

# MaxGauge for Java

## End to End Monitoring Solutions

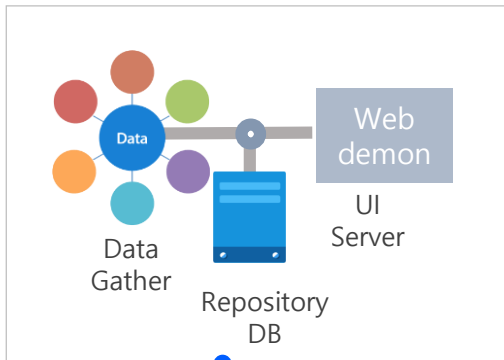
The Next Revolution is Here

EXEM CO., LTD.

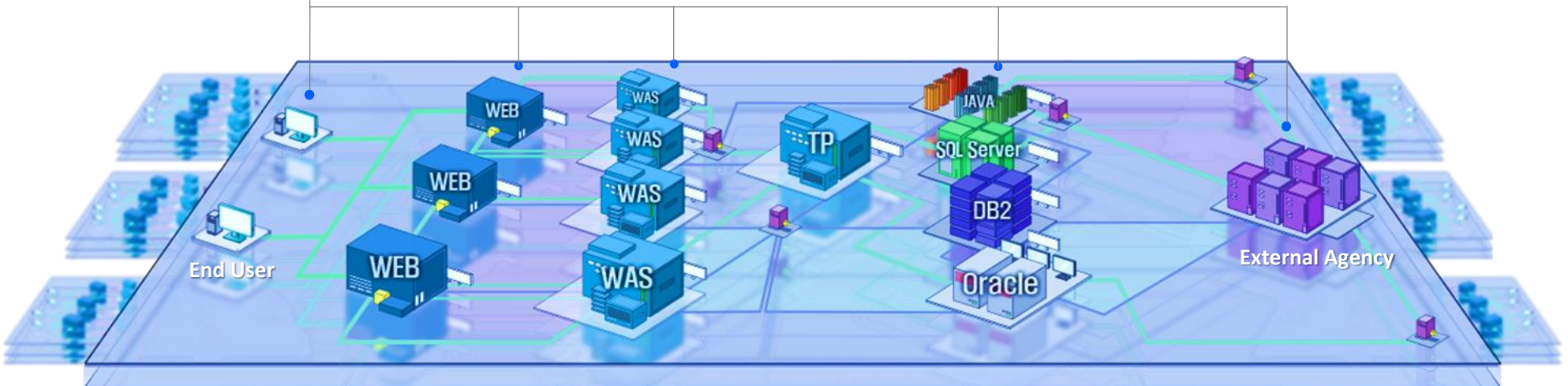


## What is MFJ's End-to-End Performance Management?

End-to-End Transaction Performance Management in a Complex Business Service Environment Composed of N-Tiers  
Integrated Performance Management of Transactions Across the Entire System From the User Terminal (Browser) to WEB~WAS~TP~DB



**"Real-time Monitoring Feature"** identifies in which section out of many servers the error or performance degradation was generated from the transaction point of view.  
**"Real-time Diagnosis"** helps to quickly identify the root cause of error/performance delays.  
**"Post Analysis Feature"** maximizes the stability of system operation by providing preventative measures.

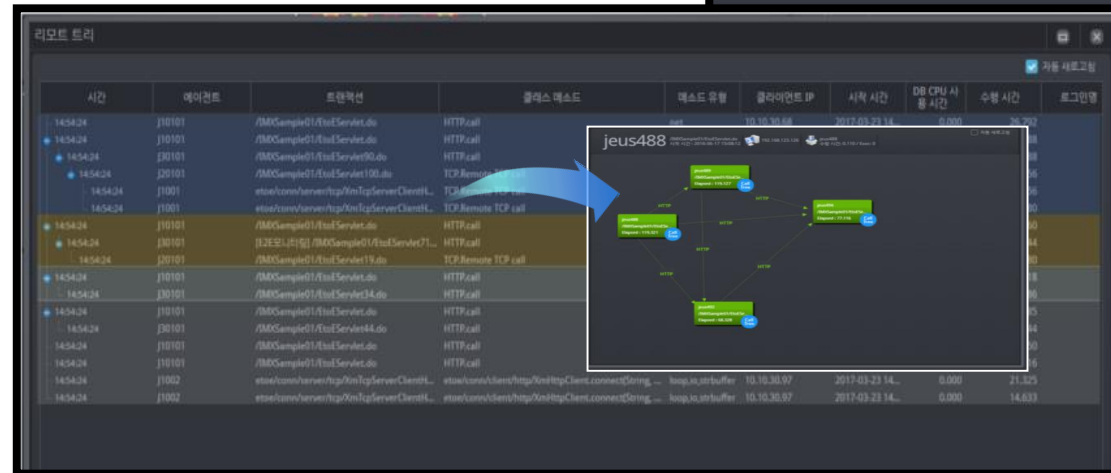
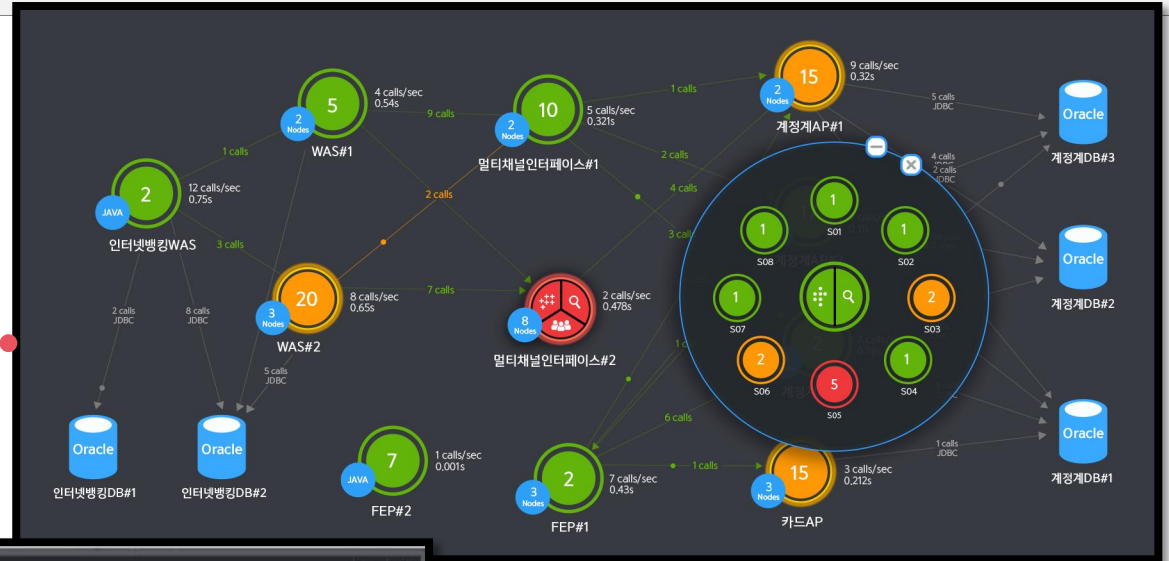


## (Special Features) – Topology view, Real-time Remote Tree

- Provides Topology View – Easily identify the transaction process between nodes across the entire system and quickly detect errors through real-time events (Useful)
- Provides Remote Tree – With a call tree view of a transaction being processed in real-time, identify in which section the performance delay is taking place (For Identifying External Institution Delays/Errors)

### [Topology View]

- Monitoring of real-time transaction situation of the entire system (Time spent between nodes, number of transaction processed between nodes, grouping)
- Provides alarm generation status and detailed analysis by each node (group) by linking with real-time events.

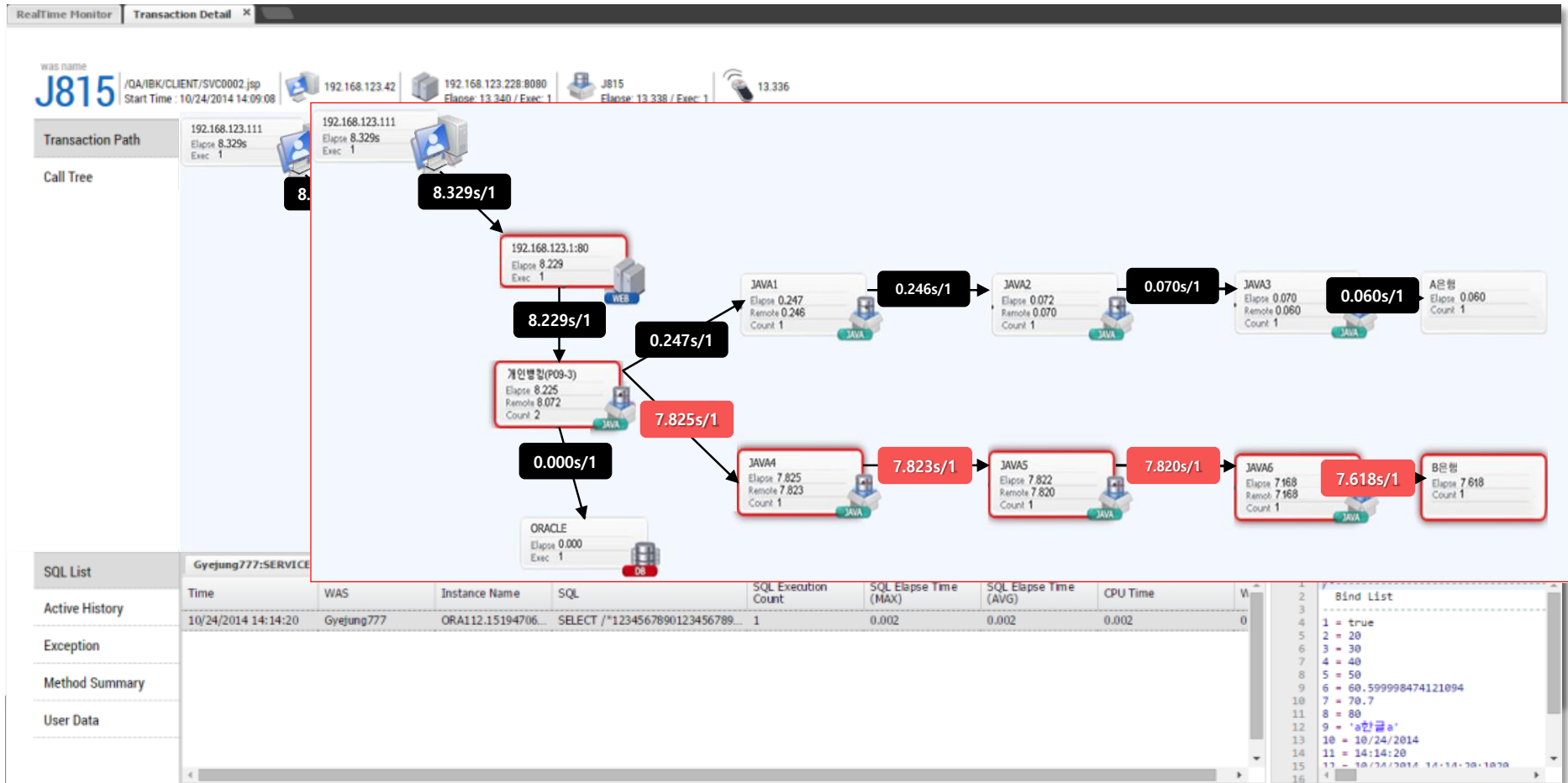


### [Remote Tree]

- With the call tree view of an active transaction, you can intuitively identify the section in which the AP level's performance delay is taking place.
- Displays the status of each transaction's process between nodes with a call tree view.

## (Special Features) – Transaction Path view

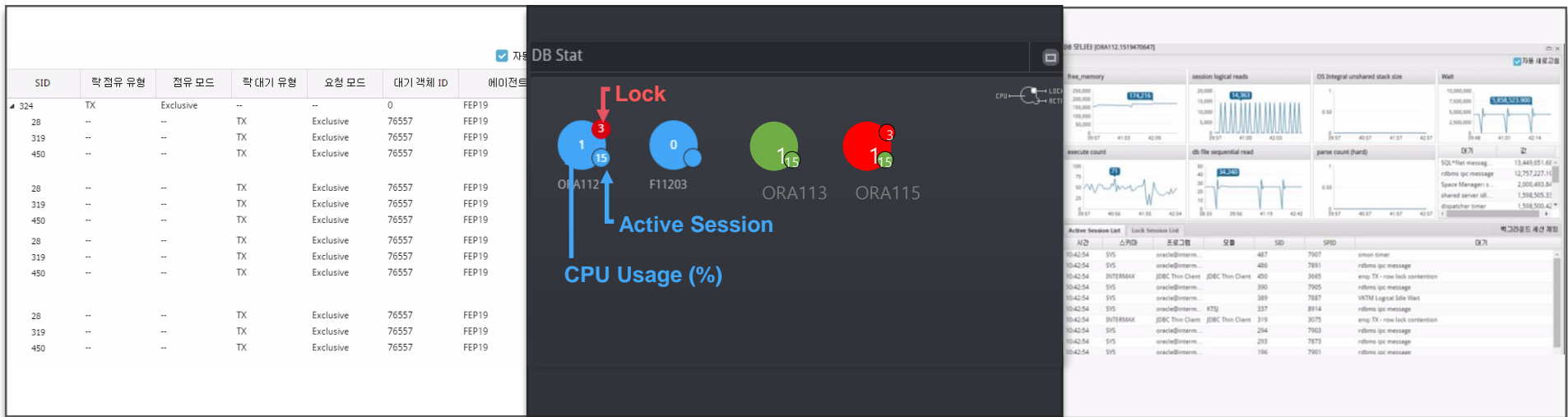
- Transaction Path View – Allows you to intuitively identify the delay section by displaying each transaction's end-to-end flow and elapsed time between nodes.
- A separate icon is used to represent each section from user (CS, Browser) – WEB – WAS – TP – DB – Remote.



