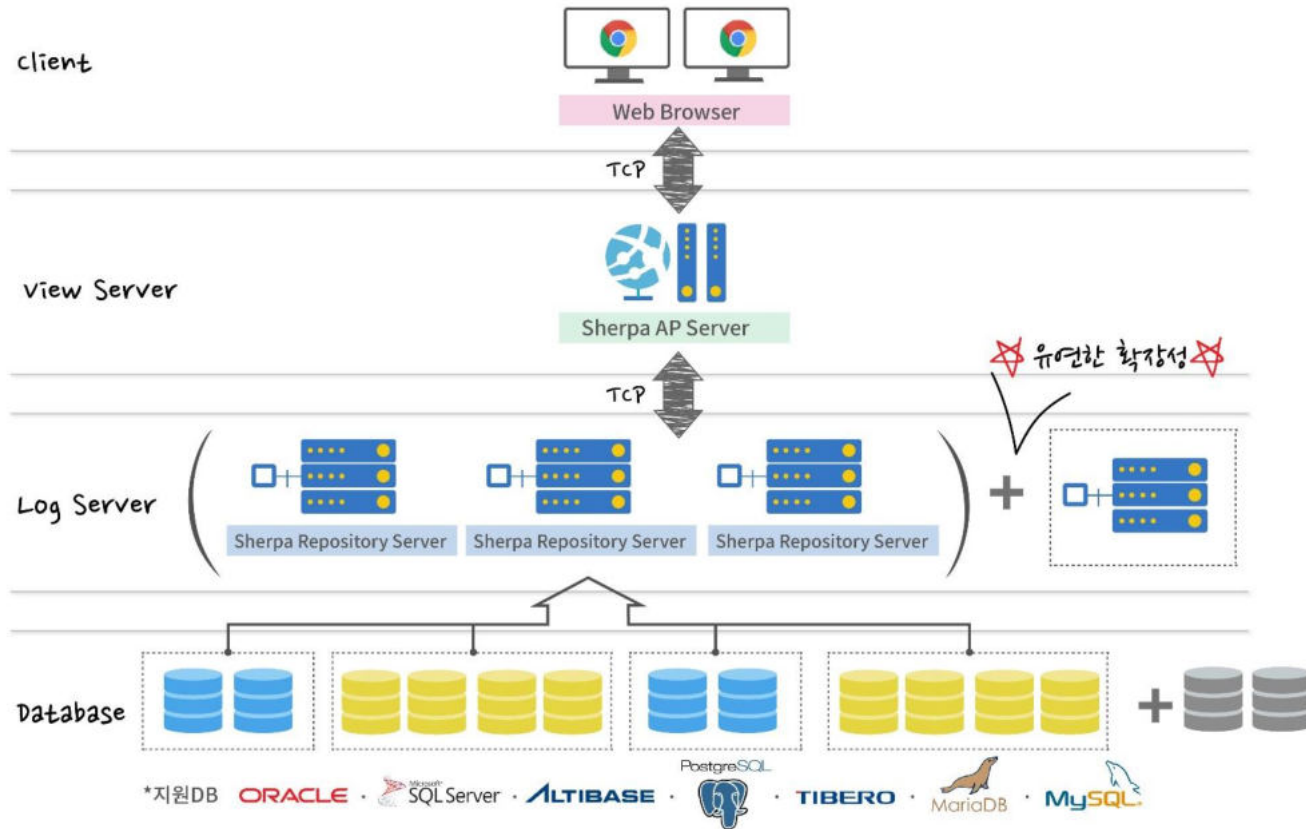


## II. 셀파 특징점

1. 아키텍처의 우수성
2. 성능의 우수성
3. 주요기능
4. WAS-DB 통합 모니터링

# 1. 아키텍처의 우수성

“모니터링 및 성능데이터 수집 시 **시스템 자원 사용 최소화**”

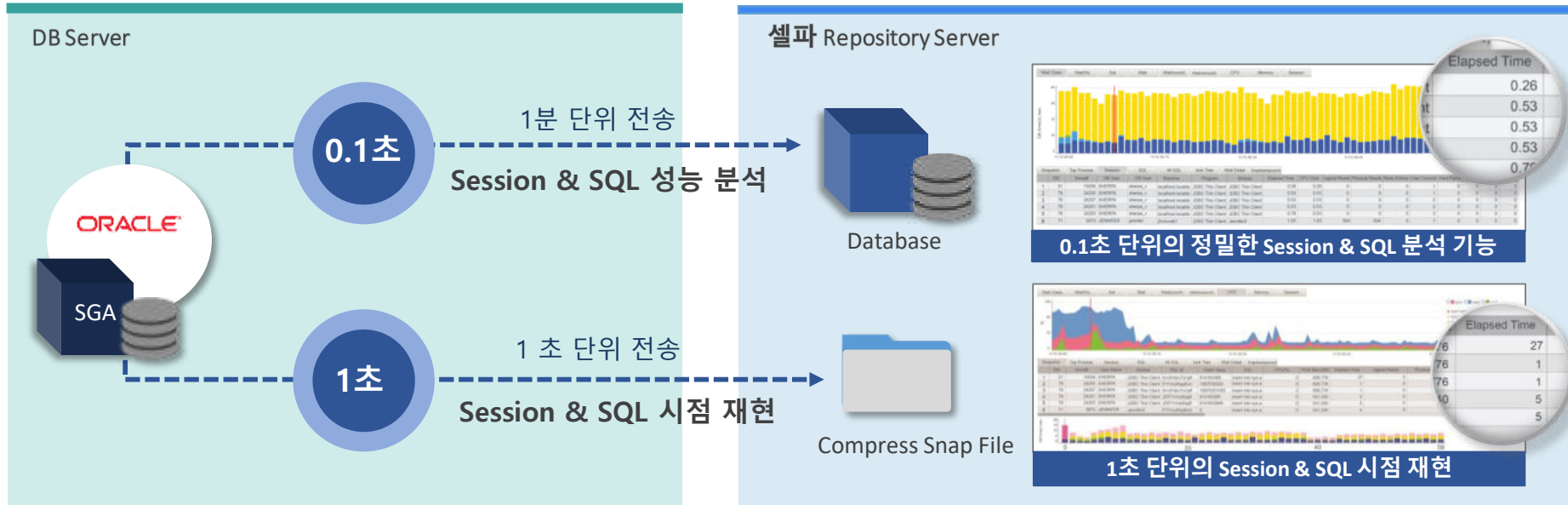


시스템 자원 사용 최소화를 위해 수집 성능 정보를 I/O 대기 없이 곧바로 단일 Repository로 구성된 수집서버로 전송하여 관리

# 1. 아키텍처의 우수성(계속)

“DMA(Direct Memory Access) 아키텍처로 부하 없이 정밀한 성능 Data 수집”

“국내외 경쟁 제품 중 가장 정밀하게 수집하며 실시간 수준의 분석 가능 ”



## 수집 Data

- 1초 단위 Session, SQL 정보
- 0.1초 단위 Session, SQL 성능 정보(국내외 유일)

**특허출원**  
(DB 성능데이터 초정밀 산출)

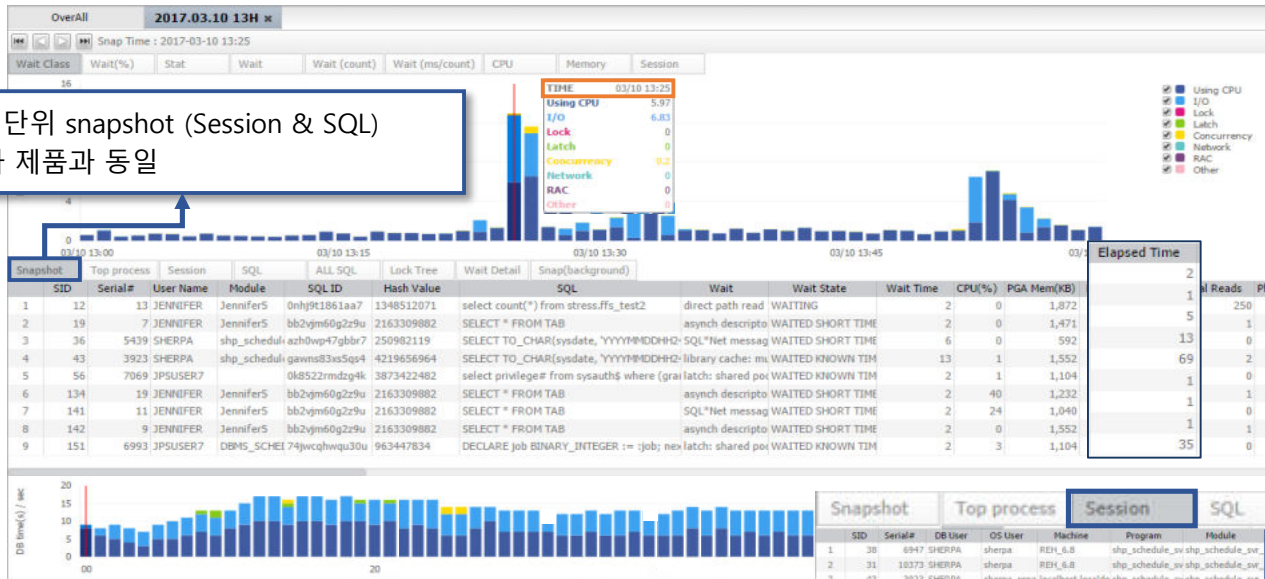
## 분석 기능

- 1초 단위로 전송하여 부하 시점을 실시간 수준으로 분석 가능
- 0.1초 단위의 수집 데이터를 1분 단위로 집계하여 정밀한 분석 가능

# 1. 아키텍처의 우수성(계속)

“최대 0.01초(Default 0.1초) 단위로 세션 및 SQL 정보를 정밀하게 수집”

- 1초 단위 snapshot (Session & SQL)
- 타사 제품과 동일



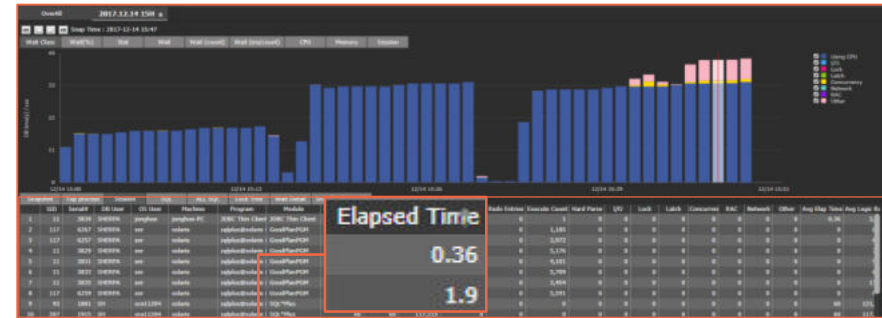
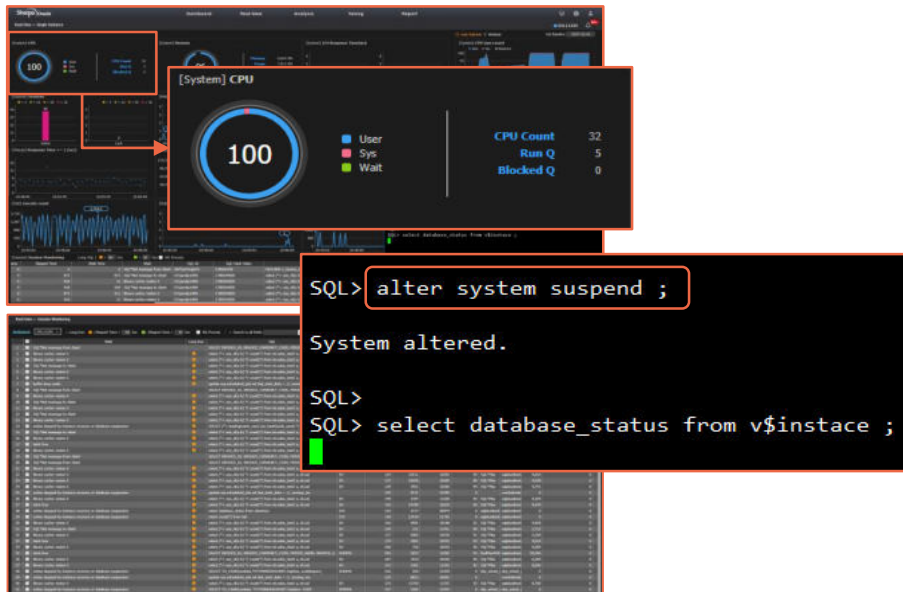
- 0.1초 단위 Session & SQL 정보를 1분 단위로 Summary하여 Repository 서버로 전송
- 타사 제품 대비 강력한 성능 우위 (원백 사례를 통해 검증)

SID	Serial#	DB User	OS User	Machine	Program	Module	ALL SQL	Logical Reads	Physical Reads	Redo Entries	Execute Count	Hard Parse	I/O	Lock	Latch	Concurr
1	38	SHERPA	sherpa	REH_6.8	shp_schedule_sv	shp_schedule_sv	0.22	0	0	0	0	0	0	0	0	0
2	31	SHERPA	sherpa	REH_6.8	shp_schedule_sv	shp_schedule_sv	0.22	238	6	0	21	0	0	0	0	0
3	43	SHERPA	sherpa_rena	localhost.localds	shp_schedule_sv	shp_schedule_sv	0.22	0	0	0	1	0	0	0	0	0.43
4	56	JPSUSER7	ora11202	EPL_AIX	oraed@EPL_AIX	DBMS_SCHEDULER	0.43	0	0	0	1	0	0	0	0	0.65
5	151	SYS	ora11202	EPL_AIX	oraed@EPL_AIX	DBMS_SCHEDULER	0.65	0	0	0	3	0	1	0	0	1.08
6	151	JPSUSER7	ora11202	EPL_AIX	oraed@EPL_AIX	DBMS_SCHEDULER	0.65	19,561	19,815	0	1	0	0	60	0	0
7	57	SHERPA	sherpa	REH_6.8	JOBCL Thin Client	JOBCL Thin Client	0.65	0	0	0	1	0	0	0	0	9.93
8	142	JENNIFER	jennifer	JENNIFER	Thin Client	JENNIFER	0.65	0	0	0	1	0	0	0	0	0
9	145	SHERPA	sherpa	REH_6.8	JOBCL Thin Client	JOBCL Thin Client	3.02	21,200	21,387	0	2	0	50	0	0	0
10	143	JENNIFER	jennifer	JENNIFER	Thin Client	JENNIFER	0.22	0	0	0	1	0	0	0	0	0
11	186	SHERPA	sherpa	REH_6.8	shp_schedule_sv	shp_schedule_sv	0.22	0	0	0	1	0	0	0	0	0
12	36	SHERPA	sherpa_rena	localhost.localds	shp_schedule_sv	shp_schedule_sv	0.22	0	0	0	1	0	0	0	0	0
13	19	JENNIFER	jennifer	JENNIFER	Thin Client	JENNIFER	54.17	19,561	19,815	0	1	0	60	0	0	0
14	12	JENNIFER	jennifer	JENNIFER	Thin Client	JENNIFER	0.22	19,497	19,751	0	0	0	60	0	0	0
15	184	SHERPA	sherpa_rena	localhost.localds	shp_schedule_sv	shp_schedule_sv	0.22	60	60	0	1	0	0	0	0	0
16	64	SHERPA	sherpa	REH_6.8	shp_schedule_sv	shp_schedule_sv	0.22	60	60	0	1	0	0	0	0	0

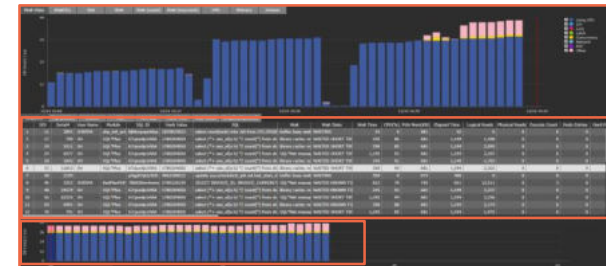
SID	Hash Value	SQL ID	SQL Text	Elapsed Time	CPU Time	lock	Latch	Concurr	Network	RAC	Other	Logical Reads	Physical Reads	Redo Entries	Execute Count	Hard
1	31	4079814962	fydhdmhu29k	0.22	0.22	0	0	0	0	0	0	0	0	0	0	0
2	141	2163309882	bb2vjn60g2z9u	0.22	0.22	0	0	0	0	0	0	0	0	0	0	0
3	151	144723942	9p9p94w0mc6	0.22	0.22	0	0	0	0	0	0	0	0	0	0	0
4	17	3346500003	yk15fg3r2d3	0.22	0.22	0	0	0	0	0	0	0	0	0	0	0
5	38	3346500003	yk15fg3r2d3	0.22	0.22	0	0	0	0	0	0	0	0	0	0	0
6	56	1897638902	zfvn9p9j9k	0.22	0.22	0	0	0	0	0	0	0	0	0	0	0
7	56	159997941	4yb4104skwq	0.22	0.22	0	0	0	0.43	0	0	0	0	0	0	0
8	43	4219656964	gavns83a5q4	0.22	0	0	0	0	0	0	0	238	6	0	21	0
9	151	159997941	4yb4104skwq	0.22	0.22	0	0	0	0.65	0	0	0	0	0	0	0
10	151	159997941	4yb4104skwq	0.22	0.22	0	0	0	1.08	0	0	0	0	0	0	0
11	151	815042050	1k216h99392	0.43	0	0	0	0	0	0	0	0	0	0	0	0
12	143	2163309882	bb2vjn60g2z9u	0.43	0.22	0	0	0	0	0	0	0	0	0	0	0
13	57	1326525299	70zdw9j2avm	0.43	0.22	0	0	0	9.93	0	0	0	0	0	0	0
14	57	3140068572		36.26	36.26	0	0	0	0	0	0	0	0	0	0	0
15	143	1348512071	0nh9t1861aa7	52.88	2.37	50	0	0	0	0	0	21,200	21,387	0	0	0
16	64	1461136608	F4hzy5H88h	60	60	0	0	0	0	0	0	0	0	0	0	0

## 2. 성능의 우수성

“데이터베이스 Hang 상태에서 유실 없는 성능 데이터 수집”



DB Hang 상태에서도 0.1 초(최대 0.01초) 단위 세션 및 SQL 정보 정상 수집



DB Hang 상태에서도 1 초 단위 세션 및 SQL 정보 정상 수집

- 한국정보통신기술협회(TTA) 품질성능 평가시험(BMT) 결과로 입증된 안정성

제품	요청사	평가결과	CPU 사용률	평가항목 및 평가 방법
셀파오라클	기획재정부	1위	0.10%	<ul style="list-style-type: none"> <li>평가 항목 : 솔루션 사용 중 DB서버의 CPU 사용률 측정</li> <li>평가 방법 : 솔루션 설치 전 / 후 사용률을 각각 측정</li> </ul>
	국방전산정보원	1위	0.59%	
	강원랜드	1위	1.5%	

\* 서버 사양에 따라 일부 차이가 있을 수 있음



### 3. 주요기능 - Multi Instance 모니터링

“여러 인스턴스에 대해 OS 자원 및 DB성능 지표를 통한 **부하 발생 현황 모니터링**”



# 3. 주요기능 - Multi Instance 모니터링(계속)

“CPU 과부하 모니터링”



1. CPU 과부하 인스턴스 확인



2. CPU 과부하 Top 프로세스 추적

Server Monitoring

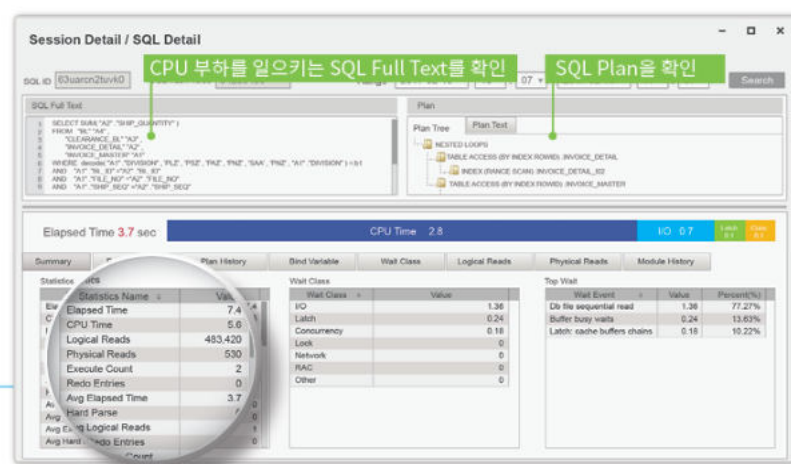
PID	User	CPU	Memory(KB)	Time	Command
1 23508	oogrid	100	115,436	2:17	oraclePCAU
2 12189	orapcaut	64.17	316,036	1:44	oraclePCAU
3 22245	oogrid	51.67	55,400	3:52	oraclePCAU
4 18190	oogrid	40.83	326,032	0:16	oraclePCAU

3. CPU 과부하 세션 확인

Session Monitoring

User name	SID	Serial	SPID	CPU(%)	Module	Program	SQL	Wait
1 KHRD	79	3456	22245	51.67	PICTBA01	JDBC Thin Cl	SELECT c.e ra: db file sequential	
2 KHRD	67	3895	23508	100	SQL*Plus	sqlplus@PCA	SELECT SUM( direct path read	
3 KHRD	104	5450	18190	40.83	PIHEB01	JDBC Thin Cl	SELECT ?+ IN: gc or request	

4. CPU 과부하 SQL 및 Stat 정보 확인



· Logical Reads, CPU Time과 같은 CPU 부하와 연관된 Stat 지표 발생량을 분석

# 3. 주요기능 - Multi Instance 모니터링(계속)

## "Lock 모니터링"

### 1. Lock 세션 발생 확인



### 2. Lock 정보 확인

Lock Tree

Instance: node1

Instance	SID	Lock Status	SPID	Serial#	Lock Type	Hold Mode	Request Mode	Wait Time	Object ID	Status	Name
node1	10	H	6299	6647	TX	None	None	49	152967	INACTIVE	SQL*Net message from
node1	67	W	2510	444149	TX	None	EX	45	152967	ACTIVE	enq:TX - row lock content
node1	90	W	22895	48745	TX	None	EX	29	152967	ACTIVE	ACTIVE - row lock content

Labels: Lock holder (row 1), Lock Waiter (row 2)

### 3. Holder Session Kill

Session Detail

SQL Text 확인

```

SQL Full Text
1 SELECT * FROM emp_emp1
2 WHERE emp_emp1.emp_id = 10
3 AND emp_emp1.emp_dept = 10
4 AND emp_emp1.emp_sal > 1000
5 AND emp_emp1.emp_hiredate < TO_DATE('2017-01-01', 'YYYY-MM-DD')
6 AND emp_emp1.emp_job = 'MANAGER'
7 AND emp_emp1.emp_manager = 10
8 AND emp_emp1.emp_organizational_level = 10
9 AND emp_emp1.emp_organizational_level = 10
10 AND emp_emp1.emp_organizational_level = 10
11 AND emp_emp1.emp_organizational_level = 10
12 AND emp_emp1.emp_organizational_level = 10
13 AND emp_emp1.emp_organizational_level = 10
14 AND emp_emp1.emp_organizational_level = 10
15 AND emp_emp1.emp_organizational_level = 10
16 AND emp_emp1.emp_organizational_level = 10
17 AND emp_emp1.emp_organizational_level = 10
18 AND emp_emp1.emp_organizational_level = 10
19 AND emp_emp1.emp_organizational_level = 10
20 AND emp_emp1.emp_organizational_level = 10
21 AND emp_emp1.emp_organizational_level = 10
22 AND emp_emp1.emp_organizational_level = 10
23 AND emp_emp1.emp_organizational_level = 10
24 AND emp_emp1.emp_organizational_level = 10
25 AND emp_emp1.emp_organizational_level = 10
26 AND emp_emp1.emp_organizational_level = 10
27 AND emp_emp1.emp_organizational_level = 10
28 AND emp_emp1.emp_organizational_level = 10
29 AND emp_emp1.emp_organizational_level = 10
30 AND emp_emp1.emp_organizational_level = 10
31 AND emp_emp1.emp_organizational_level = 10
32 AND emp_emp1.emp_organizational_level = 10
33 AND emp_emp1.emp_organizational_level = 10
34 AND emp_emp1.emp_organizational_level = 10
35 AND emp_emp1.emp_organizational_level = 10
36 AND emp_emp1.emp_organizational_level = 10
37 AND emp_emp1.emp_organizational_level = 10
38 AND emp_emp1.emp_organizational_level = 10
39 AND emp_emp1.emp_organizational_level = 10
40 AND emp_emp1.emp_organizational_level = 10
41 AND emp_emp1.emp_organizational_level = 10
42 AND emp_emp1.emp_organizational_level = 10
43 AND emp_emp1.emp_organizational_level = 10
44 AND emp_emp1.emp_organizational_level = 10
45 AND emp_emp1.emp_organizational_level = 10
46 AND emp_emp1.emp_organizational_level = 10
47 AND emp_emp1.emp_organizational_level = 10
48 AND emp_emp1.emp_organizational_level = 10
49 AND emp_emp1.emp_organizational_level = 10
50 AND emp_emp1.emp_organizational_level = 10
51 AND emp_emp1.emp_organizational_level = 10
52 AND emp_emp1.emp_organizational_level = 10
53 AND emp_emp1.emp_organizational_level = 10
54 AND emp_emp1.emp_organizational_level = 10
55 AND emp_emp1.emp_organizational_level = 10
56 AND emp_emp1.emp_organizational_level = 10
57 AND emp_emp1.emp_organizational_level = 10
58 AND emp_emp1.emp_organizational_level = 10
59 AND emp_emp1.emp_organizational_level = 10
60 AND emp_emp1.emp_organizational_level = 10
61 AND emp_emp1.emp_organizational_level = 10
62 AND emp_emp1.emp_organizational_level = 10
63 AND emp_emp1.emp_organizational_level = 10
64 AND emp_emp1.emp_organizational_level = 10
65 AND emp_emp1.emp_organizational_level = 10
66 AND emp_emp1.emp_organizational_level = 10
67 AND emp_emp1.emp_organizational_level = 10
68 AND emp_emp1.emp_organizational_level = 10
69 AND emp_emp1.emp_organizational_level = 10
70 AND emp_emp1.emp_organizational_level = 10
71 AND emp_emp1.emp_organizational_level = 10
72 AND emp_emp1.emp_organizational_level = 10
73 AND emp_emp1.emp_organizational_level = 10
74 AND emp_emp1.emp_organizational_level = 10
75 AND emp_emp1.emp_organizational_level = 10
76 AND emp_emp1.emp_organizational_level = 10
77 AND emp_emp1.emp_organizational_level = 10
78 AND emp_emp1.emp_organizational_level = 10
79 AND emp_emp1.emp_organizational_level = 10
80 AND emp_emp1.emp_organizational_level = 10
81 AND emp_emp1.emp_organizational_level = 10
82 AND emp_emp1.emp_organizational_level = 10
83 AND emp_emp1.emp_organizational_level = 10
84 AND emp_emp1.emp_organizational_level = 10
85 AND emp_emp1.emp_organizational_level = 10
86 AND emp_emp1.emp_organizational_level = 10
87 AND emp_emp1.emp_organizational_level = 10
88 AND emp_emp1.emp_organizational_level = 10
89 AND emp_emp1.emp_organizational_level = 10
90 AND emp_emp1.emp_organizational_level = 10
91 AND emp_emp1.emp_organizational_level = 10
92 AND emp_emp1.emp_organizational_level = 10
93 AND emp_emp1.emp_organizational_level = 10
94 AND emp_emp1.emp_organizational_level = 10
95 AND emp_emp1.emp_organizational_level = 10
96 AND emp_emp1.emp_organizational_level = 10
97 AND emp_emp1.emp_organizational_level = 10
98 AND emp_emp1.emp_organizational_level = 10
99 AND emp_emp1.emp_organizational_level = 10
100 AND emp_emp1.emp_organizational_level = 10
    
```