(Special Features) – WAS + DB Connection Monitoring

# DB Platform Performance Monitoring with a WAS+DB Integrated Performance Management Tool

In the event of an individual SQL's performance delay, you can use the [DB Connection Monitoring View](#) to quickly diagnose performance up to the DB platform.



## DB Lock Tree Details Monitoring

Provides lock information in a tree structure of holder and waiter sessions.

## DB Instance Monitoring

Displays monitoring DB's CPU usage, lock generation and main stats Displays color according to threshold values set in the **Alert Configuration**

- Blue: Normal
- Orange: Warning
- Red: Critical

## Specific DB Instance Details Monitoring

- Provides real-time graphs of DB's main stats and waits
- Displays detailed information about active sessions and lock sessions.

(Special Features) – WAS + DB Connection Monitoring

- Check DB session info on WAS active transaction list
- Monitor by connecting the DB Lock Holder and Waiter to WAS transaction
- View DB execution time and workload and wait status info at a glance.



# WAS Transaction and DB Session Connected Analysis

DB Session Info						WAS Transaction							
SID	Hold Lock Type	Hold Mode	Wait Lock Type	Request Mode	Wait Object ID	Agent	Transaction	Class Method	Client IP	Log... Na...	Start Time	CPU Time	Elaps... Time
137	TX	Exclusive	--	--	0	vmware_jeus_agent	/IMX_Test/DB_Lock.do	test_InterMax/dblock.do...	192.168.123.87		19:34:25	0.000	16.820
153	--	--	TX	Exclusive	52519	vmware_jeus_agent	/IMX_Test/DB_Lock.do	oracle/jdbc/driver/Oracl...	192.168.123.87		19:34:11	0.961	30.479
141	--	--	TX	Exclusive	52519	vmware_jeus_agent	/IMX_Test/DB_Lock.do	oracle/jdbc/driver/Oracl...	192.168.123.87		19:34:11	1.369	30.477
131	--	--	TX	Exclusive	52519	vmware_jeus_agent	/IMX_Test/DB_Lock.do	oracle/jdbc/driver/Oracl...	192.168.123.87		19:34:26	0.324	15.601
146	--	--	TX	Exclusive	52519	vmware_jeus_agent	/IMX_Test/DB_Lock.do	oracle/jdbc/driver/Oracl...	192.168.123.87		19:34:19	1.071	22.721

Transaction generated DB Lock Holder and Waiter info

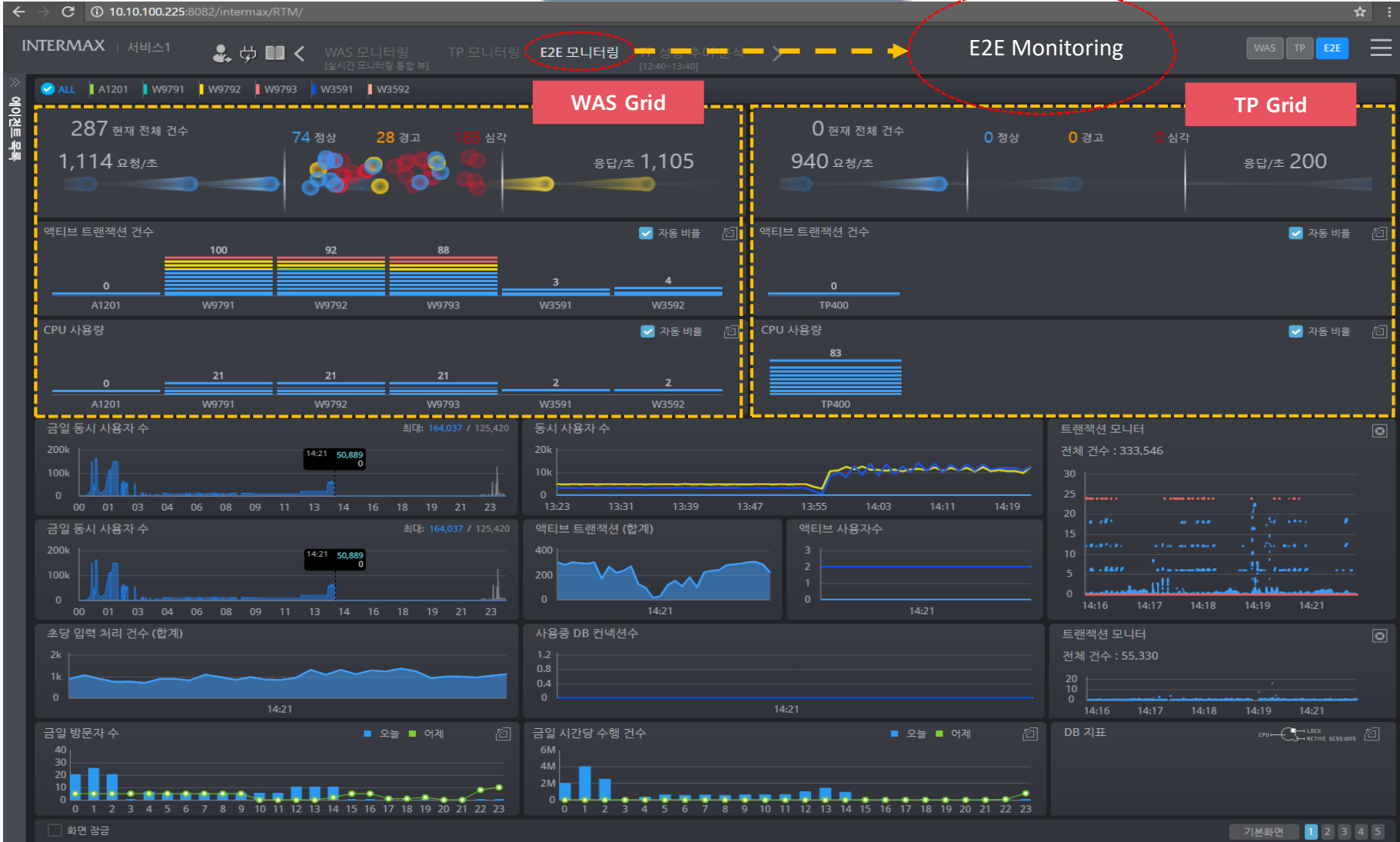


WAS Transaction										DB Session Info					
Agent	Transaction	Class Method	Elapse Time	DB Time	Wait Time	Pool	SID	State	SQL 1	SQL Execution Count	Fetch Count	Prepare Count	PGA Usage	Logical Reads...	
MCA104	[MCA] ibk/jdg/server/cs/Rea...	ibk/xm/testcase/XmSleep.sleep(long)	0.010	0.000	0.000		0	SLEEPING		0	0	0	0	0	
MCA104	[MCA] ibk/jdg/server/cs/Rea...	ibk/xm/testcase/XmSleep.sleep(long)	0.037	0.000	0.000		0	SLEEPING		0	0	0	0	0	
Gyejung...	SERVICE=GYE_SVC0001	ibk/xm/testcase/XmSleep.sleep(long)	0.208	0.000	0.000	IBKTEST	430	SLEEPING	SELECT /*1234567890...	1	0	1	0	0	
Gyejung17	SERVICE=GYE_SVC0001	ibk/xm/testcase/XmSleep.sleep(long)	0.421	0.000	0.000	IBKTEST	444	SLEEPING	SELECT /*1234567890...	1	0	1	0	0	
JEUS815	/QA/IBK/CLIENT/SVC0001.jsp	ibk/xm/client/XmClient.connect(byte)	2.038	0.000	0.000		0	NETWORK_IO		0	0	0	0	0	
JEUS815	/QA/IBK/CLIENT/SVC0001.jsp	ibk/xm/client/XmClient.connect(byte)	2.314	0.000	0.000		0	NETWORK_IO		0	0	0	0	0	

Transaction executed SQL's DB workload info

(Feature) Real-time Monitoring – E2E Monitoring Main Screen

Real-time E2E Monitoring - Screen



(Feature) Real-time Monitoring – WAS Monitoring Main Screen

Real-time WAS Monitoring – Screen





(Feature) Real-time Monitoring – TP Monitoring Main Screen

Real-time TP Monitoring - Screen

The screenshot displays the INTERMAX TP Monitoring interface. The navigation menu on the left includes '호스트 목록', 'OVERALL', 'abx71', and 'TP400'. The main dashboard shows the following metrics:

- 현재 전체 건수: 0
- 요청/초: 0
- 응답/초: 0
- 액티브 트랜잭션 건수: 0
- CPU 사용량: 11
- 초당 처리량: 0-300
- 응답 시간: 0-1
- 프로세스 수: 0-20
- 트랜잭션 모니터: 전체 건수 : 39,889
- 클라이언트 수: 0-4k
- 큐잉 수: 0-20
- 큐잉 시간: 0-1

The 'TP Slog' table shows the following error logs:

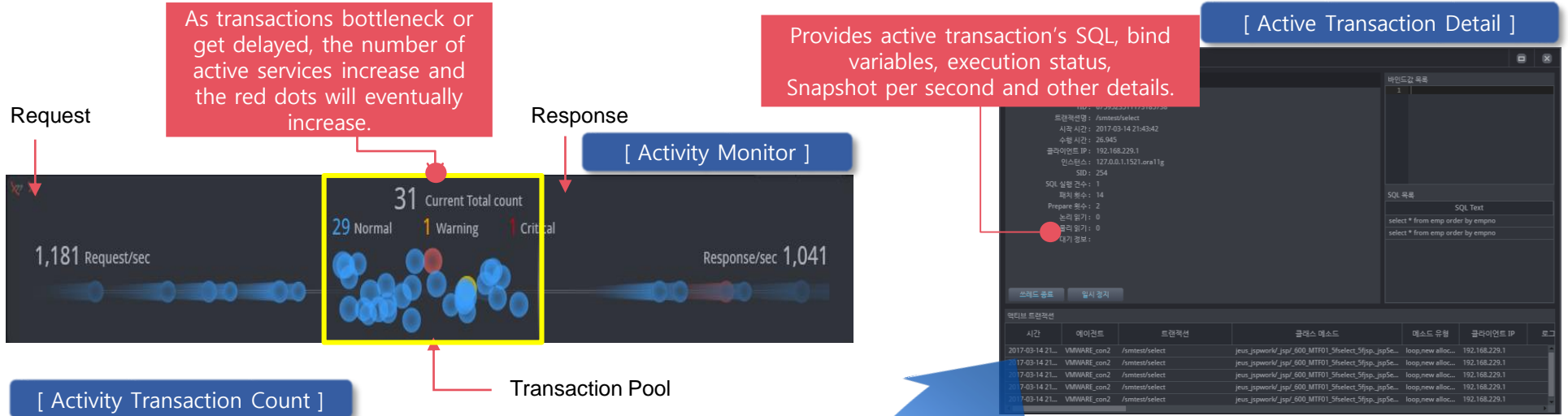
시간	Node Name	Proc. Name	Proc. IDS	Error Level	Code	Message	Service Code
03-31 17:16:41	TP400	TMS		Error	TMS0205	xa_recover error : -3	TMS2038
03-31 17:16:41	TP400	TMS		Error	TMS0213	transaction recovery failure	TMS3100
03-31 17:16:41	TP400	TMS		Error	TMS0213	transaction recovery failure	TMS3100
03-31 17:15:44	TP400	TMM		Error	TMM0023	write error: tproc (CLH) closed, msgtype = 158	TMM0027

The '실시간 tadmin' table shows the following active transactions:

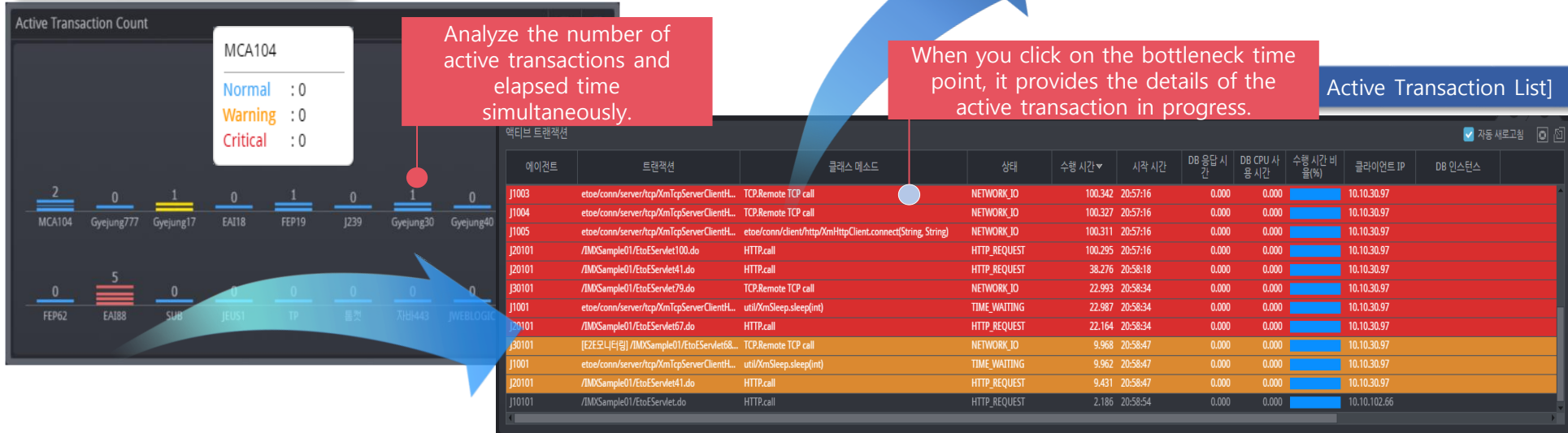
time	agent	clhno	svrname	pid	statu	count	average
03-31 17:18:10	TP400	0	sdltest	21364960	1	0	0.000
03-31 17:18:10	TP400	0	fdltest	3276864	1	376	0.000
03-31 17:18:10	TP400	0	mxtinfo	5636318	1	11	0.000
03-31 17:18:10	TP400	0	svr2	9699554	1	7	0.000
03-31 17:18:10	TP400	0	svr2	9502958	1	8	0.000

**(Feature) Real-time Monitoring – Only the active transactions that are currently in progress are retrieved on the Active Transaction List (1/2)**

- Active Monitor : Real-time request process performance monitoring / Identify delays in the Transaction Pool
- Check detailed information on the delayed transactions.



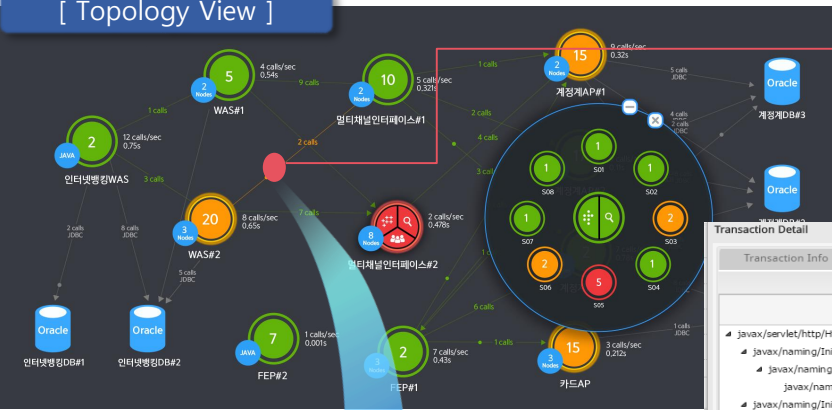
**[ Activity Transaction Count ]**



## (Feature) Real-time Monitoring – View the total system call tree of the active transactions currently in progress.

- Provides class elapsed time executed in the transaction and the method type, execution count and other information.
- Provides the call-tree statistics data by each section for the entire system and can be viewed at a glance.

### [ Topology View ]



On the Real-time Topology View, it provides 'Remote Tree' and the 'Real-time Pathview' which can trace the active transaction's call information throughout the entire system in real-time

### [ Remote Tree ]

시간	에이전트	트랜잭션
14:54:24	J10101	/IMXSample01/EtoEServlet.do
14:54:24	J10101	/IMXSample01/EtoEServlet.do
14:54:24	J30101	/IMXSample01/EtoEServlet90.do
14:54:24	J20101	/IMXSample01/EtoEServlet100.do

Provides analysis of the root cause by connecting with real-time diagnostics screen for the transactions in which problems have been generated such as locks or elapsed time delays.

### [ Call Tree ]

Transaction Detail

Class	Method	Exception Count	Execute Count
javax/servlet/http/HttpServletRequest	service(HttpServletRequest, Http...	0	0
java/naming/InitialContext	init(Hashtable)	0	0
java/naming/InitialContext	getDefaultInitCtx()	0	0
java/naming/spi/NamingManager	getInitialContext(Hashtable)	0	0
java/naming/InitialContext	getURLDefaultInitCtx(String)	0	0
java/naming/spi/NamingManager	hasInitialContextFactoryBuilder()	0	0
java/naming/InitialContext	getURLScheme(String)	0	0
java/naming/InitialContext	getDefaultInitCtx()	0	0
java/naming/NameImpl	recordNamingConvention(Prop...	0	0
java/naming/NameImpl	toBoolean(String)	0	0
java/naming/NameImpl	extractComp(String, int, Vect...	0	0
java/naming/NameImpl	isA(String, int, String)	0	0

Method Parameter

Method	Parameter	Return
Bind Value List	1	
SQL List	SQL Text	SELECT * FROM test_dummy.t WHERE 1 = 1 and...

### [ Thread Dump ]

Transaction Detail

Active Transaction

Time	Agent	Transaction	Class	Method	Method Type	Client IP	Login N
2015-04-07 09:00	SUB	/IMX_Test/Sleep_Test.do	javax/servlet/http/HttpServlet	service(HttpServletRequest, Http...	net	192.168.123.42	

Method Parameter

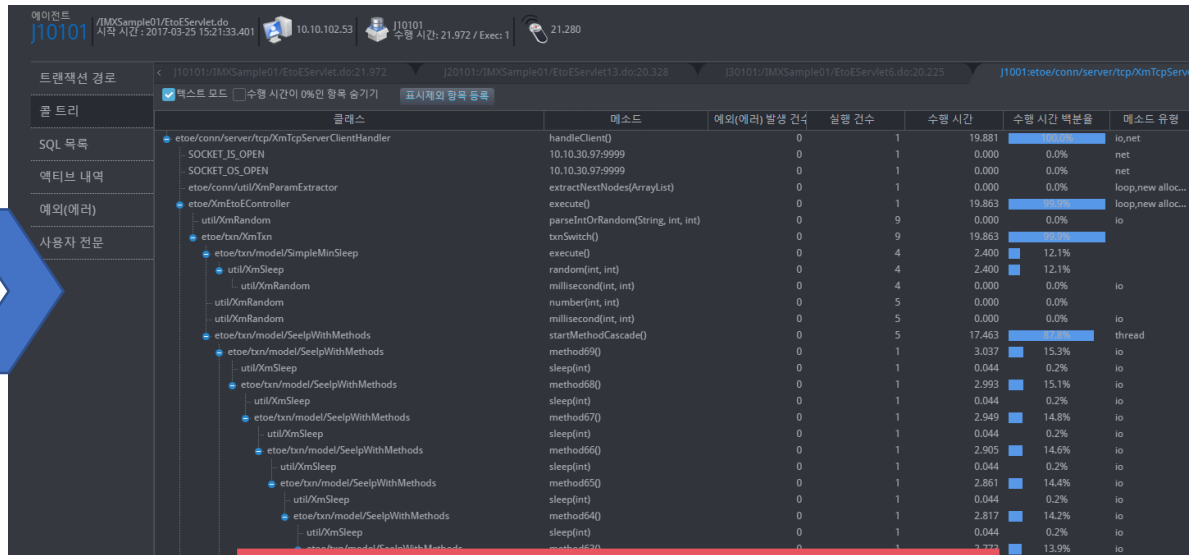
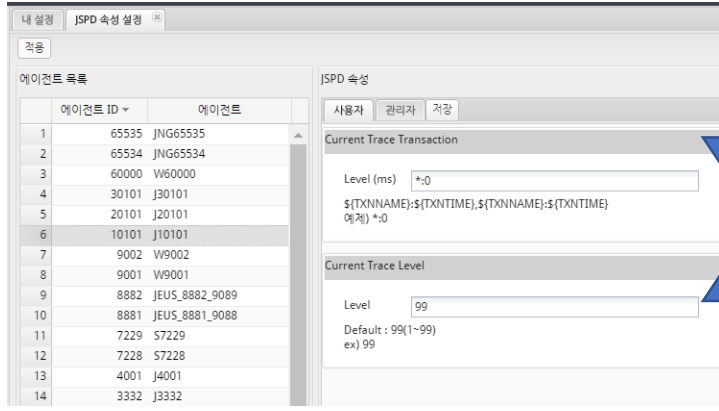
Method	Parameter	Return
Bind Value List	1	
SQL List	SQL Text	SELECT * FROM test_dummy.t WHERE 1 = 1 and...



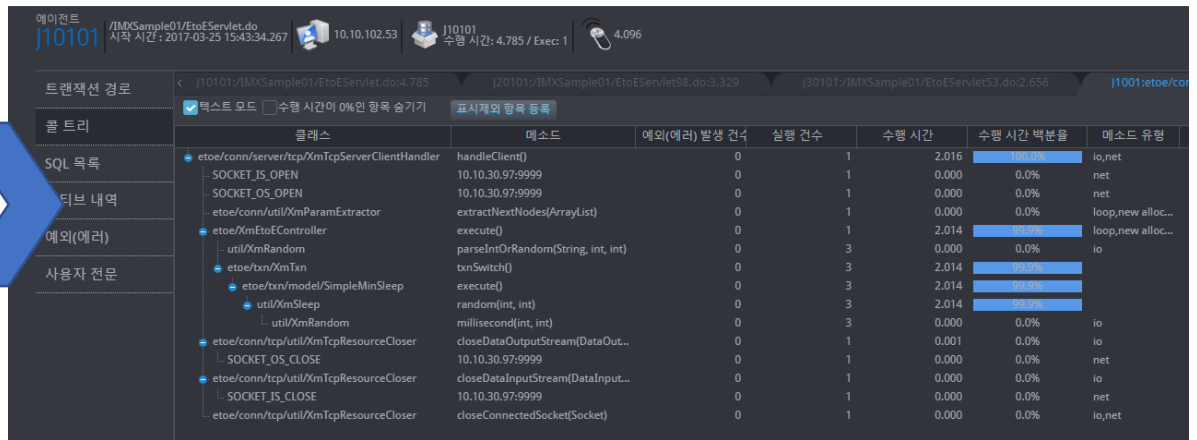
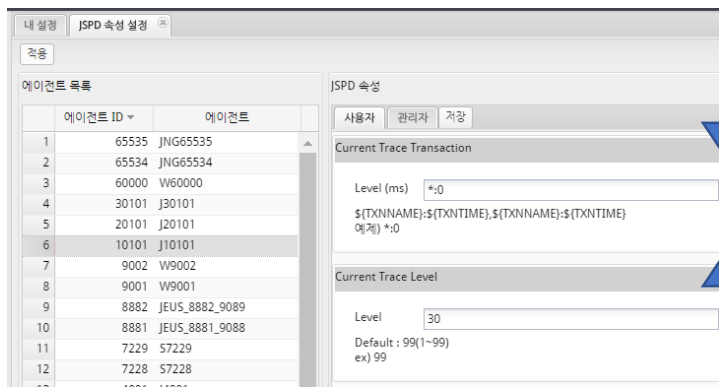
**(Feature) Real-time Monitoring – Adjust analysis level (depth or top/bottom structure information) dynamically without process restart.**

- Make changes to the call tree's collection depth and apply in real-time through environment settings for each agent.

[ Trace Level Setting 99(Max) ]



[ Trace Level Setting 30 ]



Adjust the call tree's collection level and apply in real-time.

**(Feature) Real-time Monitoring – When an error occurs, the root cause of the problem must be clearly indicated.**

- When a transaction error or delays occur, it displays each individual transaction's execution structure in a topology format for an easy recognition of the problem area.
- Provides if any error detected in the corresponding section, stack trace info for the error, SQL execution performance data for a detailed analysis of the root cause of the problem.

[ Transaction Detail Features ]

[ SQL Execution Data ]

SQL 바인드 값 적용된 SQL

```

1 /*
2 Bind Value List
3
4 1 = 'SMITH'
5 2 = 'CLERK'
6 3 = 7902
7 4 = 1000
8 5 = 2000
9 6 = 20
10 7 = 7369
11 -----*/
12
13
14 update emp set ename=:1, job=:2, mgr=:3, sal=:4, comm=:5, deptno:

```

SQL 문 검색

로그

```

1 java.lang.NullPointerException
2
3 at jdbc.XmDbcUseConnectionPool.executeSimpleLiteral(XmDbcUseConnectionPool.java:141)
4 at etoe.txn.model.DbPoolSyntaxError.execute(DbPoolSyntaxError.java:11)
5 at etoe.txn.XmTxn.txnSwitch(XmTxn.java:54)
6 at etoe.txn.XmTxn.startIoThread(XmTxn.java:30)
7 at etoe.XmEtoController.execute(XmEtoController.java:67)
8 at servlet.XmEtoServlet.doPost(XmEtoServlet.java:69)
9 at javax.servlet.http.HttpServlet.service(HttpServlet.java:725)
10 at javax.servlet.http.HttpServlet.service(HttpServlet.java:818)
11 at jeus.servlet.engine.ServletWrapper.executeServlet(ServletWrapper.java:242)
12 at jeus.servlet.engine.ServletWrapper.execute(ServletWrapper.java:206)
13 at jeus.servlet.engine.HttpServletRequestProcessor.run(HttpServletRequestProcessor.java:319)

```

Error generated section is indicated with a red X mark for distinction.

Filters the delayed transactions with red so it can be recognized instantly.

[ Error Stack Trace ]

## (Feature) Transaction Trace – Provides a structure map (graph, line, graph and more) of each transaction’s flow throughout the entire system. (2/2)

- Provides not only the transaction trace on the analysis screen but the real-time transaction trace as well.
- View the entire connected system through the topology view and **real-time section analysis** is possible through real-time transaction trace.



Provides a topology view which allows you to see the transaction flow between the system nodes and intuitively identify the delay sections.