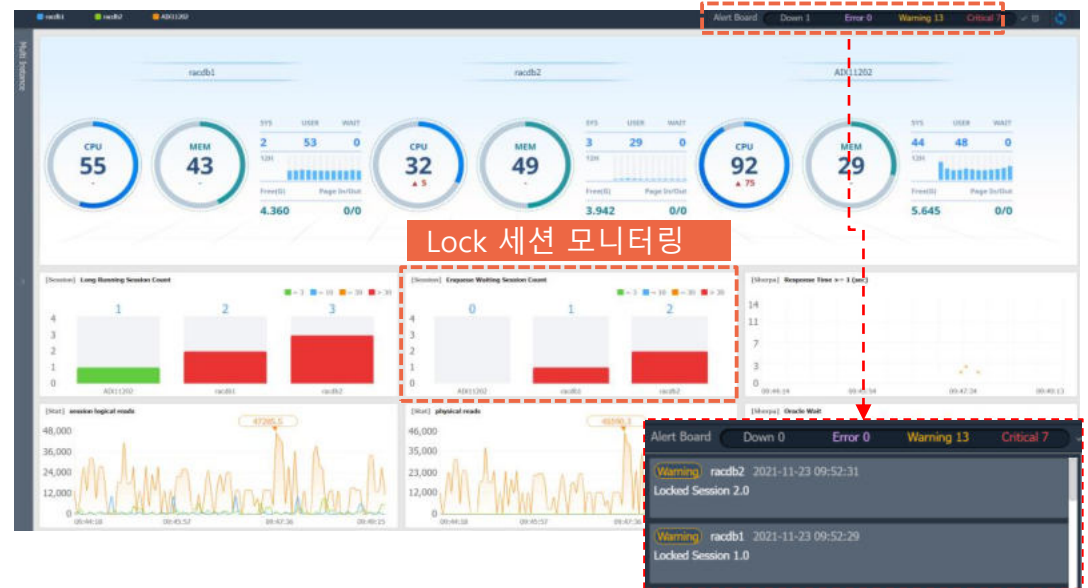
Kill Session

Session Information

SID: 10  
Serial# 6647  
Status: INACTIVE  
SPID/CPID: 6299 / 6298  
User Name: SCOTT  
Program: sqlplus@ORA\_LINK (TNS V1 - V3)  
Module: sqlplus@ORA\_LINK (TNS V1 - V3)  
Host Event: SQL\*Net message from client  
Message: ORA\_LINK

Undo Usage

Undo Size: 1 MB  
Undo Blocks: 127 Blocks  
Undo Records: 127



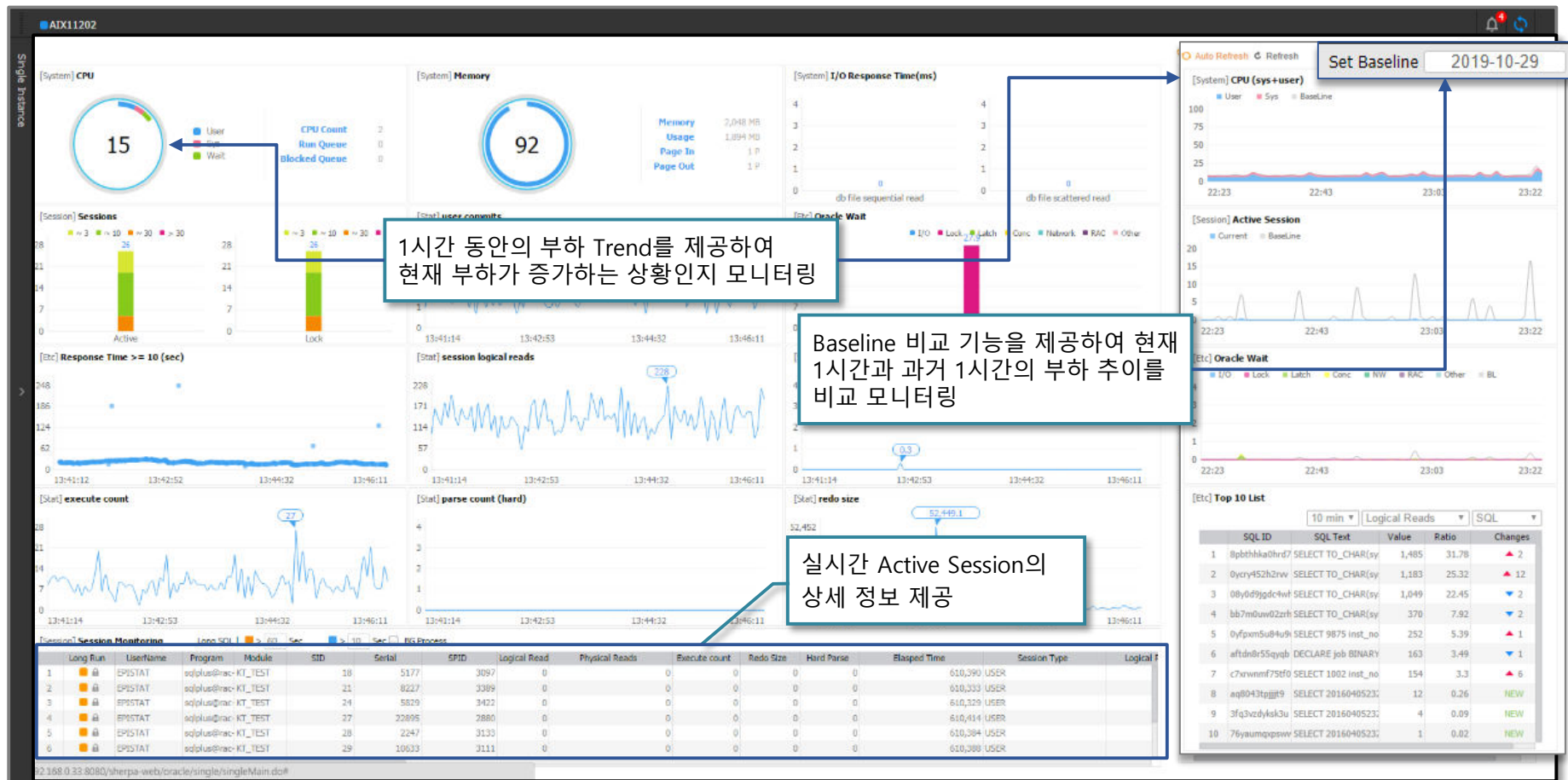
Lock 대기 세션 수 초과 Alert 발생 시 Lock 발생 시점의 분석화면으로 연계

### Alert 연계를 통한 Lock 분석



### 3. 주요기능 - Single Instance 모니터링

“하나의 인스턴스에 대해 현재 DB 상황과 수집된 과거 Baseline의 DB 상황을 비교 모니터링”



### 3. 주요기능 - Full Table Scan SQL 모니터링

“신규 업무 추가 후 운영DB에서 개발 의도와 다르게 Full Scan하는 SQL 모니터링”

- 신규업무 개발 후 운영 DB에 적용 시 개발 의도와 다르게 Full Table Scan 하면서 운영DB에 부하를 주는 경우가 발생할 수 있어 운영 DB서버에서 Full Table Scan하는 SQL만 별도로 검출할 수 있는 기능

The screenshot displays a database monitoring tool interface with the following sections:

- SQL List:** A table listing SQL queries. The first row is highlighted in green, indicating a Full Table Scan. The table has columns: Table Owner, Table Name, SQL ID, Hash Value, Plan Hash Value, SQL Order, SQL Text, Last Load Time, Num Rows, Blocks, Empty Block, Last Analyzed, Partitioned, and Temporary.
- Table Owner(Schema):** A dropdown menu showing selected owners: JENNIFER and STRESS.
- SQL Full Text:** A text area showing the full SQL query: `1. SELECT * FROM LOCK_TEST`.
- Execution Plan:** A section showing the execution plan for the selected query, including the statement and table access method: `TABLE ACCESS FULL:JENNIFER.LOCK_TEST`.

	Table Owner	Table Name	SQL ID	Hash Value	Plan Hash Value	SQL Order	SQL Text	Last Load Time	Num Rows	Blocks	Empty Block	Last Analyzed	Partitioned	Temporary
1	JENNIFER	LOCK_TEST	ayk15fg3rg2d3	3346500003	672268810	1	SELECT * FROM LOCK_TEST	2021-05-18 15:45:14	1	1	0	2021-05-16 06:01:11	NO	N
2	JENNIFER	LOCK_TEST	dcrzpm7118gu	3457196538	1018147572	2	update lock_test set name = 'lee'	2021-05-18 15:45:14	1	1	0	2021-05-16 06:01:11	NO	N
3	STRESS	EMP2	5n541szvnrp83	4148942083	3641005271	1	select count(*) from stress.emp2	2021-05-18 15:45:14	57,660	57,660	0	2017-03-09 22:10:30	NO	N
4	STRESS	FFS_TEST2	0nhj9t1861aa7	1348512071	3763623111	1	select count(*) from stress.ffc_test2	2021-05-18 15:45:14	36,241	36,241	0	2017-03-08 22:12:05	NO	N

### 3. 주요기능 - Literal SQL 모니터링

“데이터베이스의 부하를 유발하는 Top Literal SQL\* 검출”

Search Literal SQL

Real-time > Search Literal SQL

Instance: SOL11204 | SQL Count > 10 | Execute Count < 5 | Schema Not In: SYS, SYSTEM

TOP 10 SQL Count (Literal SQL 쿼리 결과 상위 10개만 표시됩니다.)

Top 1: 139 | Top 2: 100 | Top 3: - | Top 4: - | Top 5: - | Top 6: - | Top 7: -

Force Matching Signature	SQL ID	Executions	Plan Hash Value	Child Number	Optimizer Env Hash Value	Parsing Schema Name	SQL Text
11346731057457102876	4hkvb1am6h5xy	1	3357320190	0	1806367558	SHERPA	SELECT sid, serial# as serialNo, status FROM v\$session WHERE sid = '10' AND ser
11346731057457102876	6hw2at0n9pft5	1	3357320190	0	1806367558	SHERPA	SELECT sid, serial# as serialNo, status FROM v\$session WHERE sid = '102' AND s
11346731057457102876	09wsc7ypmq7x4	1	3357320190	0	1806367558	SHERPA	SELECT sid, serial# as serialNo, status FROM v\$session WHERE sid = '102' AND s
11346731057457102876	d83cvfdtjgak6	1	3357320190	0	1806367558	SHERPA	SELECT sid, serial# as serialNo, status FROM v\$session WHERE sid = '102' AND s
11346731057457102876	cv3fnx83103ra	1	3357320190	0	1806367558	SHERPA	SELECT sid, serial# as serialNo, status FROM v\$session WHERE sid = '107' AND s
11346731057457102876	2245uas1qm9gw	1	3357320190	0	1806367558	SHERPA	SELECT sid, serial# as serialNo, status FROM v\$session WHERE sid = '107' AND s

동일한 Force Matching Signature 값을 가진 SQL을 Literal SQL로 판단

```
SELECT sid,
serial# AS serialno,
status
FROM v$session
WHERE sid = '10'
AND serial# = '1125'
```

- 동일 Literal SQL 별로 그룹핑 후 Top 10개 Literal SQL 정보 제공

\* Literal SQL은 실행계획을 공유하지 못해 과도한 hard parsing을 유발하는 SQL로, Literal SQL이 많을 경우 DB서버의 CPU를 과도하게 사용할 수 있기 때문에 성능 개선(튜닝)이 필요함

### 3. 주요기능 - 사용자 정의형 대시보드

“사용자가 직접 모니터링 항목을 SQL로 정의하고 재구성 할 수 있는 **SQL 대시보드 제공**”

사용자가 SQL Dashboard 설정

Query Name	Instance	Chart Title	Refresh Interval	Type	SQL Text	Scale	Unit	Threshold	Description
1	19N13283	session	10	Line	select count(*) as char	0		0	
2	19N13283	s	3	Bar	select count(*) as char	0		0	
3	19N13283	w	3	Grid	select count(*) as char	0		0	

SQL 작성 및 등록

```

[Step 1] SQL
1 SELECT line_no as 라인번호, round(defect_cnt/prod_cnt)+100 as 불량율
2 FROM (select '1' as line_no, count(prod) as prod_cnt, count(defect) as defect_cnt
3 from t_prod_log where linecode = '0010A'
4 union all
5 select '2' as line_no, count(prod) as prod_cnt, count(defect) as defect_cnt
6 from t_prod_log where linecode = '0010A'
7 union all
8 select '3' as line_no, count(prod) as prod_cnt, count(defect) as defect_cnt
9 from t_prod_log where linecode = '0010C');
10
    
```