- Real-time Active Transaction Lists Monitoring
- Provides Transaction Monitor through the Tx-Viewer



[ Real-time Path View ]

A unique solution which can trace transactions that are currently in progress, not completed transactions.

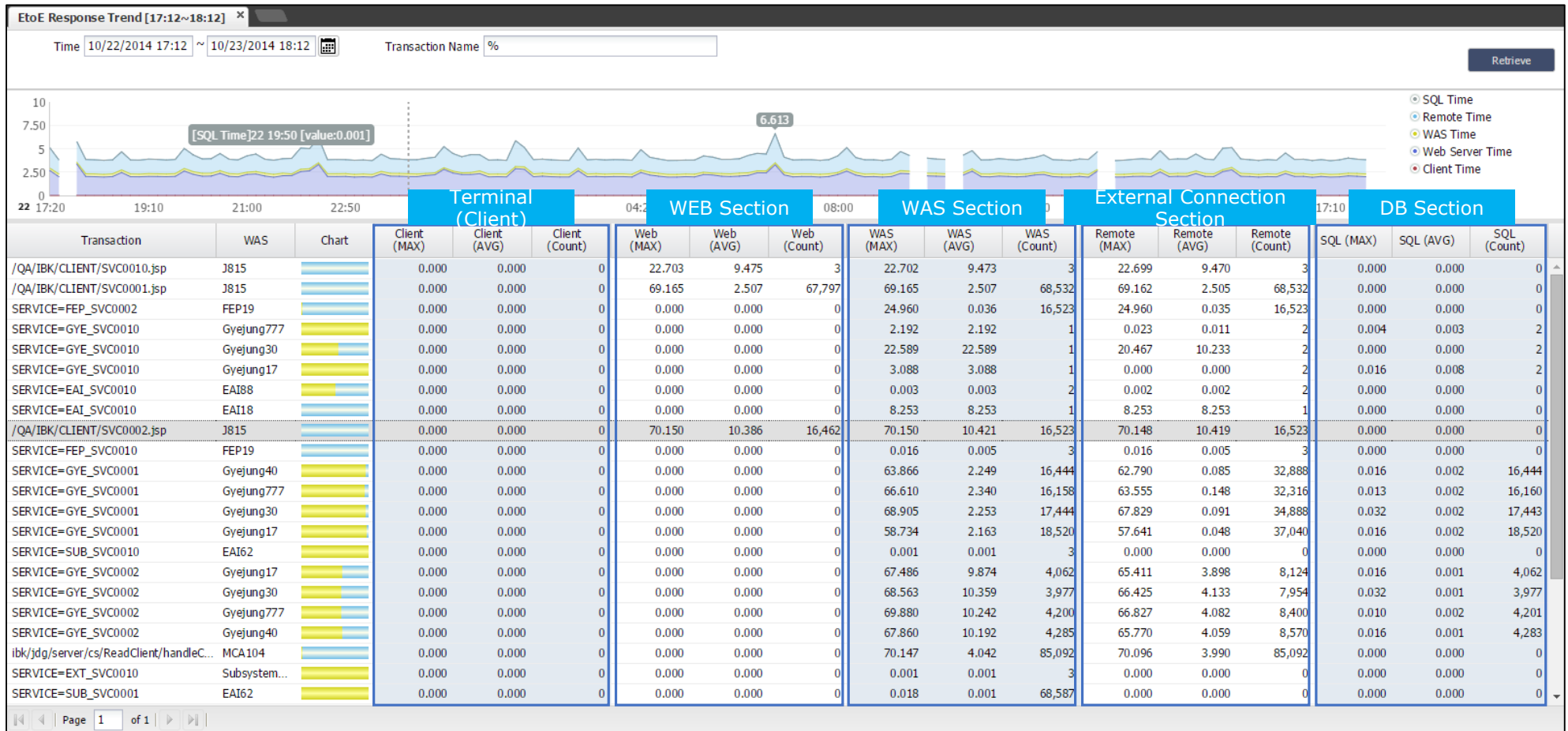


according to tasks so they can be managed as a group, and you can even view by each node in detail for each group.

# MFJ E2E Main Features

**(Feature) Transaction Trace – Provides the elapsed time for each section. (By each task server, web server, WAS server, TPM server, daemon, DB server, and more)(1/3)**

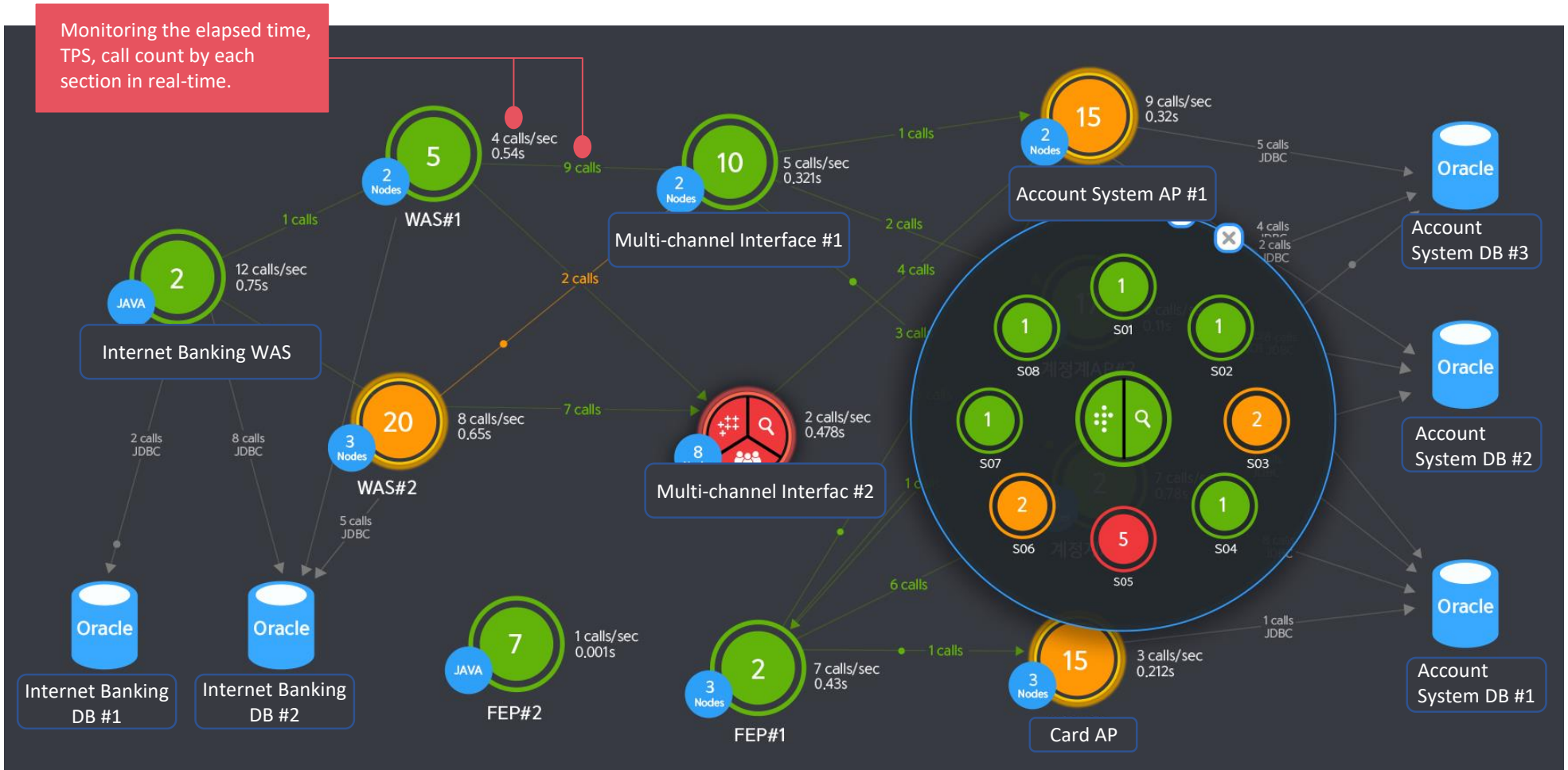
- Check transaction elapsed time by each End-to-End section.
- Based on the transaction name, identify the bottlenecks by measuring the elapsed time in each section from Client–Web Server–WAS–DB or even the external requests.



## Middleware Monitoring (End to End) Features Requirement

(Feature) Transaction Trace – Provides the elapsed time by each section (By each task server, web server, WAS server, TPM server, Daemon, DB server, and more)(2/3)

- Provides the average elapsed time and the call count through the real-time dashboard screen.





# MFJ E2E Main Features

**(Feature) Transaction Trace – Provides the elapsed time by each section (By each task server, web server, WAS server, TPM server, Daemon, DB server, and more) (3/3)**

- Provides not only the real-time elapsed time by each section but also instant connection for *Real-time Diagnostic Analysis* by each individual active transaction.



[ Displays Real-time Event Alarms ]

1 calls/sec  
3.292 s

JAVA  
jeus90

jeus90	Active Transactions	5	WARNING
jeus90	Concurrent Users	30	CRITICAL
jeus90	Elapsed Time	6.366	CRITICAL

Displays alarm info according to threshold values set for elapsed time, number of concurrent users, active transactions, TPS, and more.

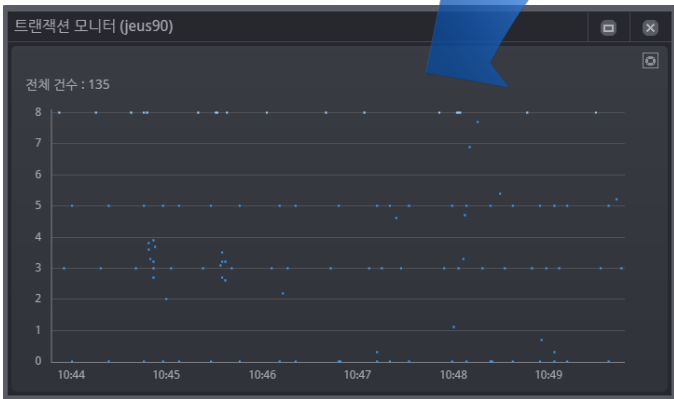


[ Active Transaction ]

액티브 트랜잭션 (jeus88\_PP → 10.10.30.110.1521.intermax)

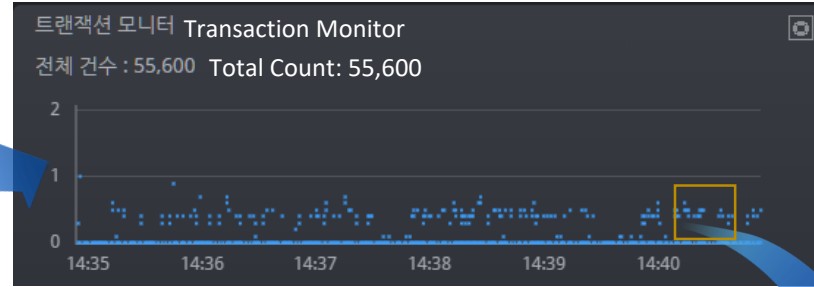
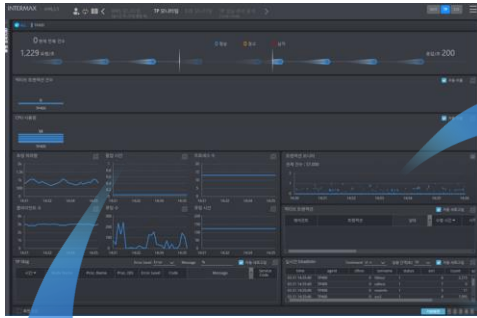
아이전트	트랜잭션	클래스 메소드	상태	수행 시간	시각
jeus88_PP	/IMX_Test/Literal_Insert.do	oracle/jdbc/driver/OraclePreparedStatement.executeUpdate()	STM1_EXECUTE	3.060	10:47:2

[ Transaction Monitor ]

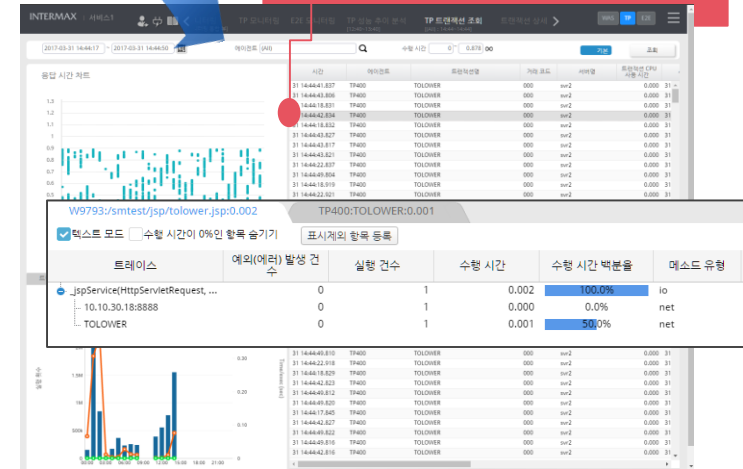


## (Feature) Transaction Trace – Main performance stats and real-time monitoring through TP monitoring

- Provides TP monitoring's real-time main performance stats
- Provides performance stats monitoring equivalent to Tmadmin management commands
- Provides real-time error message monitoring through Slog collection



Analyze the call tree of delayed services on the transaction detailed analysis screen by dragging the analysis section.



Provides 10 TP main performance stats (Menu selection screen can be configured)

지표 변경 Stat Change

초당 처리량

- aq 개수
- 실패 건수
- 에러 건수
- 응답 시간
- 초당 입력 처리 건수
- 초당 처리량
- 큐잉 수
- 큐잉 시간
- 클라이언트 수
- 프로세스 수

Check Performance stats equivalent to tmadmin manager tool

실시간 tmadmin Real-time tmadmin Command st-v 실행 간격(초) 10 자동 새로고침

time	agent	chno	svname	status	svri	count	qcount	emcount
03-31 14:57:20	TP400	0	sdctest	1	7	0	0	0
03-31 14:57:20	TP400	0	fdctest	1	6	1,276	0	0
03-31 14:57:20	TP400	0	mxtinfo	1	5	11	0	0
03-31 14:57:20	TP400	0	svr2	1	4	5,883	0	0

TP Slog Error Level Error Message % 자동 새로고침

시간	Node Name	Proc. Name	Proc. IDS	Error Level	Code	Message	Service Code
03-31 17:06:37	TP400	TMS		Error	TMS0205	xa_recover error : -3	TMS2038
03-31 17:06:37	TP400	TMS		Error	TMS0213	transaction recovery failure	TMS3100
03-31 17:06:21	TP400	TMM		Error	TMM0023	write error: tproc (CLH) closed, msgtype = 158	TMM0027
03-31 17:00:19	TP400	TMS		Error	TMS0205	xa_recover error : -3	TMS2038
03-31 17:00:19	TP400	TMS		Error	TMS0213	transaction recovery failure	TMS3100

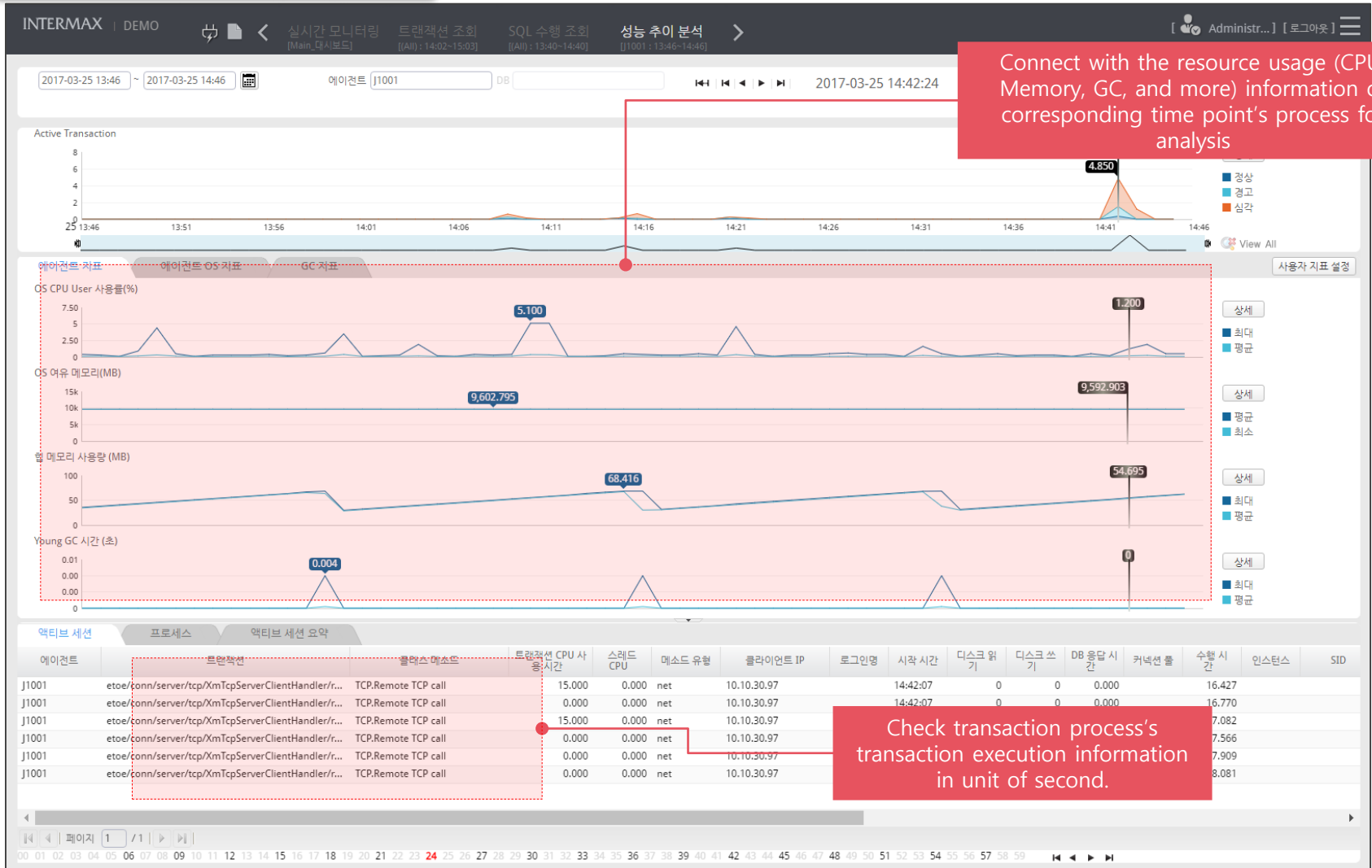
In the event of a TP slog log generation, monitor with real-time collection.

# MFJ E2E Main Features

**(Feature) Transaction Trace – Transaction process’s CPU, Memory, GC and other information are matched up with transaction’s information and are displayed simultaneously.**

- Execution information of a section in which a specific time point’s transaction is being executed can be connected with the corresponding process’s resource usage information for analysis.

[ Performance Trend Analysis Feature ]



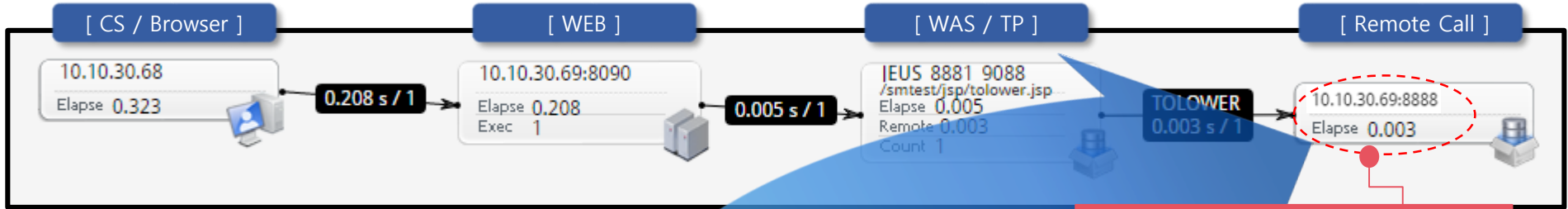
Connect with the resource usage (CPU, Memory, GC, and more) information of corresponding time point's process for analysis

Check transaction process's transaction execution information in unit of second.

# MFJ E2E Main Features

**(Feature) Transaction Trace – Provides IP, port, elapsed time information of the section in which the monitoring product’s agent (collection process) is not installed in the next generation target system.**

- Provides 'call pathway (call service)', 'IP', 'Port', 'elapsed time' information of remote call section in which an agent is not installed.
- Provides call information details in Call Tree and Path View.

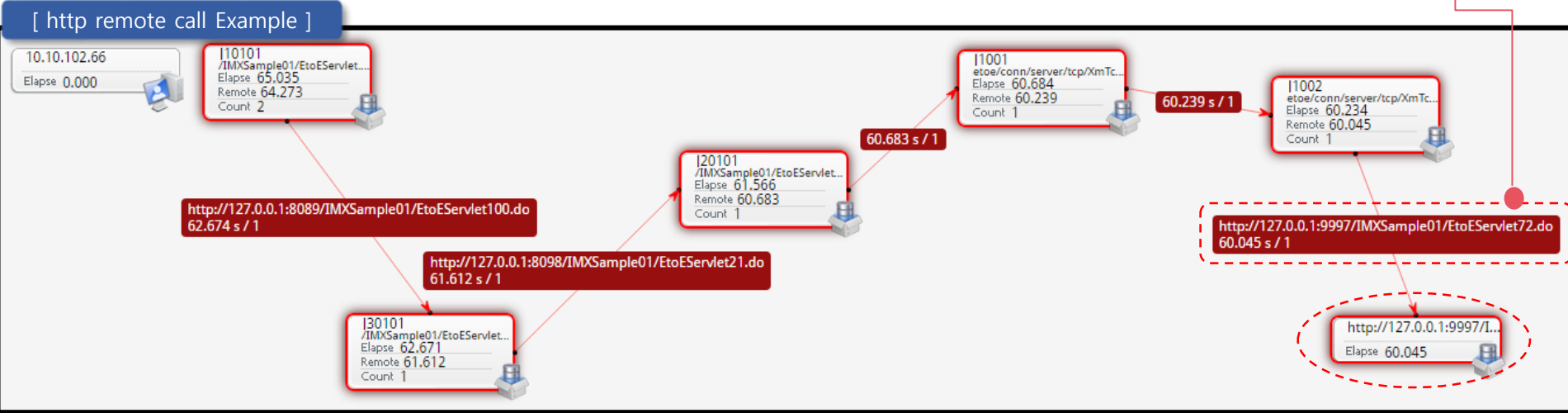


Provides IP/Port elapsed time, call count, call pathway information for section in which an agent is absent. (http / tcp / Sync / Async Call)

[ Displays Remote Call IP/Port based on Call Tree ]

Exceptions (Error) Count

클래스 Class	메소드 Method	예외(에러) 발생 건수	실행 건수 Execution Count	0.005	100.0%	io	유형
jeus.jspwork/_jsp/_600_tolower_5fjsp	_jspService(HttpServletRequest, H...	0	1	0.005	100.0%	io	
SOCKET_IS_OPEN	10.10.30.69:8888	0	1	0.000	0.0%	net	
TP	TOLOWER	0	1	0.004	80.0%	net	

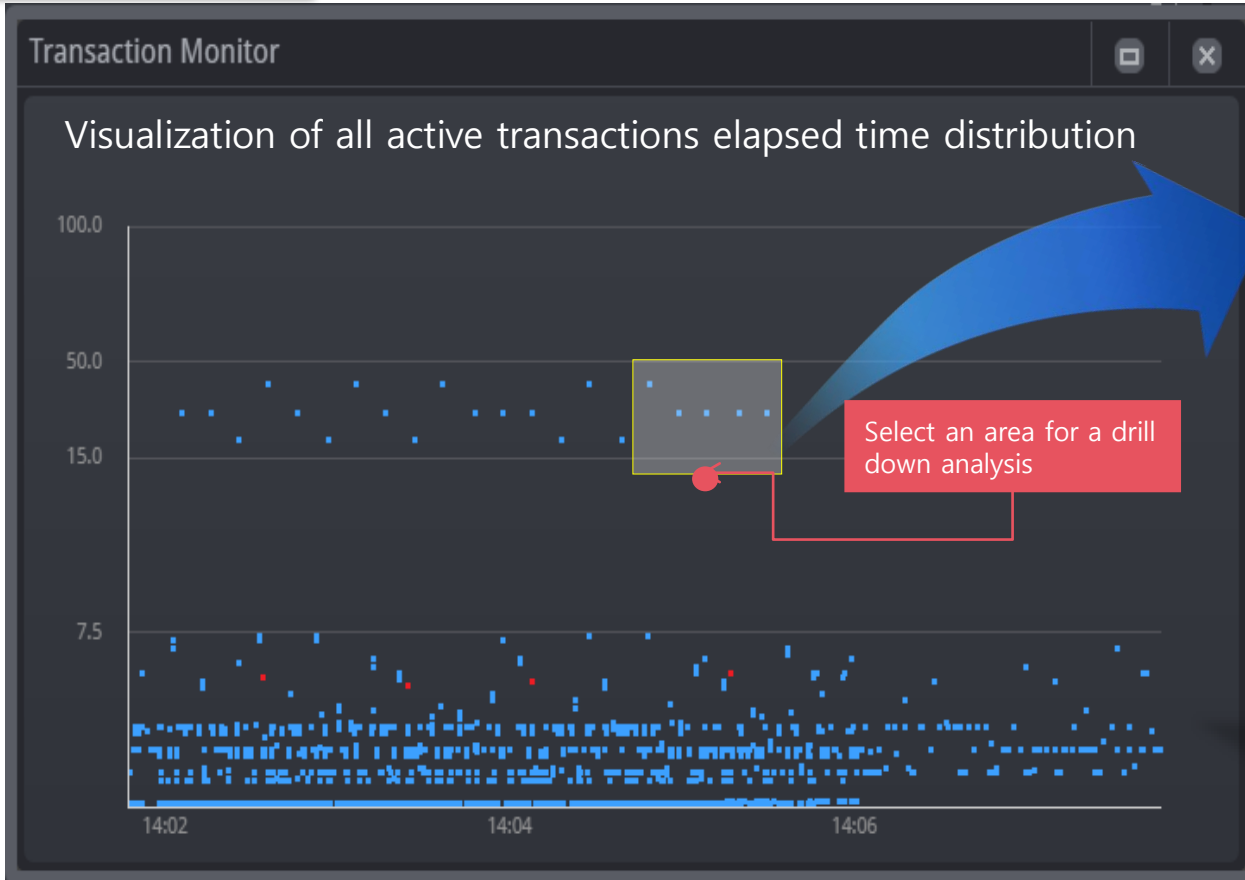


# MFJ E2E Main Features

(Feature) Transaction Trace – Represented by dots on Time series dot chart graph and a click/drag on the graph area opens a pop up window which displays the relevant monitoring information (1/3)

- Identify all of individual transaction's elapsed time distribution; exception generated transactions indicated in red.
- Check transaction details profiling information through the distribution chart.