출력할 차트 형태 지정

DB성능 모니터링 | 업무 모니터링

회계결산 진행현황

```

SELECT line_no as 라인번호, round(defect_cnt/prod
FROM (select '1' as line_no, count(prod) as prod_cnt
from t_prod_log where linecode = '0010A'
union all
select '2' as line_no, count(prod) as prod_cnt
from t_prod_log where linecode = '0010A'
) as all;
    
```

정수기관별 국고금 수납처리현황

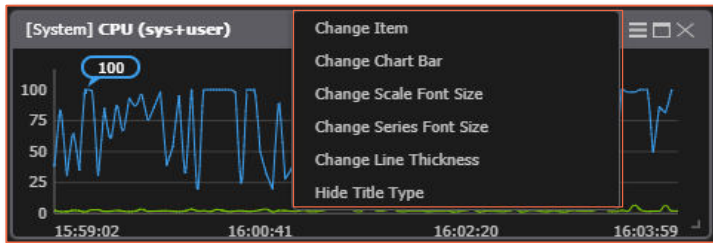
```

SELECT to_char(logtime,'HH'), count
FROM total_prod_item
GROUP BY to_char(logtime,'HH');
    
```

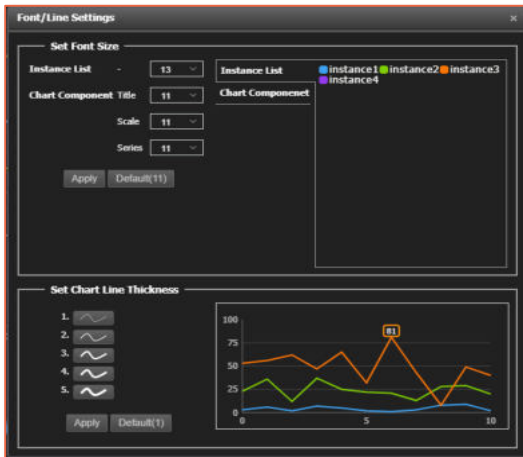
- Interval(주기)을 자유롭게 설정 가능하고, 차트 타입은 bar, grid, line 형태로 제공
- 최초 connection 맺은 후 재사용하여 모니터링

### 3. 주요기능 - 사용자 정의형 대시보드(계속)

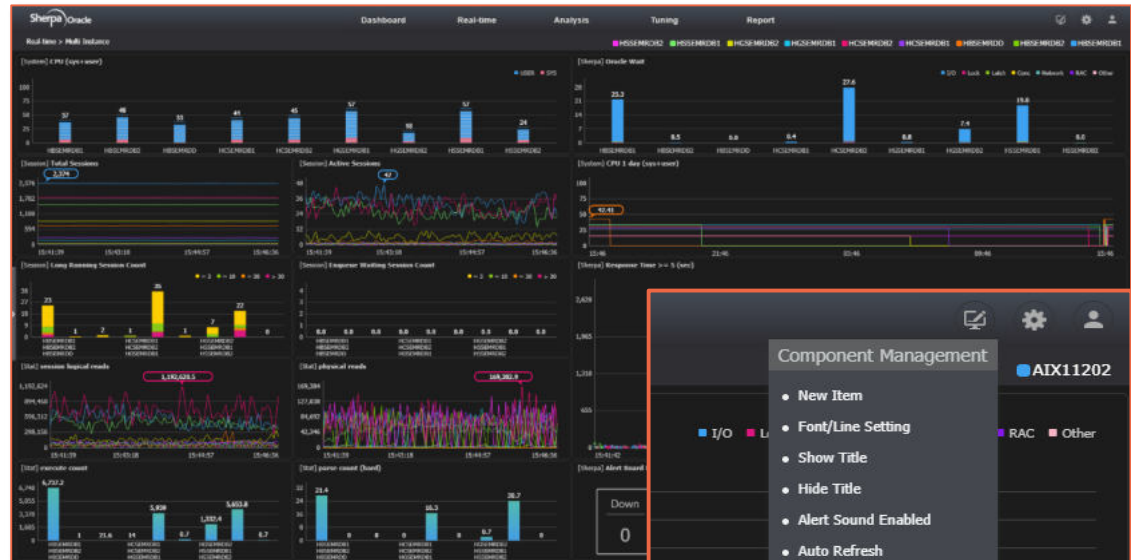
“사용자 별로 대시보드 화면을 자유롭게 구성”



- 개별 차트에 대한 Layout 설정



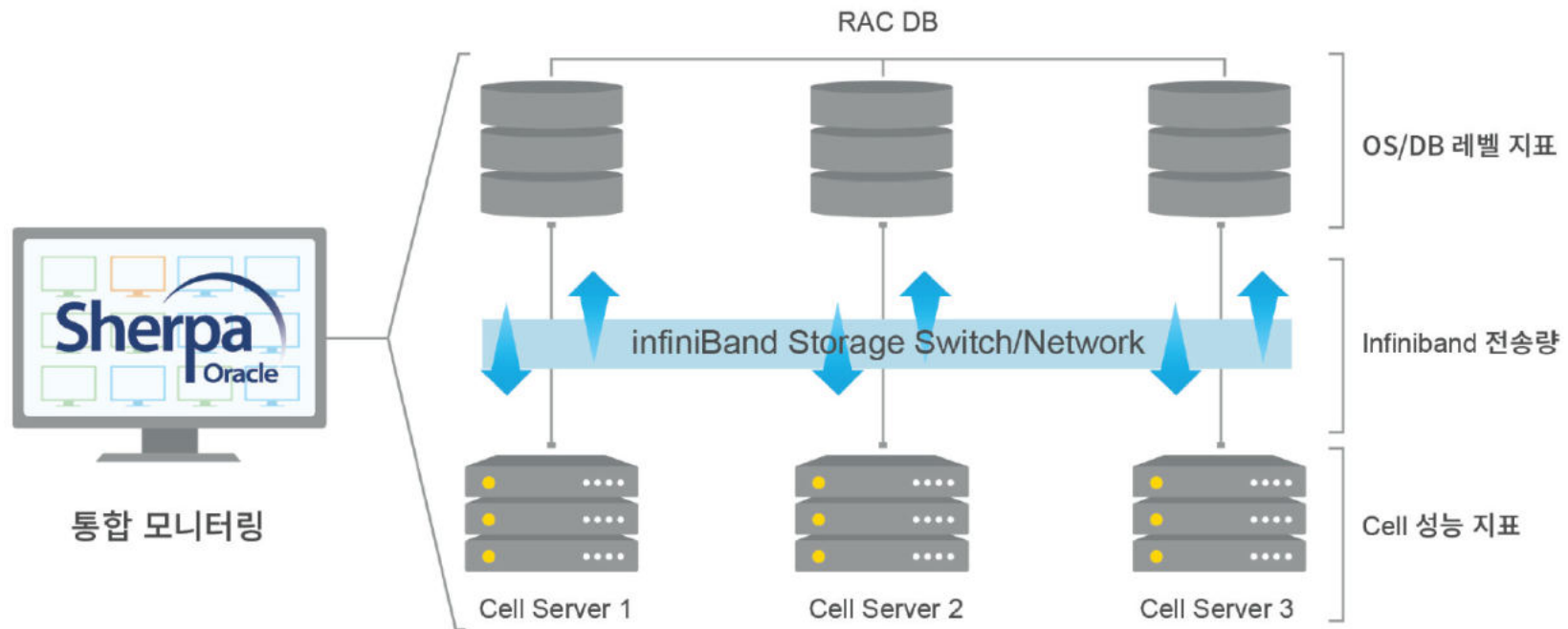
- 전체 차트에 대한 Layout 일괄설정



- 사용자 별 Layout 저장 / 불러오기
- 대시보드 화면을 자유롭게 구성



#### “Exadata/OSC(Oracle Super Cluster)의 주요 성능지표 모니터링 ”



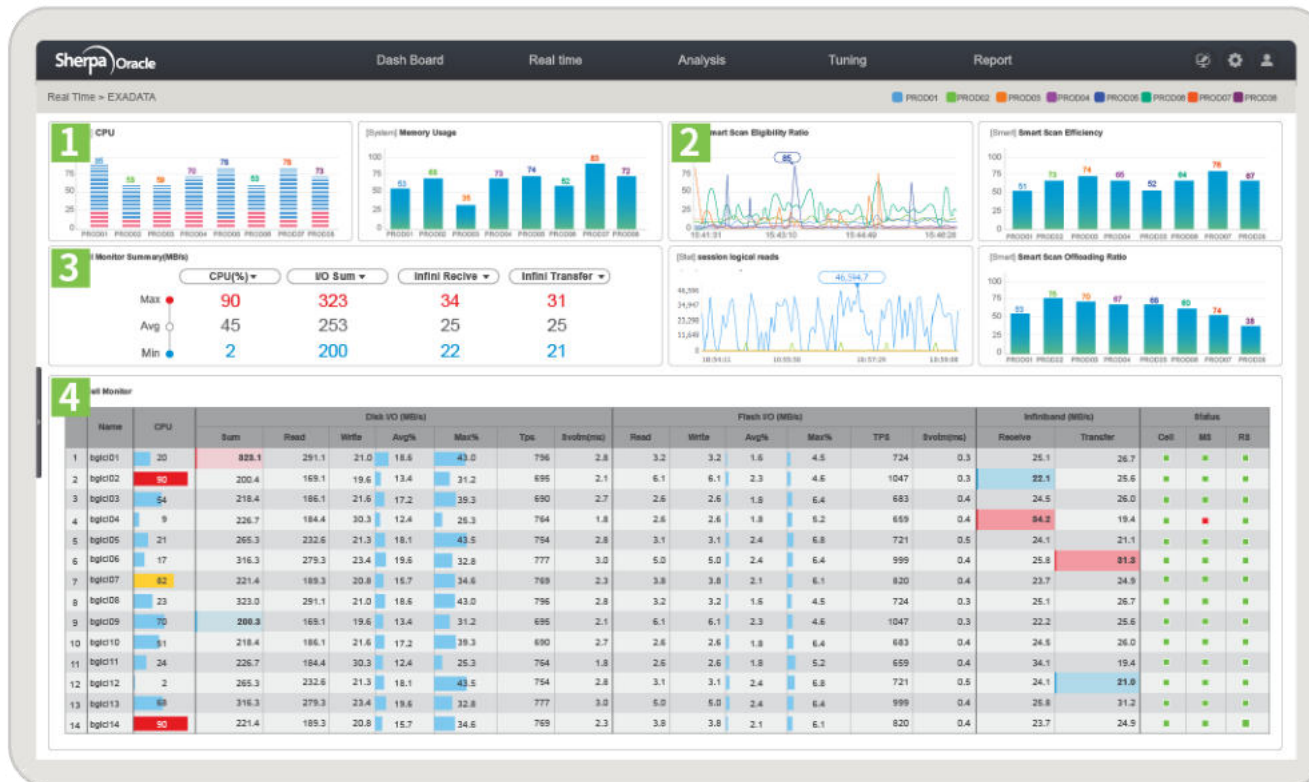
<Exadata / OSC 전용 모니터링 개념도>

\* 주요 모니터링 항목 : Cell 서버에 대한 성능, Cell 서버와 DB서버 간 네트워크 전송량(Infiniband)등

### 3. 주요기능 - 엑사 대시보드(계속)

## “Exadata/OSC(Oracle Super Cluster)의 주요 성능지표 모니터링 ”

- DB서버의 OS지표와 EXADATA 관련 지표를 함께 모니터링 (1,2 번)
- Cell 서버 Summary 를 통해 Cell 서버 전체 현황 모니터링 (3번)
- Cell 서버 모니터링 지표로 CPU Load Balancing 여부와 I/O 현황 모니터링 (4번)



- 1 DB서버 OS지표
- 2 DB서버 스마트 스캔 지표
- 3 Cell 서버 Summary
- 4 Cell 서버 모니터링 지표

- [항목] · CPU
- Disk I/O
  - Flash I/O
  - Infiniband I/O
  - Service Check

### 3. 주요기능 - 엑사 대시보드(계속)

- **Exadata / OSC** (Oracle Super Cluster) 지원
- 엑사 모니터링 시 대부분의 DBA들이 사용하는 **sar.sh** 스크립트 기반의 **대시보드 화면 구성**
- **Cell 서버들 간의 CPU, I/O 지표를 통한 로드밸런싱 여부 확인**
- **Cell 서비스 데몬에 대한 다운여부 감지**  
(Cell 서버 관리, 모니터링 서비스 : **cellsrv, ms, rs**)
- DB 서버의 EXADATA 관련 지표와 Cell 서버의 지표를 함께 모니터링

### 3. 주요기능 - Top N Analysis

#### "Peak 구간에 대한 튜닝 대상 SQL 검출"

(1) Top N 분석  
- 30 Min 선택  
(30분 단위 분석)

(2) Top N 분석 기간  
- 09:00 ~ 16:00  
(7시간)