## [ Transaction Monitor ]



Select a desired transaction and execute analysis on the transaction path view, call tree, and more.

# MFJ E2E Main Features

(Feature) Transaction Trace - Represented by dots on Time series dot chart graph and a click/drag on the graph area opens a pop up window which displays the relevant monitoring information (2/3)

- Through the "Transaction long term trend monitor" feature, you can **search the real-time monitoring section details**
- While real-time monitoring, instantaneous provision of transaction details information (call tree, SQL, path view, bind variable, error (exception), user text, and more).

The interface is divided into two main sections:

- [ Transaction long term trend monitoring ]**: A top-right window showing a time series dot chart. The y-axis represents transaction count (4-8) and the x-axis represents time (11:21-11:24). A red box highlights a specific data point, with a callout: "View transactions by various search conditions".
- [ Real-time transaction detail view ]**: A larger window below showing a detailed view of a transaction. It includes a table of transaction details and a call tree.

**Transaction Details Table:**

시간	에이전트	트랜잭션	시작 시간	수행 시간	예외(에러)	클라이언트 IP	SQL 수행 시간	SQL 실행 건수	SQL 패치 횟수	SQL 패치 시간
25 11:17:11	J10101	/IMXSample01/EtoEServlet.do	25 11:17:05	5.826		10.10.30.68	0.000	0	0	0.000
25 11:17:16	J20101	/IMXSample01/EtoEServlet90.do	25 11:17:11	5.689		10.10.30.97	0.000	0	0	0.000

**Call Tree:**

- 에이전트 J10101
- 클래스: javax/servlet/http/HttpServlet
- 메소드: service(HttpServletRequest, HttpServletResponse)
- 클래스: javax/servlet/http/HttpServlet
- 메소드: doPost(HttpServletRequest, HttpServletResponse)
- 클래스: etoe/XmEtoEController
- 메소드: execute()
- 클래스: etoe/conn/XmConnect
- 메소드: startNoThread()
- 클래스: etoe/conn/client/http/XmHttpClient
- 메소드: connect(String, String)
- 클래스: HTTP
- 메소드: call
- 클래스: SOCKET\_IS\_OPEN
- 메소드: SOCKET\_IS\_OPEN
- 클래스: SOCKET\_OS\_OPEN
- 메소드: SOCKET\_OS\_OPEN
- 클래스: SOCKET\_IS\_OPEN
- 메소드: SOCKET\_IS\_OPEN
- 클래스: SOCKET\_OS\_OPEN
- 메소드: SOCKET\_OS\_OPEN
- 클래스: etoe/txn/XmTxn
- 메소드: txnSwitch()
- 클래스: etoe/txn/model/SimpleMinSleep
- 메소드: execute()
- 클래스: util/XmSleep
- 메소드: random(int, int)

**Call Tree Table:**

클래스	메소드	예외(에러) 발생 건수	실행 건수	수행 시간	수행 시간 백분율	메소드 유형
javax/servlet/http/HttpServlet	service(HttpServletRequest, HttpServletResponse)	0	1	5.826	100.0%	
javax/servlet/http/HttpServlet	doPost(HttpServletRequest, HttpServletResponse)	0	1	5.826	100.0%	io
etoe/XmEtoEController	execute()	0	1	5.826	100.0%	loop,new alloc...
etoe/conn/XmConnect	startNoThread()	0	2	4.930	84.6%	
etoe/conn/client/http/XmHttpClient	connect(String, String)	0	2	4.930	84.6%	loop,io,strbuffer
HTTP	call	0	2	4.899	84.1%	net
SOCKET_IS_OPEN	SOCKET_IS_OPEN	0	1	0.000	0.0%	net
SOCKET_OS_OPEN	SOCKET_OS_OPEN	0	1	0.000	0.0%	net
SOCKET_IS_OPEN	SOCKET_IS_OPEN	0	1	0.000	0.0%	net
SOCKET_OS_OPEN	SOCKET_OS_OPEN	0	1	0.000	0.0%	net
etoe/txn/XmTxn	txnSwitch()	0	1	0.896	15.4%	
etoe/txn/model/SimpleMinSleep	execute()	0	1	0.896	15.4%	
util/XmSleep	random(int, int)	0	1	0.896	15.4%	

**Callout:** "Drag desired X-Y section for instant viewing" (pointing to a yellow box on the dot chart).

**(Feature) Transaction Trace - Represented by dots on Time series dot chart graph and a click/drag on the graph area opens a pop up window which displays the relevant monitoring information (3/3)**

- Analyze query monitoring executed in the transaction
- When bind variable is used in the corresponding query, and if it is possible to identify the query before and after binding, then SQL sorting feature is provided.
- In case client's sensitive information is to be included in the collection data, bind SQL encryption and other security features provided.

The screenshot displays the INTERMAX monitoring interface. At the top, there are navigation tabs for various transaction states. Below, a table shows the transaction trace for 'JEUS6\_con1'. The table has columns for '클래스' (Class), '메소드' (Method), '예외(여기) 발생 건' (Exception Occurrence), '실행 건수' (Execution Count), '수행 시간' (Execution Time), '수행 시간 백분율' (Execution Time Percentage), and '메소드 유형' (Method Type).

Two pop-up windows titled 'SQL 전문 보기' (SQL Text View) are overlaid on the screen. The first window shows a 'Bind Value List' with the following values:

```

1 /*-----*/
2 Bind Value List
3
4 1 = 'SMITH'
5 2 = 'CLERK'
6 3 = 7902
7 4 = 1000
8 5 = 2000
9 6 = 20
10 7 = 7369
11 -----*/
12
13 update emp set ename=:1, job=:2, mgr=:3,
14

```

A red callout box points to the list with the text 'Bind variables in query'. The second pop-up window shows the same SQL text with the bind variables replaced by their values:

```

1 UPDATE emp
2 SET
3   ename= 'SMITH',
4   job= 'CLERK',
5   mgr= 7902,
6   sal= 1000,
7   comm= 2000,
8   deptno= 20
9 WHERE empno = 7369

```

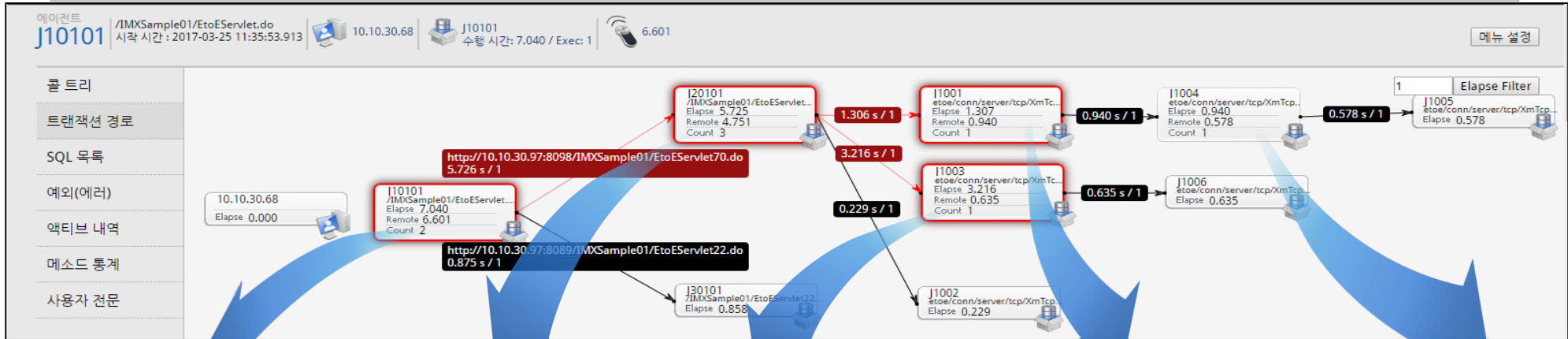
A red callout box points to the SQL text with the text 'Indicates bind mapped queries'. A blue arrow points from the bottom of the first pop-up window to the second one.

At the bottom of the main interface, there is a table for 'SQL 목록' (SQL List) with columns for '인스턴스' (Instance), 'SID', 'TIME', 'SQL 실행 건수' (SQL Execution Count), 'SQL 전체 수행 시간' (SQL Total Execution Time), 'SQL 최대 수행 시간' (SQL Max Execution Time), 'SQL 평균 수행 시간' (SQL Avg Execution Time), 'DB CPU 사용 시간' (DB CPU Usage Time), '대기 시간' (Wait Time), '논리 읽기' (Logical Reads), and '물리 읽기' (Physical Reads). The following table shows the data for the SQL list:

인스턴스	SID	TIME	SQL 실행 건수	SQL 전체 수행 시간	SQL 최대 수행 시간	SQL 평균 수행 시간	DB CPU 사용 시간	대기 시간	논리 읽기	물리 읽기	SQL Text
127.0.0.1.1521.ord	87	2017-03-09 18:18:30	1	598.346	598.346	598.346	0.003	598.343	4	0	update emp set ename=:1, job=:2, mgr=:3, sal=:4, comm=:5, deptno=:6 where empno=:7
	87	2017-03-09 18:08:31	1	0.001	0.001	0.001	0.001	0.000	0	0	update emp set ename=:1, job=:2, mgr=:3, sal=:4, comm=:5, deptno=:6 where empno=:7

## (Feature) Transaction Trace – View the entire call tree for completed transactions

- Displays the transaction movement information for all sections on the transaction path into which the transaction has flown.
- For sections in which the transactions are slow, go to call tree intuitively for performance tuning.
- MFJ *provides the total call tree information, not just for a particular method, by applying the super fast profiling technique. (Profiling is always provided during operation)*



J10101:IMXSample01/EtoEServlet.do:7.04 | J20101:IMXSample01/EtoEServlet.do:5.725 | J1003:etoe/conn/server/tcp/XmTcpServerClientHandler/run():3.216 | J1001:etoe/conn/server/tcp/XmTcpServerClientHandler/run():1.307 | J1004:etoe/conn/server/...

텍스트 모드  수행 시간이 0%인 항목 숨기기

클래스	메소드	예외(에러) 발생 건수	실행 건수	수행 시간	수행 시간 백분율	메소드 유형
javax/servlet/http/HttpServlet	service(HttpServletRequest, HttpS...	0	1	7.040	100.0%	
servlet/XmEtoEServlet	doPost(HttpServletRequest, HttpS...	0	1	7.040	100.0%	io
etoe/XmEtoEController	execute()	0	1	7.040	100.0%	loop,new alloc...
etoe/conn/XmConnect	startNoThread()	0	2	6.617	94.0%	
etoe/conn/client/http/XmHttpClient	connect(String, String)	0	2	6.617	94.0%	loop,io,stribuffer
FILE_IS_OPEN	C:\Program Files\Java\jdk1.7.0_75...	0	2	0.000	0.0%	io
HTTP	call	0	2	6.601	93.8%	net
SOCKET_IS_OPEN	10.10.30.97:8098	0	1	0.000	0.0%	net
SOCKET_OS_OPEN	10.10.30.97:8098	0	1	0.000	0.0%	net
SOCKET_IS_OPEN	10.10.30.97:8089	0	1	0.000	0.0%	net
SOCKET_OS_OPEN	10.10.30.97:8089	0	1	0.000	0.0%	net
etoe/txn/XmTxn	txnSwitch()	0	1	0.423	6.0%	
etoe/txn/model/SimpleMinSleep	execute()	0	1	0.423	6.0%	
util/XmSleep	random(int, int)	0	1	0.423	6.0%	

By applying the Super Fast Profiling Technique, MFJ provides a constant profiling analytic data of the entire methods during operation, not just a particular method's performance data.



(Feature) Operation Management – We provide detailed information of SQL executed in the client company’s applications. (Items: query, fetch count, SQL parameter, session info, session ID)

- Use the SQL detail viewing feature to check the details of SQLs executed in all transactions.

[ Transaction Details Feature ]

The screenshot displays the INTERMAX application interface. At the top, there are navigation tabs for various transaction states. Below, a tree view shows the transaction details for 'JEUS6\_con1'. A table lists various classes and methods with their execution counts and times. A red box highlights a specific SQL query in the 'SQL 전문 보기' (SQL Detail View) window. The query is an update statement with bind variables. A red callout box points to the bind variables in the query, stating 'Bind variables in query'. Below the main screenshot, a table shows the SQL execution details for the highlighted query, including session ID, time, and SQL text.

SQL 목록	인스턴스	SID	TIME	SQL 실행 건 수	SQL 전체 수행 시간	SQL 최대 수행 시간	SQL 평균 수행 시간	DB CPU 사용 시간	대기 시간	논리 읽기	물리 읽기	SQL Text
예외(에러)	127.0.0.1.1521.orcl	87	2017-03-09 18:18:30	1	598.346	598.346	598.346	0.003	598.343	4	0	update emp set ename=:1, job=:2, mgr=:3, sal=:4, comm=:5, deptno=:6 where emp...
메소드 통계		87	2017-03-09 18:08:31	1	0.001	0.001	0.001	0.001	0.000	0	0	SELECT 1 FROM DUAL

Provides individual SQL query, fetch count, SQL parameter, session ID information.

# MFJ E2E Main Features

**(Feature) Operation Management – Displays in Korean text on dashboard and screen. (Item: Individual transaction name)**

- Task transaction name can be mapped 1:N and named in Korean.
- Renamed transaction is displayed with a user friendly Korean transaction name.
- Korean naming is supported for all other items including task name, agent name, node group name and more.

Environment Settings > Task Settings > Task Transaction Name Settings

[View Transaction]

업무 트랜잭션명

ID

업무명 E2E모니터링

JSP 목록

JSP 찾기

JSP명	업무명
<input type="checkbox"/> 158 /smtest/eum2/simple11.jsp	
<input type="checkbox"/> 159 /smtest/eum2/simple1-1.jsp	
<input type="checkbox"/> 160 /smtest/eum2/simple0-1.jsp	
<input type="checkbox"/> 161 /smtest/jsp/toupper.jsp	
<input type="checkbox"/> 162 TOUPPER	
<input type="checkbox"/> 163 FDSELE	
<input type="checkbox"/> 164 /IMXSample02/jsp/index.jsp	
<input type="checkbox"/> 165 /IMXSample02/EtoEServlet.do	
<input checked="" type="checkbox"/> 166 /IMXSample01/EtoEServlet22.do	
<input type="checkbox"/> 167 /IMXSample01/EtoEServlet67.do	
<input type="checkbox"/> 168 /IMXSample01/EtoEServlet60.do	
<input type="checkbox"/> 169 /IMXSample01/EtoEServlet28.do	
<input type="checkbox"/> 170 /IMXSample01/EtoEServlet88.do	

[Environment Settings > Task Settings > Task Transaction Name Settings]

[View Transaction]

트랜잭션	시작 시간	수행 시간
[샘플화면02] /smtest/jsp/ForwardTest_Body.jsp	25.11.27.15.320	2.002
[샘플화면02] /smtest/jsp/ForwardTest_Body.jsp		
[샘플화면02] /smtest/jsp/ForwardTest_Body.jsp		
[샘플화면01] /smtest/jsp/jwtest3.jsp		
[샘플화면01] /smtest/jsp/jwtest3.jsp		
[샘플화면01] /smtest/jsp/jwtest2.jsp		
[샘플화면01] /smtest/jsp/jwtest2.jsp		
[샘플화면01] /smtest/jsp/jwtest2.jsp		
[샘플화면01] /smtest/jsp/jwtest.jsp		
[샘플화면01] /smtest/jsp/jwtest.jsp		
[샘플화면01] /smtest/jsp/jwtest.jsp		
[샘플화면01] /smtest/jsp/jwtest.jsp		
[E2E모니터링] /IMXSample01/EtoEServlet22.do		

주요 거래목록

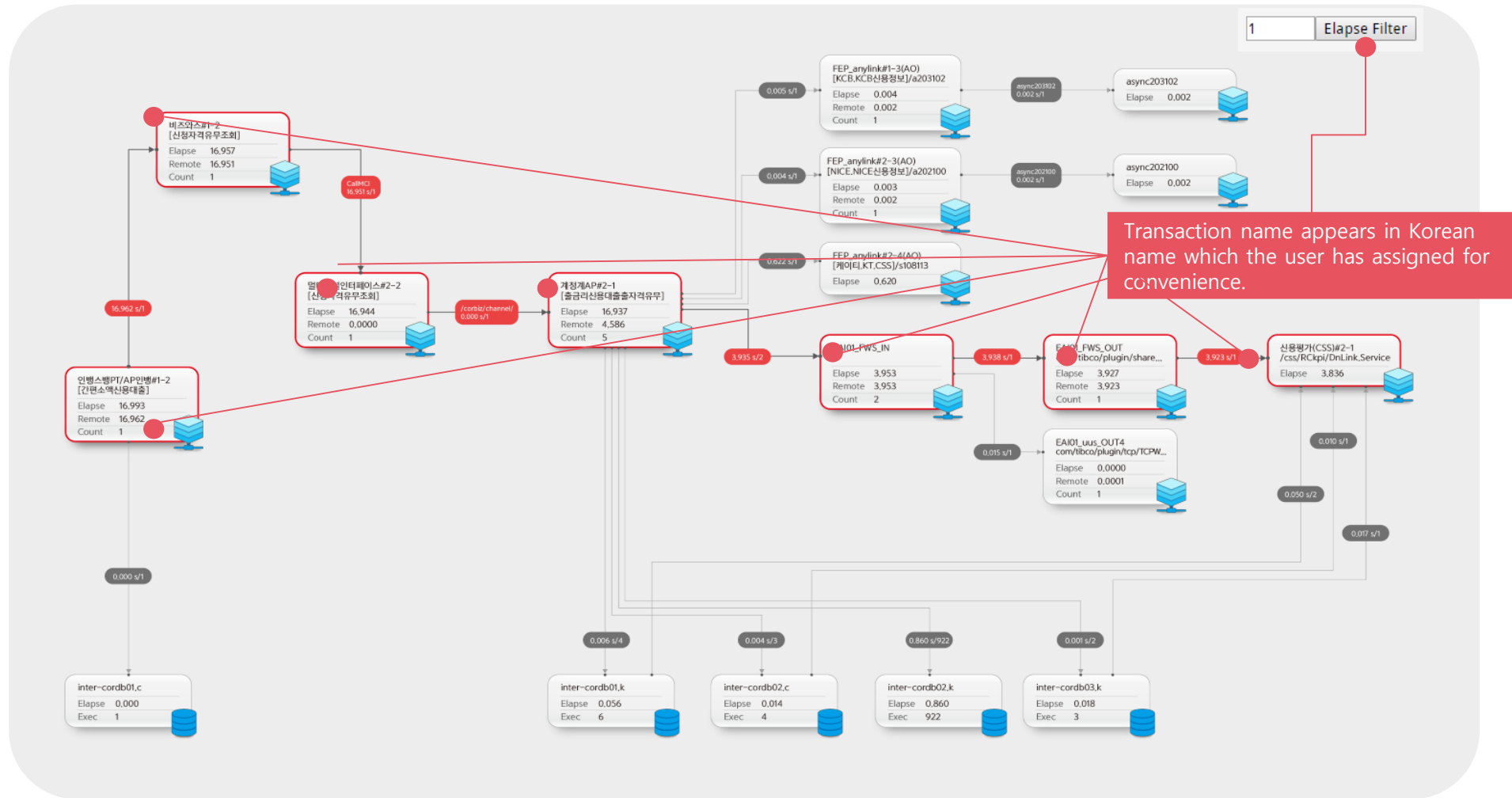
거래이름	최근	일차	충당도
[대인] /b20/mnu/CERCSD6040000102A	10초	5초	50% ↓
[이체하기] /b20/mnu/PBKTRND1000000301V	8초	7초	10% ↓
[이체결과조회] /b20/mnu/PBKTRND070300	7초	9초	20% ↓
[조회편가입] /b20/mng/CMMBJD300000801V	9초	9초	1% ↓
[이벤트] /b20/mnu/CMMN07020100	9초	4초	50% ↓
[영업보고서] /b20/mnu/HOMBA020304	4초	2초	50% ↓
[잔액조회] /b20/mng/pbkng010000000V	2초	5초	10% ↓
[카드발급/계송조회] /b20/mnu/ymod0203100	5초	7초	20% ↓
[자영업이체신청조회/변경/변지] /b20/mnu/PBKTRND08200	10초	9초	1% ↓
[직장인신용대출] /b20/acd/FPMLDND500000010A	8초	9초	50% ↓
[이체하기] /b20/mnu/PBKTRND1000000301V	7초	5초	10% ↓
[이체결과조회] /b20/mnu/PBKTRND070300	9초	7초	20% ↓
[조회편가입] /b20/mng/CMMBJD300000801V	9초	9초	1% ↓
[이벤트] /b20/mnu/CMMN07020100	4초	9초	50% ↓
[영업보고서] /b20/mnu/HOMBA020304	2초	4초	10% ↓
[잔액조회] /b20/mng/pbkng010000000V	5초	2초	20% ↓
[카드발급/계송조회] /b20/mnu/ymod0203100	7초	5초	1% ↓
[자영업이체신청조회/변경/변지] /b20/mnu/PBKTRND08200	9초	7초	50% ↓
[이체하기] /b20/mnu/PBKTRND1000000301V	9초	9초	1% ↓

The transaction name can be renamed into a user friendly name in Korean.

# MFJ E2E Main Features

(Feature) Operation Management – Transaction delays and root cause analysis screen are intuitively perceptible and concise performance data is provided. (2/6)

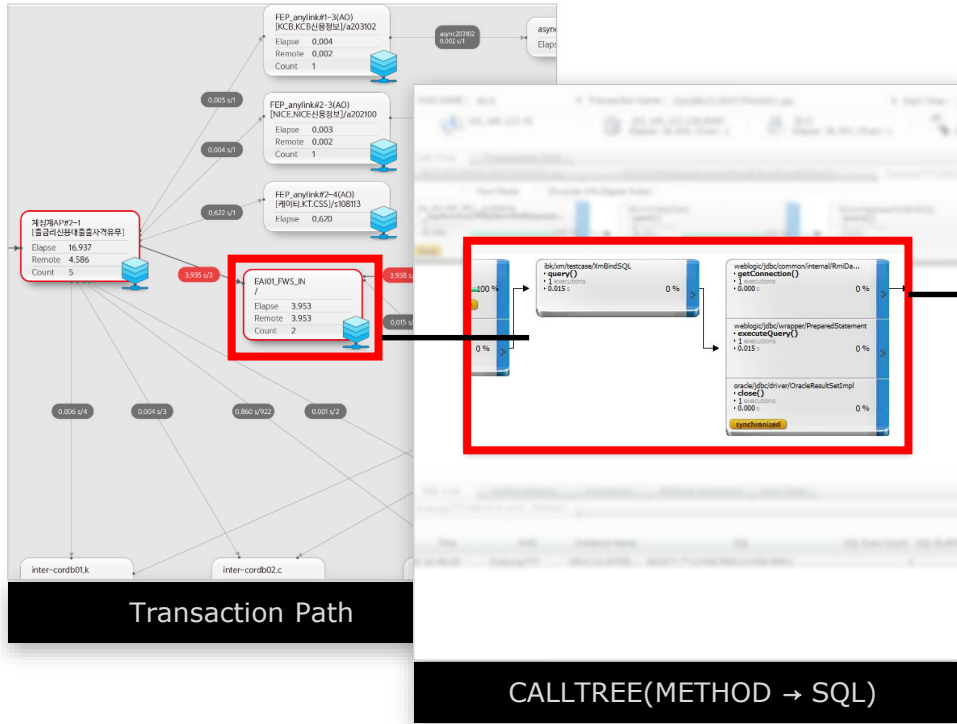
- According to execution time filter feature, you can intuitively identify the delay sections.



# MFJ E2E Main Features

(Feature) Operation Management – Transaction delays and root cause analysis screen are intuitively perceptible and concise performance data is provided. (3/6)

- Execute Root Cause Analysis with the least number of clicks.
- Easily analyze the root cause of problem and identify performance issues through intuitive UI.



**CLASS SOURCE**

**SOURCE Change Trace**

The 'CLASS SOURCE' view displays the source code of the class 'Gyejeong' (Path: jsp\_servlet/\_bk/\_gyejeong\_jspService). The code includes imports for various packages like 'ibk.xm.testcase', 'javax.servlet.jsp', and 'java.io'. The 'SOURCE Change Trace' view shows a comparison of source code between two different versions or environments, with changes highlighted in orange and blue.

Provides various features for Trouble Shooting

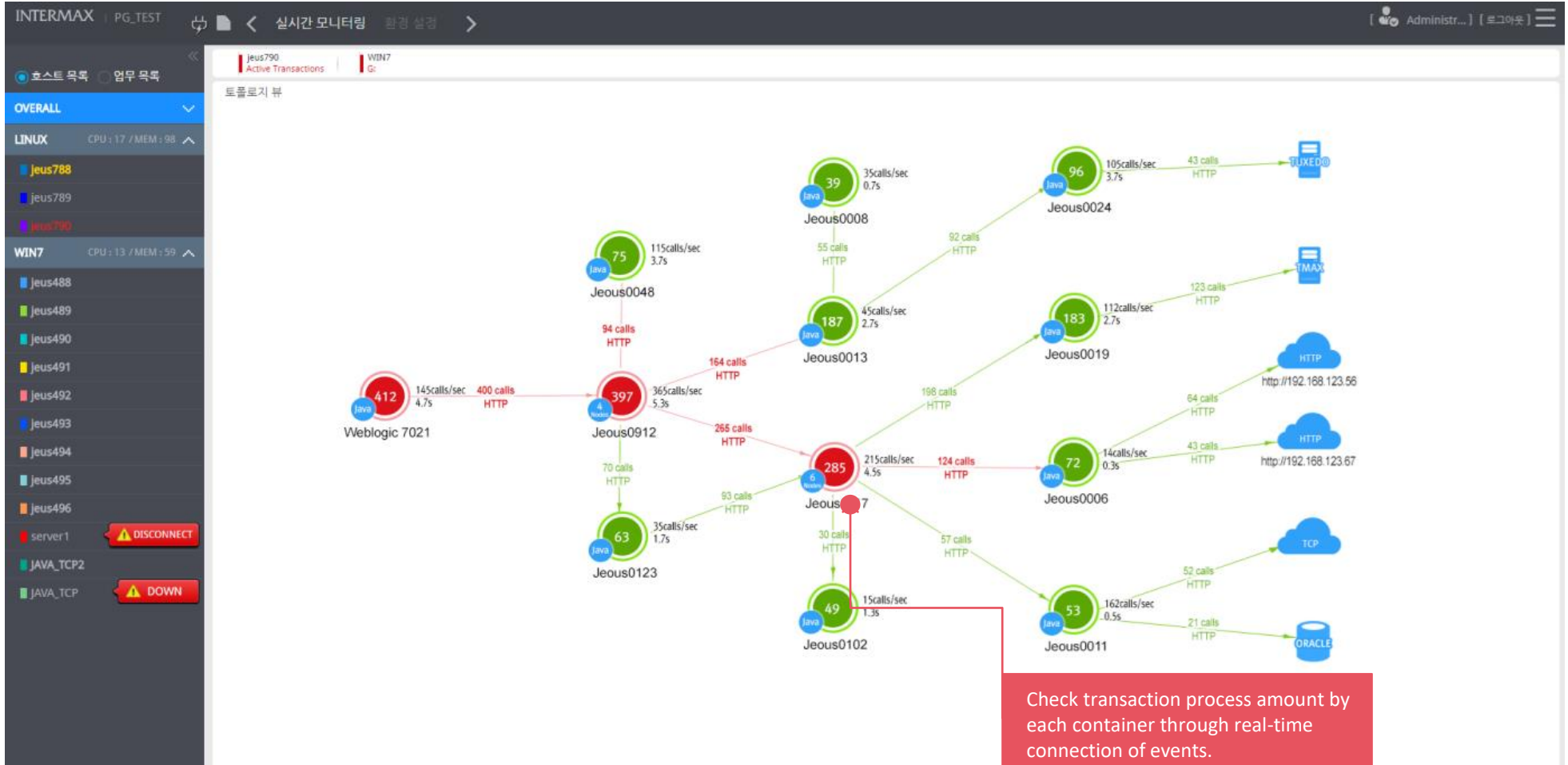
- Call Tree Analysis
- Exceptions

- DB Lock status, workload information
- Source comparison, environment setting files comparison
- Elapsed by section
- GC, memory status
- Class and JSP source view