(3) 14:00 ~ 14:30분까지 30분 단위의 Wait Class + CPU 응답시간 추이 차트

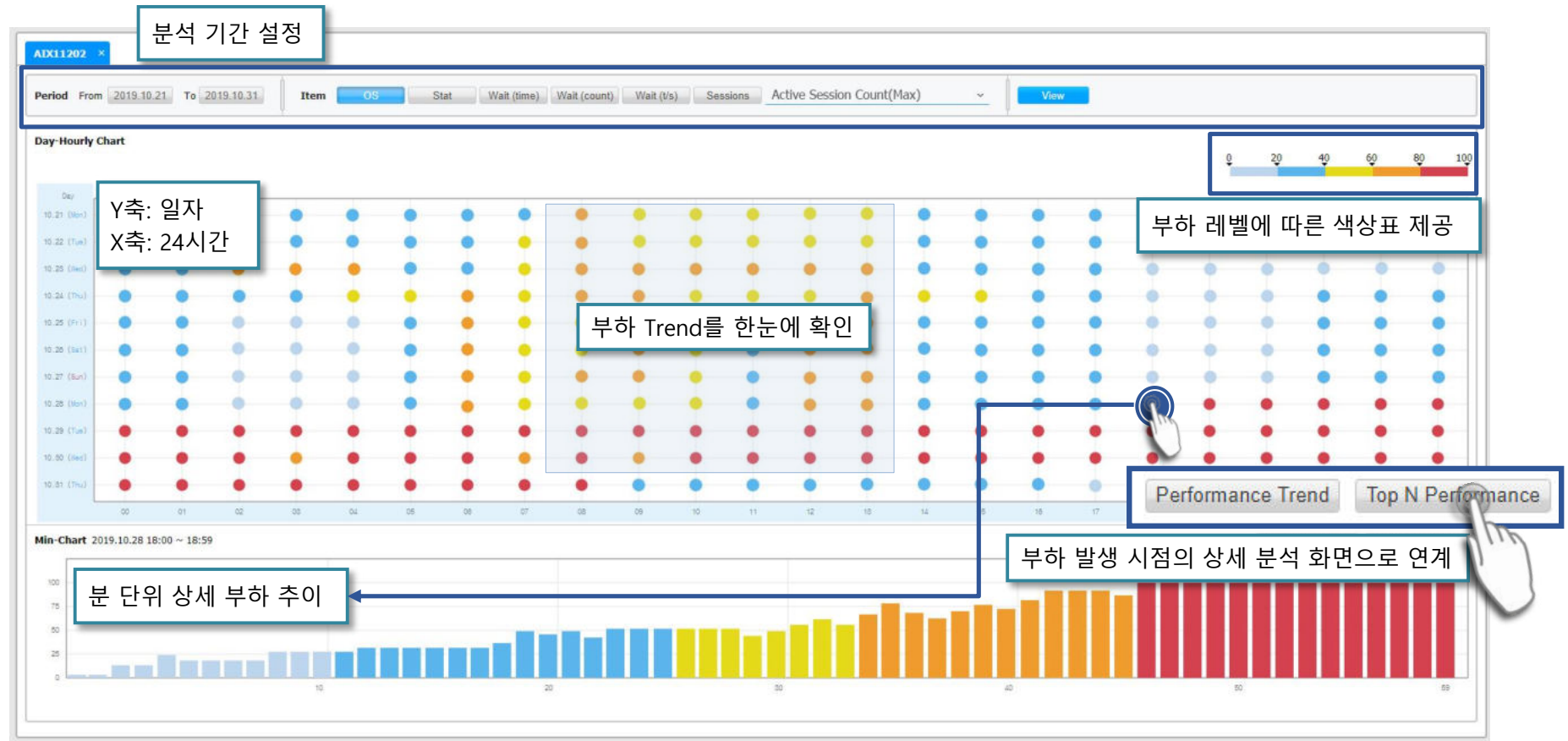
(4) Peak구간 : 14:00 ~ 14:30분, 30분 동안의 Logical Reads 기준 Top SQL 출력

SQL ID	Hash Value	SQL Text	Value	Ratio(%)
1	0nhj9t1861aa7	select count(*) from stress.ffs...	25,366,540	71.62%
2	5q6vkb9muyuvh	select /*+ index(a stress.ffs_te...	5,859,785	16.54%
3	5n541szvnrp83	select count(*) from stress.em...	4,126,597	11.65%
4	7vb4x5awc0hd3	select 'AIX11202' as Instance,	53,520	0.15%
5	dcrrzpm7118gu	update lock_test set name = '...	9,786	0.03%
6	61q4216jbvru1	SELECT * from ( SELECT /*+ ...	1,021	0%
7	ajpcnht79pcns	SELECT /*+ leading(undo_raw...	568	0%
8	faydhmtku29k	SELECT TO_CHAR(sysdate, 'YY...	448	0%
9	aw70nb17md4e5	SELECT TO_CHAR(sysdate, 'YY...	422	0%
10				
11				
12				
13	9u92fmsk3k3q5	SELECT TO_CHAR(sysdate, 'YY...	18	0%

- DB서버 부하 추이에 따라 30분/1시간/1일/7일 등 다양한 분석단위의 Top N 분석을 지원하며 0.1초로 수집한 Session & SQL 정보를 기반으로 주요지표에 대한 Top N 분석 기능을 제공합니다.

### 3. 주요기능 - Heatmap Trend

“장기간의 데이터를 부하 레벨별 색상으로 분류하여 부하 Trend 파악”



- 일자별 시간별 대상 DB의 부하추이를 한눈에 직관적으로 파악 가능
- 부하 시점의 상세 분석화면(Performance Trend, Top N Performance) 연계 기능

# 3. 주요기능 - Longterm Compare Analysis

## “일/주/월 단위 성능 지표 비교 분석 레포트”

(3) 10.23일과 10.28일 CPU와 Active Session 부하 추이를 출력  
 - CPU 증가 시 Active Session도 함께 증가했는지  
 - 23일 대비 28일 16시경부터 20시까지 Active Session이 증가하고 CPU도 함께 증가했다는 사실을 확인할 수 있음

(1) 원하는 비교 구간을 일/시간 단위로 선택  
 - 10.23일과 10.28일 비교

(2) CPU(sys+user) 와 Active Session 지표 선택  
 - CPU 와 Active Session 간 상관관계 분석

설정한 차트 타입 및 타이틀에 대한 범례 표시

- 두개 성능 지표간 상관관계 분석
- 일 단위 or 월 단위 비교 분석을 통해 레포트 활용 가능

### 3. 주요기능 - Plan 성능 비교 분석

#### “Plan 변경으로 인한 성능 비교 및 Plan 비교 분석”

- 일자별 Plan 변경 이력 제공하며, 선택한 성능 판단 기준으로 변경된 Plan에 대한 Better Plan과 Worse Plan 추이 모니터링 기능 제공

일자 별 변경 된 Plan 건수

- 전체 Changed Plan : 5건
- Better Changed Plan : 1건
- Worse Changed Plan : 4건

[Plan 성능 판단 기준]

- Elapsed Time
- CPU Time
- Buffer Gets
- Disk Reads
- Costs

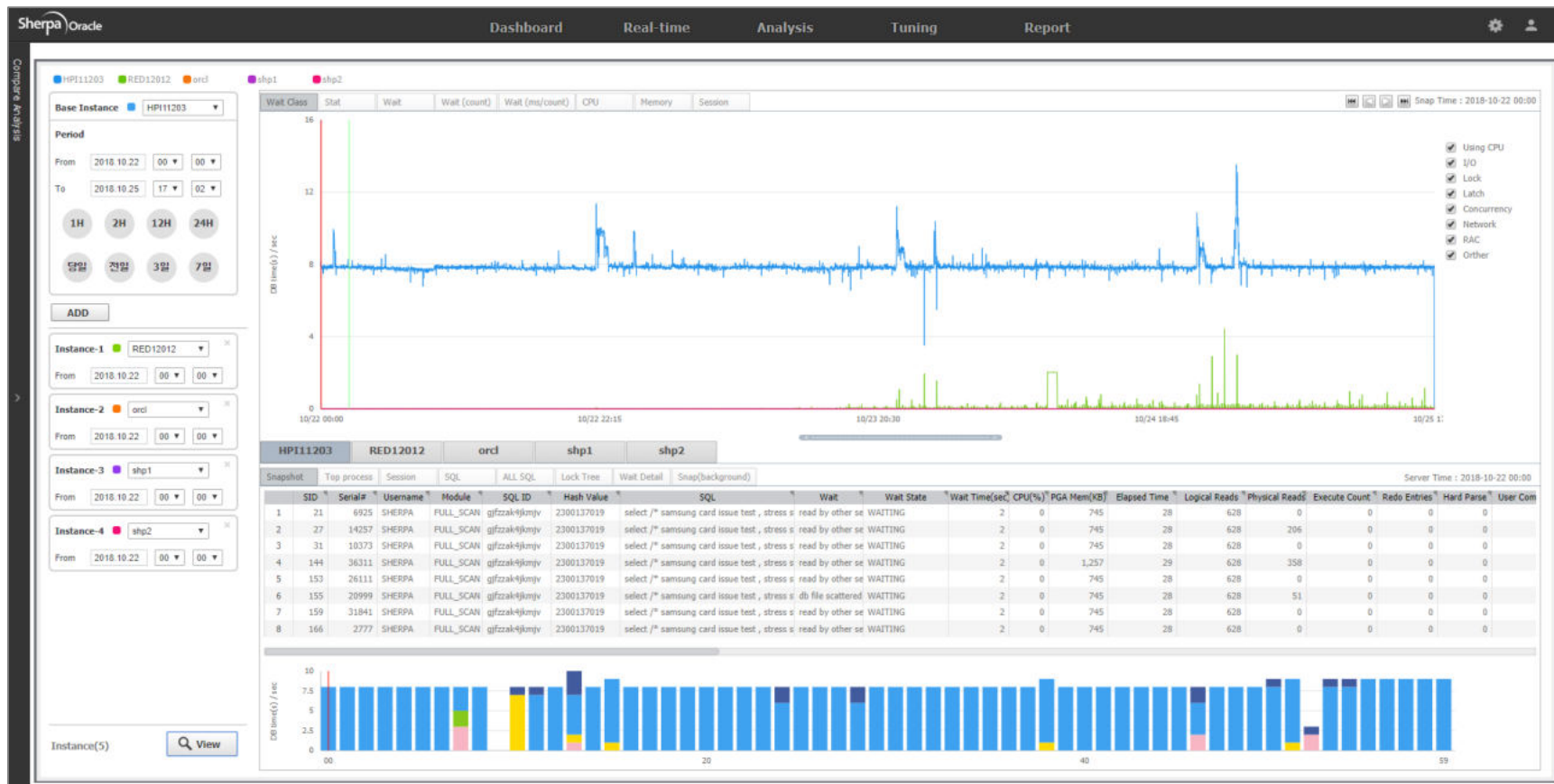
변경된 Plan 더블클릭 시,  
“Plan Compare” 화면으로 연계  
변경된 Plan에 대한 성능 분석 기능 제공

Plan 별 성능 비교 및 변경된 Plan Text 비교

### 3. 주요기능 - RAC 성능 비교 분석

## “RAC 성능 비교 분석”

- 2개 이상의 Instance를 비교 분석 할 수 있으며, 다른 인스턴스 혹은 같은 Instance 의 다른 날짜 구간을 선택 하여 비교





### 3. 주요기능 - Performance Impact Analysis

#### “테이블/인덱스에 대한 Access Type 분석”

- 특정 테이블 별 SQL Plan을 분석하여 해당 테이블에 Access한 SQL 수행내역과 SQL 사용횟수, Access Type(Full Sacn, Index Scan 등) 분석 가능
- 신규업무를 개발하여 운영 DB에 적용할 경우, 개발한 SQL들이 적절한 Access Type으로 실행되고 있는지 분석하여 운영 DB에 발생할 수 있는 잠재적 위험을 사전에 확인하고 조치할 수 있음

The screenshot displays the Performance Impact Analysis interface for the table AP\_INVOICES. It includes a table info section, a result analysis section with a donut chart, and an SQL list table. Annotations highlight specific parts of the interface.

**Table Info**

ID	Owner	Table Name	Tablespace Name	Num Rows	Blocks	Empty Block	Last Analyzed	Partitioned	Temporary	Lot Type
1	SH	AP_INVOICES	EXAMPLE	500,000	4,276	0	12/12/2017	NO	N	

**Result Analysis**

Access Type: Full Scan (37.5%), By Index (62.5%), ETC (0%)

SQL Count: 8  
Used Count: 8

**SQL List**

SQL ID	Used Count	Full Scan	By Index	Hash Value	Plan Hash Value	Child Number	Time Stamp
1 ygd7zbvwpnv6	1	1	0	4157264742	2290814231	0	12/12/2017
2 1ygd7zbvwpnv6	1	0	1	4157264742	2997799156	0	12/12/2017
3 5dxnj1x8x2s6b	1	1	0	1372676299	3986716158	0	12/12/2017
4 6g73k23aj4wz0	1	1	0	2790126239	2290814231	0	12/12/2017
5 6g73k23aj4wz0	1	0	1	2790126239	2997799156	0	12/12/2017
6 78b82t6m4vwz	1	1	0	2790126239	2290814231	0	12/12/2017
7 78b82t6m4vwz	1	0	1	2790126239	2997799156	0	12/12/2017
8 fm7yz8prtq5va	1	1	0	1872205674	2299141589	0	12/12/2017

**Annotations:**

- AP\_INVOICES 테이블에 대한 Access Type 분석
- AP\_INVOICES 테이블에 Access한 SQL 리스트
- 전체 8개 SQL 중에 Full Scan 4건, Index Scan 3건
- SQL Full Text
- 실행계획

### 3. 주요기능 - SQL Analysis

“개별 SQL에 대해 응답시간, 자원사용, Plan 이력 등의 추이 변화와 분석이 가능”

The screenshot displays the SQL Analysis tool interface. At the top, the 'SQL Detail' section shows the SQL text: `SELECT /*+ index(a stress_ffs_test2_idx0) */ Max(ename) FROM stress_ffs_test2 a WHERE ename = 1000`. The 'Plan' section shows the execution plan: `SELECT STATEMENT [ ALL_ROWS COST:50# CARDINALITY: ]`, `SORT AGGREGATE`, and `TABLE ACCESS BY INDEX ROWID:STRESS_FFS_TEST2` with `INDEX RANGE SCAN:STRESS_FFS_TEST2_IDX0(Search Col:1)`. Below this, the 'Summary' tab is active, showing various statistics and wait events. Several callout boxes highlight key features: '개별 SQL에 대하여 Range 설정을 통해 일별, 분단위별 성능 추이 제공' (Daily and weekly performance trends for individual SQL via range settings), 'SQL 수행 시점의 Plan 이력 제공' (Plan history at the time of SQL execution), 'Execute History' (Bar chart showing execution history), 'Wait Class' (Bar chart showing wait class history), 'Logical Reads' (Bar chart showing logical reads history), 'Physical Reads' (Bar chart showing physical reads history), and 'Module History' (Table showing module history).

### 3. 주요기능 - 사용자 정의형 Alert

“사용자가 원하는 Alert을 SQL로 등록하여 사용할 수 있어  
운영 시 필요한 Alert 항목이 생길 경우 언제든지 추가하여 활용 가능”



- 사용자가 SQL을 직접 작성하여 Alert 항목 등록
- SMS(메신저 등) 발송 Alert 메시지 편집 기능

### 3. 주요기능 - Excel Export 기능

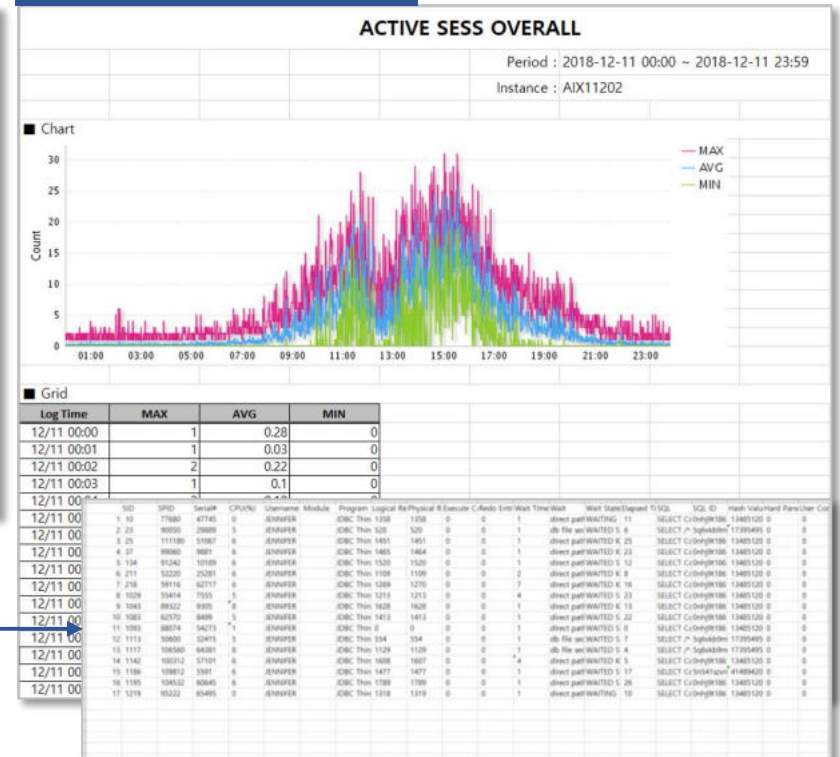
“대부분의 분석 모니터링 화면에서 Excel Export 기능을 통하여 원하는 항목에 대한 보고서 기능을 제공”

Performance Trend 분석 화면



“우클릭 > Excel export” 기능을 통하여, 분석데이터에 대한 Excel 형태의 리포트 제공

Export 된 분석 데이터



### 3. 주요기능 - Sherpa KMS 기능

“실시간 Session Detail 화면 및 Performance Trend 분석화면에서 특정 대기 이벤트 지표에 대해 학습 가이드 기능 제공”

#### Performance Trend 화면

#### Session Detail 화면

#### 결과 KMS

#### Oracle 주요 지표

**db file sequential read**

**Basic Info**

db file scattered read와 달리 상용 블록 I/O와 함께 발생하는 이벤트로 가장 흔한 이슈가 되는 것은 Index Scan 시 발생 수지이다. Index Scan 뿐 아니라 ROWID에 의한 Table Scan, constraint file/header 읽기, Undo Segment 읽기 작업에 의해 발생.

- ▶ cursor: pin S
- ▶ cursor: pin S wait on X
- ▶ cursor: pin X
- ▶ db file parallel write
- ▶ db file scattered read
- ▶ db file sequential read
- ▶ OPS lock handle
- ▶ direct path read
- ▶ direct path read temp
- ▶ direct path write
- ▶ direct path write temp
- ▶ eng: HW - contention
- ▶ eng: SQ - contention
- ▶ eng: ST - contention
- ▶ eng: TM - contention
- ▶ eng: TX - allocate ITL entry
- ▶ eng: TX - index contention
- ▶ eng: TX - row lock contention
- ▶ eng: UL - contention

Oracle 주요 성능지표에 대한 학습 가이드 문서(KMS)를 연계 제공

### 3. 주요기능 - Sherpa KMS 기능(계속)

“SQL Tuning Tool에서 plan 분석 시 Access Path 에 대한 학습 가이드 기능 제공”

The screenshot displays the SQL Tuning Tool interface on the left and the Sherpa KMS web interface on the right. The SQL Tuning Tool shows a query plan with an 'EXPLAIN PLAN' table. The Sherpa KMS interface provides detailed information about an 'INDEX UNIQUE SCAN' operation.

Object Typ	Object Ow	Object Na	Num Row	distinct k	Blocks	Last An
TABLE	SYS	OBJ\$	73,137	0	1,230	2021.05.1
INDEX (UNI)	SYS	I_OBJ1	73,137	73,137	260	2021.05.1
INDEX (UNI)	SYS	I_USER2	134	134	1	2021.05.1
CLUSTER	SYS	USER\$	134	0	7	2021.05.1
INDEX (CLU)	SYS	I_USER#	68	68	1	2017.01.2
CLUSTER	SYS	IND\$	2,883	0	1,636	2021.05.1
INDEX (UNI)	SYS	I_IND1	2,883	2,883	7	2021.05.1

**Index Unique Scan에 대한 가이드**

**Operation**

- INDEX UNIQUE SCAN
- Index Unique Scan 은 인덱스 검색에 있어 가장 이상적인 형태입니다.
- Index 가 Unique Index 로 생성이 되어야, Unique Scan이 가능합니다.
- 이 자체만으로는 최적화된 상태이나, Nested Loops Join의 Inner 테이블인 경우, Driving Table의 결과 Rows 만큼 반복 수행될 수 있으니, 이 부분에 대한 검증은 해야합니다.

**Index**

**I\_IND1**

**Basic Info**

- Index Owner : SYS
- Index Name : I\_IND1
- Index Columns : OBJ#
- Num Rows : 2,883
- Distinct Key : 2,883
- Leaf Blocks : 7
- Last Analyzed : 2021.05.19 23:06:34

**Plan Analysis**

Searching Column : 1

Access Predicate

"I"."OBJ#"=:B1

- 인덱스 컬럼 조건 중 검색 범위를 좁여주는 효과가 있는 조건절입니다.
- Unique Scan은 인덱스 컬럼 모두가 조건절에서 "=" 형태로 사용되는 최적의 상태입니다.

Filter Predicate

- 인덱스 컬럼 조건 중 Filter 기능을 통해 Table의 Random Access를 줄여주는 조건절입니다.

# 4. WAS-DB 통합 모니터링(End-to-End) 제공

## WAS, DB 통합모니터링 및 분석 (End-to-End)

- JENNIFER WAS 모니터링 솔루션과 Sherpa DBMS 모니터링 솔루션의 통합
- Sherpa의 성능데이터로 SQL 대기구간별 소요시간과 SQL Plan 확인을 통해 SQL 응답시간에 대한 원인 분석과 즉시 조치가 가능
- 특허 취득한 기술이 적용된 WAS와 DB의 통합 모니터링(특허등록번호: 제 10-1909625호, 등록일: 2018.10.12)
- 국방전산정보원, 한국환경공단, 산업인력공단, 경찰청 등 200여 고객사에서 WAS, DB 통합모니터링(End-to-End) 운영 중



특허 취득



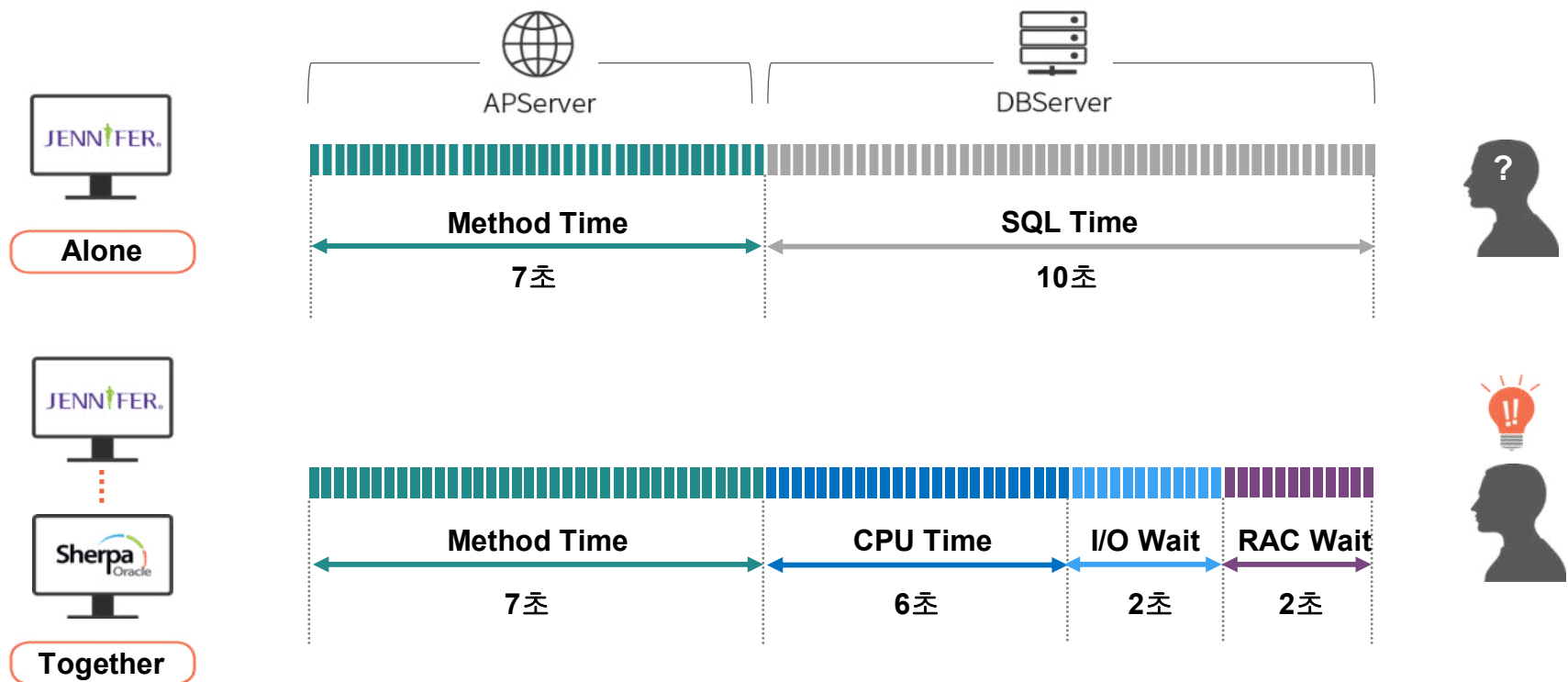
※ APM 솔루션을 JENNIFER 사용 시



타 APM 솔루션(Zenius)과 연계가 가능하여 WAS-DB 통합모니터링(End-to-End) 및 분석 제공

# 4. WAS-DB 통합 모니터링(End-to-End) 제공 - 셀파 & 제니퍼 연동

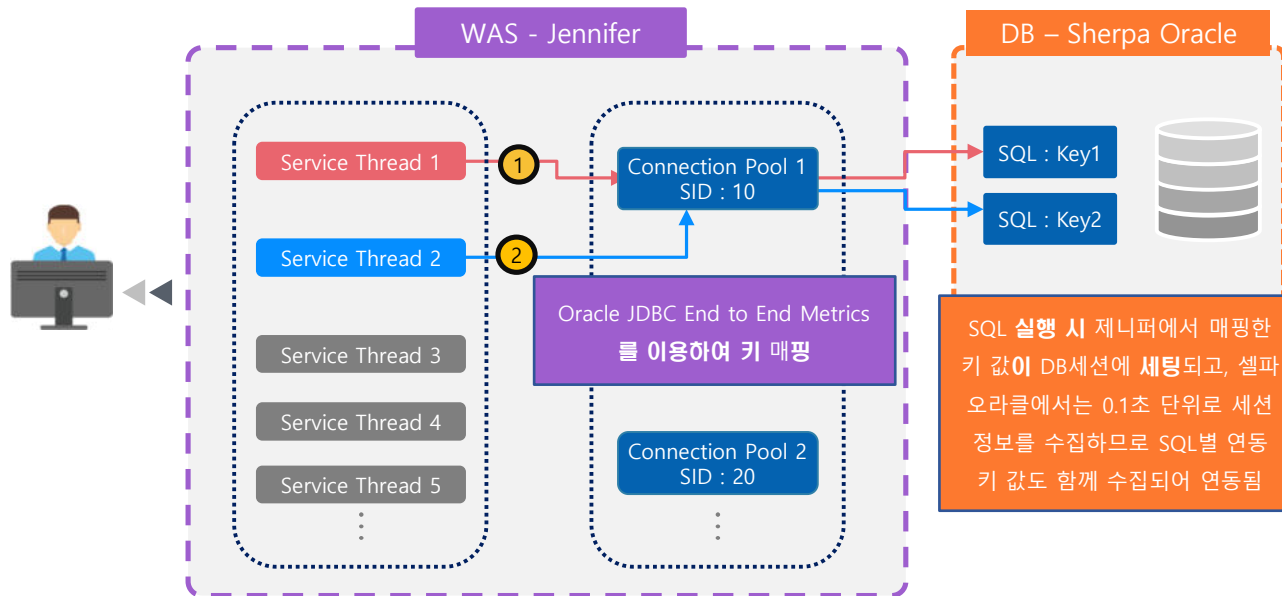
“Jennifer에서 셀파오라클을 통한 사용자 응답 시간 기반의 SQL 타임라인 분석”





# 4. WAS-DB 통합 모니터링(End-to-End) 제공 - 셸파 & 제니퍼 연동 아키텍처

셸파오라클과 제니퍼 연동 방식은 국내 어떤 솔루션보다 **가장 정밀(0.1초 이상)**하게 연동되는 방식이며 연동 방식에 대한 특허도 보유하고 있습니다.



SQL 실행 시 제니퍼에서 매핑한 키 값이 DB세션에 세팅되고, 셸파오라클에서는 0.1초 단위로 세션 정보를 수집하므로 SQL별 연동 키 값도 함께 수집되어 연동됨



<정상적으로 전송 시 v\$session 에 세팅되는 값>

SID	Service Thread	SQL문	client_identifier
10	Service1	AAA	62562bae0d7a34426e90389400000057f000001testOracle
10	Service2	BBB	62562bae62fd8371962259a3000000047f000001testOracle

- 제니퍼와 셸파오라클 간 연계 시, 실행되는 모든 SQL들에 대해 제니퍼에서 Oracle JDBC End to End Metrics 를 이용하여 세팅하며 부하는 거의 없음
- 셸파오라클의 경우 0.1초 Session & SQL정보를 수집하므로, 0.1초 이상 수행된 SQL들은 제니퍼와 연동되어 해당 SQL의 프로그램 소스와 파라미터 정보(제니퍼) 뿐만 아니라 SQL의 성능 정보(셸파오라클)까지 한번에 확인 가능

## 4. WAS-DB 통합 모니터링(End-to-End) 제공 - 셸파 & 제니퍼 연동 아키텍처

### ❖ 제니퍼 - 셸파 연동 방식 제약조건

#### 1. 연동을 위한 오라클 대상DB 제약조건

- WAS의 서비스와 DB 의 세션 간 연동 키값인 uuid 를 v\$session 의 client\_identifier 컬럼에 저장하기 때문에 v\$session의 client\_identifier 컬럼을 다른 용도로 사용하고 있을 경우 연동 불가

#### 2. 연동을 위한 셸파, 제니퍼의 버전 호환성

- 한국은행 셸파오라클 버전 : 5.0.0.22.2 이므로 아래 표에서 "5.0 이상" 에 해당됨

Chrome version	Sherpa Oracle version	Jennifer version	Sherpa & Jennifer 정방향	Sherpa 역방향
94 이상	5.0 이상	5.6.0.8 이상	Jennifer 가 https 로 연결되어야 함	기존 팝업 -> 전체 화면 조회(탭 생성됨)
94 이상	5.0 이상	5.6.0.8 미만	연결 안됨	연결 안됨
94 이상	4.2 버전대	5.6.0.8 이상	Jennifer 가 https 로 연결되어야 함	정상 조회
94 이상	4.2 버전대	5.6.0.8 미만	연결 안됨	정상 조회
93 이하	5.0 이상	5.6.0.8 이상	http 연결 가능	기존 팝업 -> 전체 화면 조회(탭 생성됨)
93 이하	5.0 이상	5.6.0.8 미만	http 연결 가능	연결 안됨
93 이하	4.2 버전대	5.6.0.8 이상	http 연결 가능	정상 조회
93 이하	4.2 버전대	5.6.0.8 미만	http 연결 가능	정상 조회

# 4. WAS-DB 통합 모니터링(End-to-End) 제공 - 셸파 & 제니퍼 연동

## Jennifer & Sherpa 대시보드



# 4. WAS-DB 통합 모니터링(End-to-End) 제공 - 셀파 & 제니퍼 연동

## 제니퍼 X-View → 셀파 연계

### 제니퍼 X-View 의 Timeline 분석

제니퍼 X-View 의 Timeline 분석

Instance Name	SID	Username	Machine	First Time	종료 시간	종료 시점	종료 시점	SQL 시간	Elapsed	Fetches	CPU 시간	ERROR	비고
ORA_3	0			0	12.20.07	12.20.07		8,813	6,308	0	0	3	NetIOCacheIndex.jsp
ORA_3	270304			0	12.20.07	12.20.07		5,003	4,797	0	0	3	NetIOCacheIndex.jsp
ORA_3	0			0	12.20.08	12.20.08		4,200	4,173	0	0	3	NetIOCacheIndex.jsp
ORA_3	4071313			0	12.20.12	12.20.12		6,330	6,166	0	0	3	NetIOCacheIndex.jsp
ORA_3	0			0	12.20.34	12.20.34		4,314	4,189	0	0	3	NetIOCacheIndex.jsp
ORA_3	4611117			0	12.20.37	12.20.37		5,102	5,067	0	0	3	NetIOCacheIndex.jsp

SQL Time 4초

DBServer

SQL 성능 상세 보기

```

select /*+ index(a stress_ffs_test2_idx01) */ max(ename) from
stress_ffs_test2 a where empno = #
    
```

Inline Parameter[1000]

• 제니퍼 서비스 프로파일을 통하여 개별 트랜잭션의 응답시간 분포 및 병목 유형의 패턴을 파악할 수 있으나 SQL Time의 세부적인 분석은 불가



### 셀파오라클의 SQL Detail 화면으로 연계

셀파오라클의 SQL Detail 화면으로 연계

SQL Time 4초

CPU Time 1.0 I/O 3.0

• SQL Plan 분석 결과 비효율적인 Index Scan 에 의한 과도한 I/O 대기시간 3초가 발생하였음

# 4. WAS-DB 통합 모니터링(End-to-End) 제공 - 셸파 & 제니퍼 연동

## 셸파 → 제니퍼 X-View 연계

**셸파오라클**

부하 SQL 정보에서 우클릭하여 Jennifer 연계

**제니퍼 X-View의 Timeline 분석화면**

Instance	Business	GUID	응답이벤트 ID	Network 시간	종료 시간	수립 시간	퇴임시간	SQL 시간
ora_Sherpa	계정관리		475621464	0	14:25:26.184	14:25:26.185		1,921

```

[ SQL 보기 ] select /*+ index(a stress_ffs_test2_idx01) */ max(name) from stress_ffs_test2 a where empno = #
    
```

- 셸파오라클 Performance Trend 에서 부하 SQL 확인
- 제니퍼 연계 기능을 통해 해당 SQL을 호출한 어플리케이션 정보 추적

- 제니퍼 X-View로 연계
  - SQL 을 호출한 program 소스 정보 확인
  - SQL 에서 실제 사용된 파라미터 정보 확인



### III. 비교자료

# 1. 셸파오라클 vs 타사제품

항목	상세기능	Sherpa Oracle	타사 모니터링 제품
일반	▪ DMA(Direct Memory Access) 방식으로 정보 수집	○	○
	▪ 정밀한 성능 정보 수집	<b>디폴트 SQL 0.1초, 세션 0.1초</b>	디폴트 SQL 0.2초, 세션 1초
	▪ 수집된 성능 정보 유실 방지 기능	○	○
	▪ 수집된 모든 성능데이터 수집서버 즉시 저장 여부	○	최소 10분 지나야 함
	▪ 최근 데이터 실시간 분석 기능	<b>즉시 분석</b>	최소 10분 지나야 함
	▪ 이기종 DB 통합 모니터링 기능	○	○
실시간 모니터링	▪ Multi Instance 모니터링	○	○
	▪ 사용자가 별도의 업무용 모니터링 차트 화면 (SQL 대시보드) 구성	○	X
	▪ ASM Disk group 별 상태를 모니터링	○	X
	▪ ASM 상태 Alert	○	X
	▪ Backup 수행 상태 모니터링 및 Alert	○	X
	▪ LISTENER 로그 Alert	○	○
	▪ CRS 로그 Alert	○	X
분석	▪ Undo, Temp 사용 세션과 사용량 수집 및 분석	○	X
	▪ 시간대 별 File System 사용량 변화 비교 분석	○	X
	▪ SQL 수행 시점의 Plan 정보 수집	○	○
	▪ 특정 기간 동안 실행계획 변경으로 인한 성능 비교 및 Plan 비교 분석	○	○
	▪ Invalid Object 발생 추이와 Object 정보 분석	○	X

## 2. 한국정보통신기술협회(TTA) BMT 결과

“셀파오라클”은 국가기관 DB모니터링 솔루션 BMT평가 1위 제품입니다.

2021년 TTA DB성능  
모니터링  
소프트웨어 품질 성능  
평가시험(BMT)  
**All PASS!**

- ☑ 모니터링, 장애진단 및 처리, 운영정보 수집, 통계/보고서, 사용자 편의기능 총 22개 항목 모두 통과

DB 모니터링  
솔루션 BMT  
**평가 1위!**

- ☑ 기획재정부 국고보조금 통합관리시스템 분리발주 사업
- ☑ 국방전산정보원 국방군수통합 기반체계 도입 - DB 모니터링 툴
- ☑ 강원랜드 DB 성능 모니터링 솔루션 도입 사업

| 근거 | 조달청 나라장터 입찰정보 물품 '개찰결과 상세조회'

기관명	사업명(대상 DB명)	회사명(제품명)	BMT점수	순위
기획재정부	국고보조금 통합관리시스템 분리발주 대상 SW 도입 (DB모니터링) (Oracle 12c)	<b>셀파소프트 (Sherpa Oracle)</b>	<b>34.7063</b>	<b>1위</b>
		E사 (M-----)	33.9075	2위
		D사 (F-----)	32.6888	3위
국방전산정보원	국방군수통합 기반체계 도입 -DB모니터링 툴 (Oracle 11g)	<b>셀파소프트 (Sherpa Oracle)</b>	<b>45.0365</b>	<b>1위</b>
		E사 (M-----)	44.3890	2위
		S사 (S-----)	43.4496	3위

※ 개찰결과에 대한 역산으로, 숫자 변동이 있을 수 있으나, 비율 산정으로, 순위 변동은 없음

| 근거 | 기획재정부 BMT 결과, Oracle 12c

구분	항목	평가항목	평가결과 (P/P*/F)
기능 확인	데이터베이스 성능 모니터링	통합화면에 데이터베이스 서버 정보 표시 기능 - 시스템(CPU, 메모리), SQL, Session, Lock	P
		성능모니터링의 동작여부를 감시 및 미 동작 시 알림 기능	P
		모니터링 대상 데이터는 모니터링 화면에서 항목별 정렬 기능	P
	장애진단 및 처리	특정 세션을 강제로 종료할 수 있는 기능	P
Disk 공간 임계치 설정 기능 및 임계치 초과 시 알림 기능		P	
Lock 개수 임계치 설정 기능 및 임계치 초과 시 알림 기능		P	
		장애를 유발하는 SQL 및 세션을 검출하여 장애유발 시 알림 기능	P
성능 확인	자원 효율성	<ul style="list-style-type: none"> <li>데이터베이스 서버의 실시간 모니터링 기능 동작 중</li> <li>데이터베이스 서버의 CPU 사용률 측정</li> <li>- 제품설치 전, 설치 후 사용률을 각각 측정</li> </ul>	<b>0.59%</b>

제품 설치 후, CPU 사용률





## IV. 성능이슈 분석 사례

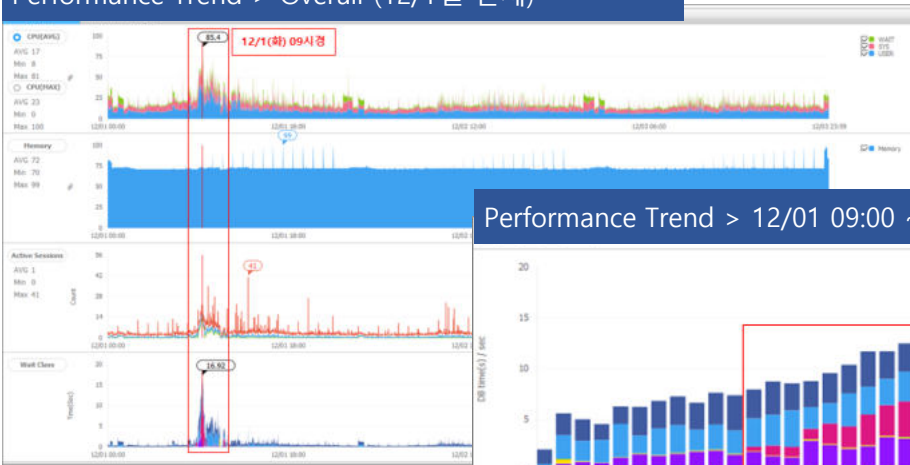
1. 성능이슈 분석 사례
2. 주요 원백 사례

# 1. 성능 이슈 분석 사례 - H공단

## [주요 현상]

- ① 매월 초 음식물쓰레기 배출 정산 시 정산작업이 20 ~ 30여분간 지연되는 현상
- ② 매달 반복되는 문제점으로 인해 업무처리에 지장을 초래함

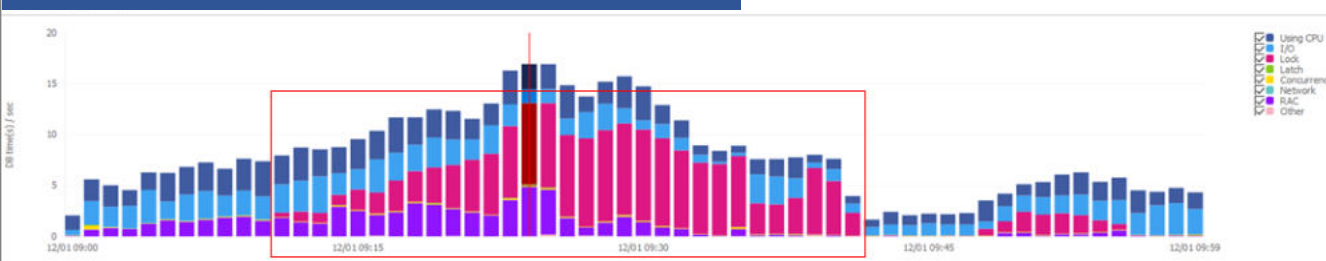
Performance Trend > Overall (12/1일 전체)



## [현상]

09:11 ~ 09:41분까지 CPU 증가, Active Session수 증가  
Lock 대기 발생

Performance Trend > 12/01 09:00 ~ 09:59 Peak 시점 분석



Snapshot	Top process	Session	SQL	ALL SQL	Lock Tree	Wait Detail	Snap(background)
1	enq: TM - contention					400	
2	gc buffer busy acquire					166.94	
3	gc or mtl block request					78.91	

SID	SPID	Serial#	CPU(%)	Username	Module	Program	Logical Reads	Physical Re.	Execute Count	Redo	Wait Time(sec)	Wait #	Wait State	Elapsed Time	SQL	SQL ID
4	21	23526	17239	0	CITYWASTE	JDBC Thin C	JDBC Thin Client	0	0	0	255	enq: TM - contention	WAITING	748	MERGE INTO TN_PW00_CTZN_BLLAMT_NEW BL USIN	6wh2ucca7qj8
5	29	23858	2973	0	CITYWASTE	JDBC Thin C	JDBC Thin Client	0	0	0	252	enq: TM - contention	WAITING	1,227	INSERT /*+ append_values */ INTO TN_PW00_CTZN	ar87mfdkg23
6	1176	23832	2329	0	CITYWASTE	JDBC Thin C	JDBC Thin Client	0	0	0	53	enq: TM - contention	WAITING	52	DELETE FROM TN_PW00_CTZN_BLLAMT_NEW WHERE	168r7nec96nt
7	1713	23846	41549	0	CITYWASTE	JDBC Thin C	JDBC Thin Client	0	0	0	256	enq: TM - contention	WAITING	541	INSERT /*+ append_values */ INTO TN_PW00_CTZN	6cfc48b80mz
8	2292	23522	22136	0	CITYWASTE	JDBC Thin C	JDBC Thin Client	0	0	0	12	enq: TM - contention	WAITING	11	DELETE FROM TN_PW00_CTZN_BLLAMT_NEW WHERE	168r7nec96nt
9	2843	23512	26709	0	CITYWASTE	JDBC Thin C	JDBC Thin Client	0	0	0	256	enq: TM - contention	WAITING	423	DELETE TN_PW00_CTZN_BLLAMT_NEW WHERE BILL_	fs5x5ach8p2r
10	2866	23864	55307	0	CITYWASTE	JDBC Thin C	JDBC Thin Client	0	0	0	126	enq: TM - contention	WAITING	125	DELETE FROM TN_PW00_CTZN_BLLAMT_NEW WHERE	168r7nec96nt

## [현상]

09:11 ~ 09:41분까지 enq:TM - Contention Lock 대기 발생  
특정 테이블에 Insert, Delete 하는 작업이 함께 수행됨

# 1. 성능 이슈 분석 사례 - H공단(계속)

## [주요 현상]

- ① 매월 초 음식물쓰레기 배출 정산 시 정산작업이 20 ~ 30여분간 지연되는 현상
- ② 매달 반복되는 문제점으로 인해 업무처리에 지장을 초래함

SQL Detail : 문제 SQL 에 대한 상세 분석

SQL Full Text

```
1 INSERT /*+ append_values */ INTO TN_PWOOD_CTZN_BLLAMT_NEW
2 (BILL_VYVV,
3 BILL_MM,
4 CTZN_SRNO,
5 CITY_ARCD,
6 APT_CD,
7 BILL_DATE,
8 BILL_KG,
9 BILL_AMT,
10 DSDNT_VN,
11 DSDNT_VNPE,
12 DSDNT_UNIT,
13 DSDNT_PATE,
14 DSDNT_AMT,
15 BEF_BILL_AMT,
16 FIN_BILL_AMT,
17
```

Elapsed Time: 312.5 sec

Lock: 312.2

Statistics Name	Value	Wait Class	Wait Class	Value	Wait Event	Value	Ratio(%)
Elapsed Time	312.54	I/O		0	enq: TM - contention	312.2	100.00
Logical Reads	34	Lock		312.2			
Physical Reads	0	Latch		0			
Redo Entries	0	Concurrency		0			
Execute Count	1	Network		0			
Hard Parse	0	RAC		0			
Cpu Time	0.34	Other		0			
Elapsed Time / Exec.	312.54						

**[원인]** 문제가 되는 테이블에 Insert 시 /\*+ Append\_Values \*/ 힌트를 사용하여 direct path load 발생  
direct path load는 해당 테이블에 TM 락을 exclusive 하게 획득하기 때문에 Insert 작업이 완료될때까지 해당 테이블에 대한 모든 DML들은 Lock 대기

**[해결]** Insert문에 있는 /\*+ Append\_Values \*/ 힌트 제거

## 2. 주요 원백 사례

고객사	사업명	도입 규모	도입 일자	원백 제품	선정 사유
현대기아차	DB 성능관리 도입	전사	2019.11	맥스게이지	<ol style="list-style-type: none"> <li><b>현기차, LG, KT 등 다수의 대형 고객사에서 원백한 안정적인 아키텍처</b> <ul style="list-style-type: none"> <li>- 수집서버로 성능데이터를 즉시 전송하여 I/O 대기없는 분석</li> <li>- 대상 Agent 서버의 자원을 최소 사용하여 안정적인 모니터링 (TTA 및 POC,BMT 결과로 검증)</li> </ul> </li> <li><b>직관적인 UI/UX를 통해 문제상황을 한눈에 확인</b></li> <li><b>누락 없는 성능 데이터 수집</b> <ul style="list-style-type: none"> <li>- Active Session 1,000 개 이상의 DB 장애 상황에서도 누락 없는 성능데이터 수집</li> <li>- 수집 서버 장애나 Network 단절 시 유실방지 기능으로 성능데이터 누락 방지</li> </ul> </li> <li><b>정밀한 성능 데이터 수집</b> <ul style="list-style-type: none"> <li>- 1/100초 주기로 정밀한 데이터 수집</li> <li>- 타사제품과는 달리 Session 및 SQL 정보 모두 수집하여 정밀한 분석</li> </ul> </li> <li><b>실시간 성능 데이터 분석</b> <ul style="list-style-type: none"> <li>- 1초 이후 즉시 분석 가능 (차별화된 아키텍처)</li> <li>- 넓은 범위 분석기간 조회 시 가장 빠른 분석</li> </ul> </li> </ol>
LG U+	DB 성능관리 도입	전사	2016.10		
LG 전자	DB모니터링 SW 납품 및 설치	전사	2012.11		
현대카드 /캐피탈 /커머셜	현대카드 캐피탈 커머셜 차세대 통합모니터링 APM DB 솔루션 도입	전사	2013.10		
롯데카드	롯데멤버스 차세대시스템 구축 프로젝트	전사	2013.06		
삼성카드	IT인프라 모니터링 솔루션 도입	전사	2017.12		
호텔신라	IT 인프라 모니터링 솔루션 도입	전사	2016.12		
LG 디스플레이	IT 인프라 모니터링 솔루션 도입	전사	2015.08	Precise for Oracle / Stable For Oracle	
KT	차세대 및 Legacy DB APM 시스템도입	전사	2012.06		



## V. 셀파 지원 환경

1. 수집서버 권장사양
2. 수집서버 용량산정

# 1. 수집서버 권장사양

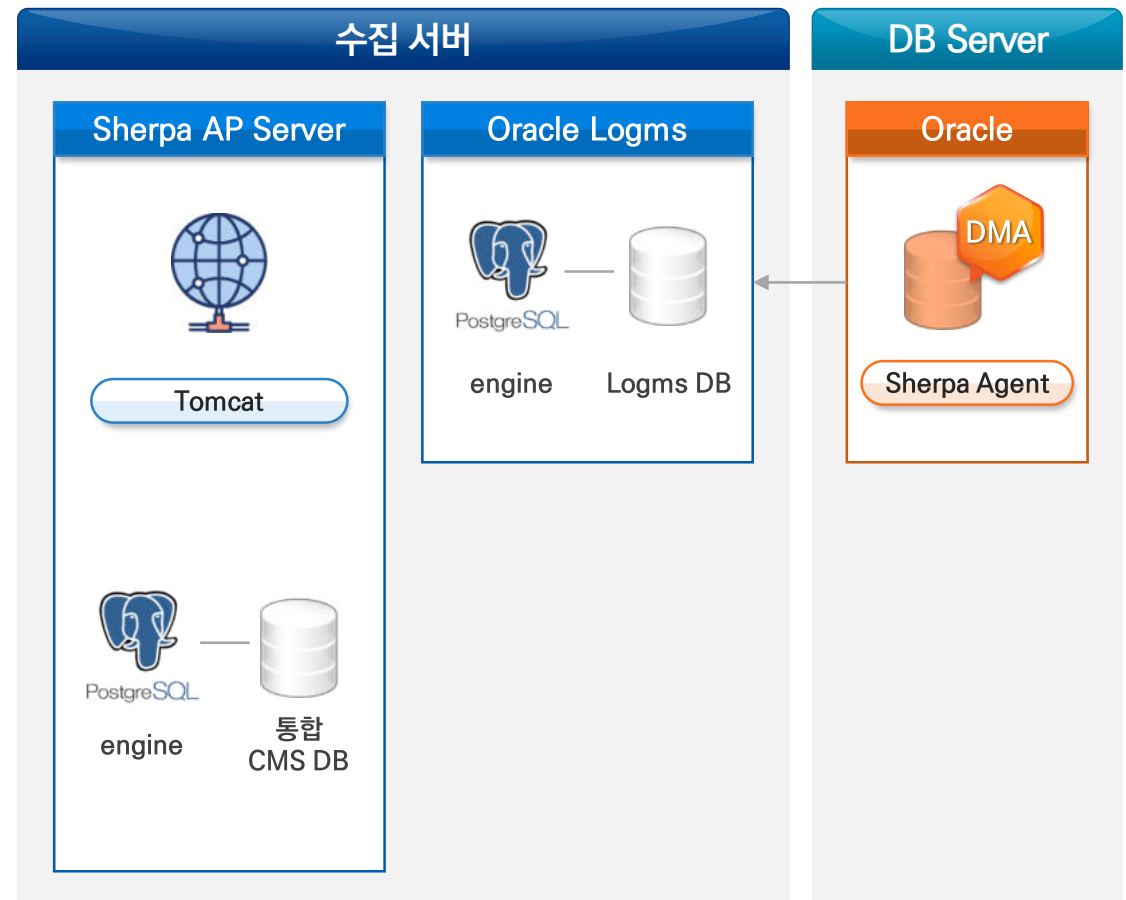
- 성능정보를 저장하는 테이블들을 파티션(10일/월)으로 구성하여 Repository 서버에서 발생할 수 있는 부하를 최소화하여, 수집기간에 상관없이 사용자에게 동일한 응답속도를 제공합니다.

## 수집서버 지원 DB 버전 및 OS

구분	세부 내용
Database	PostgreSQL v12 ( Community 버전 )
OS	Linux 64bit ( 권장 ) / Windows 64bit

## 수집서버 Spec

항목	권장 사양
CPU	4Core (2.9Ghz)
Memory	16GB
HDD	15K RPM SAS / 7200 RPM SATA
적용 Instance 수	20개 DB 기준
데이터 저장 공간	1000GB



## 2. 수집서버 용량산정

### [필요공간 산정 기준]

- 1개월 보관
- 1개 Instance 모니터링
- 평균 active session 20개

보관기간	모니터링 대상	DB Size	File Size	합계
1개월	1개 인스턴스	25 GB	5 GB	30 GB

### [20개 Instance 시, 보관 기간 별 필요 공간]

보관기간	모니터링 대상	DB Size	File Size	합계
1개월	20개 인스턴스	500 GB	100 GB	600 GB
2개월	20개 인스턴스	1000 GB	200 GB	1,200 GB
3개월	20개 인스턴스	1,500 GB	300 GB	1,800 GB
4개월	20개 인스턴스	2,000 GB	400 GB	2,400 GB

# Q & A





# 감사합니다

(주)셀파소프트 [www.sherpasoft.com](http://www.sherpasoft.com)

서울특별시 금천구 벚꽃로278(가산동, SJ 테크노빌) 813호

T 02-3273-0380 E [sales@sherpasoft.com](mailto:sales@sherpasoft.com)

 [facebook.com/sherpasoft](https://facebook.com/sherpasoft)

 [instagram.com/sherpasoftofficial](https://instagram.com/sherpasoftofficial)

 [blog.naver.com/sherpasoft](http://blog.naver.com/sherpasoft)

 [linkedin.com/company/sherpasoft](https://linkedin.com/company/sherpasoft)

 셀파소프트