# MFJ E2E Main Features

(Feature) Operation Management – Transaction delays and root cause analysis screen are intuitively perceptible and concise performance data is provided (4/6)

- Identify the work flow and system configuration's transaction flow at a glance through the real-time topology view.
- Monitor the service status through real-time connection of events.

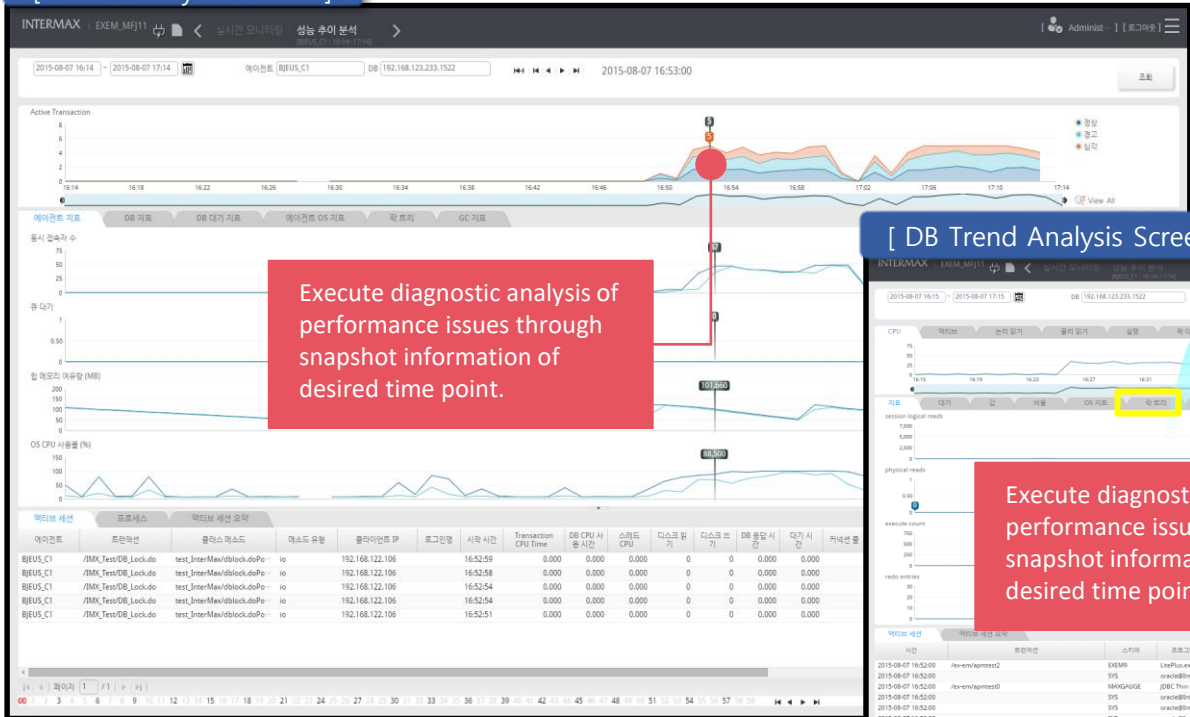


# MFJ E2E Main Features

**(Feature) Operation Management – Transaction delays and root cause analysis screen are intuitively perceptible and concise performance data is provided (5/6)**

- Transaction related data are provided in chart and snapshot data format so you can analyze a specific time point's execution status.
- Past history is provided in 3 second duration snapshots and can be used to analyze the root cause of issues just as in real-time.

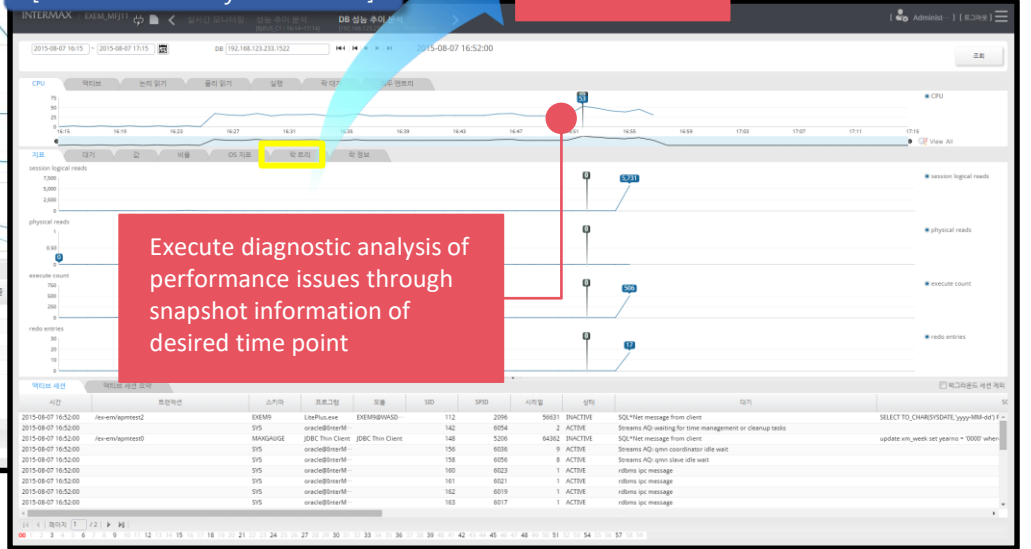
[ Trend Analysis Screen ]



- Workload information (transaction count, elapsed time, etc)
- JVM / TP / OS / DB Stat
- Resource information
- Process and active session information



[ DB Trend Analysis Screen ]



- DB workload information (wait time, logical read, physical read, wait event)
- DB Lock status, workload information
- Active DB session information (Transaction connection)

# MFJ E2E Main Features

(Feature) Operation Management – Transaction delays and root cause analysis screen are intuitively perceptible and concise performance data is provided (6/6)

- Provides a feature which allows you to set thresholds for transaction delays and in case of an event generation, to connect with the related details analysis screen.

[ Real-time Event Alarms ]

In case of a real-time event (alarm) generation, click on the corresponding event to connect to the detail analysis screen.

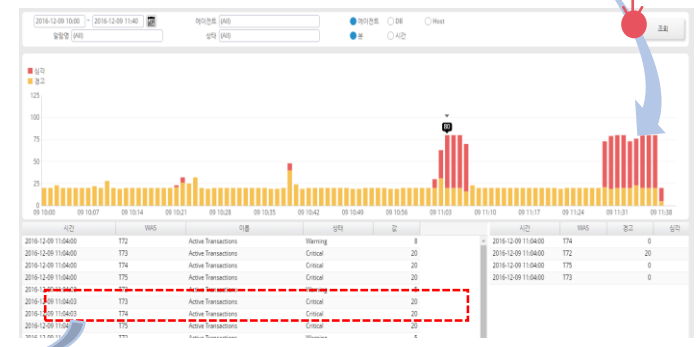
알람 발생 내역

시간	업무명	호스트명	에이전트	이벤트명	상태	값	설명
11:30:48	LINUX	T72	Active Transactions	Warning	5		
11:30:48	WIN97	J10101	Elapsed Time	Critical	151,748	/IMXSample01/EtoESer...	
11:30:48	WIN97	J10101	Elapsed Time	Critical	147,59	/IMXSample01/EtoESer...	
11:30:48	WIN97	J10101	Elapsed Time	Critical	110,01	/IMXSample01/EtoESer...	
11:30:45	WIN97	J10101	Elapsed Time	Critical	14,733		

J10101

[ Topology View – Integrated Events (Alarm) ]

Topology View – In case of an event (alarm) generation, connect to the detailed information analysis screen with one-click.



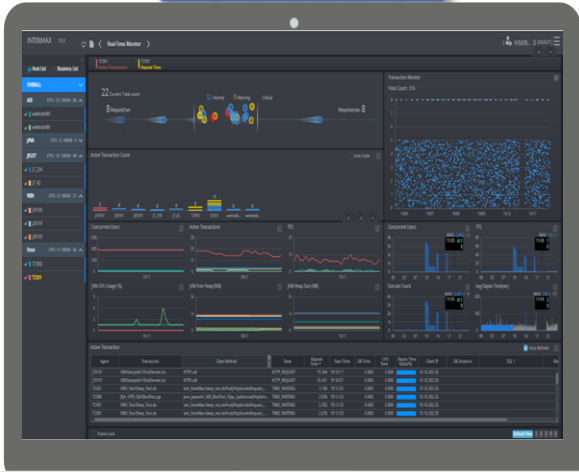
Connect with real-time call tree to analyze the elapsed time event.

# MFJ E2E Main Features

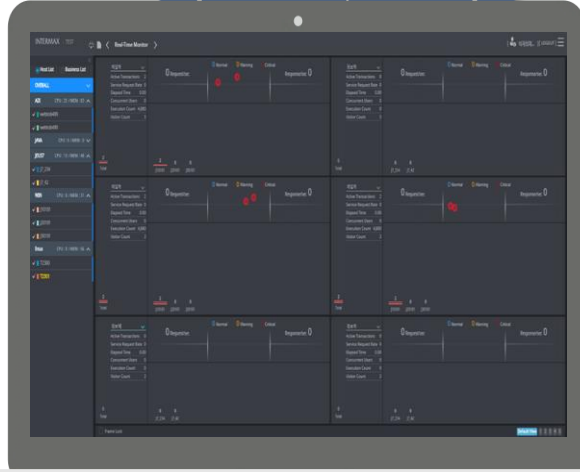
**(Feature) Operation Management – Provides various types of dashboards to meet the needs of the user and can be customized for user convenience. (1/4)**

- Provides various types of embedded dashboards
- Provides Home, Work groups, Administrator, WAS-DB integrated monitoring, topology view, and other various perspectives and types of dashboard screens to suit the user's work patterns.

[ Home screen ]



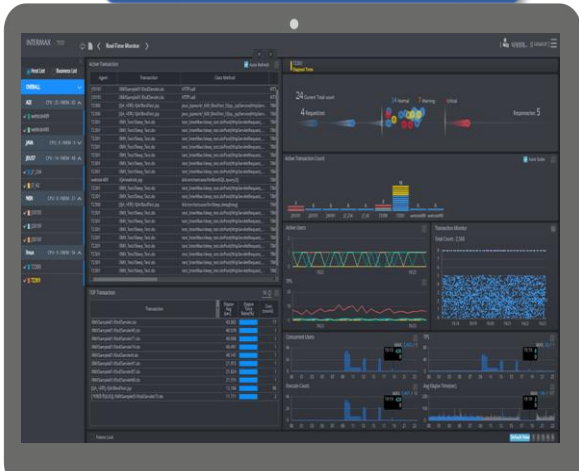
[ Work Groups Screen ]



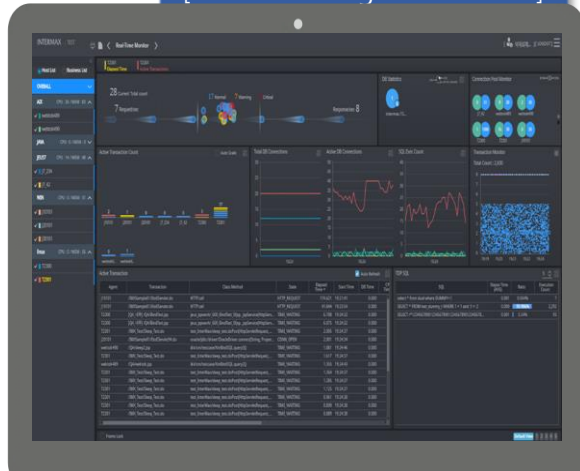
[ Topology View Screen ]



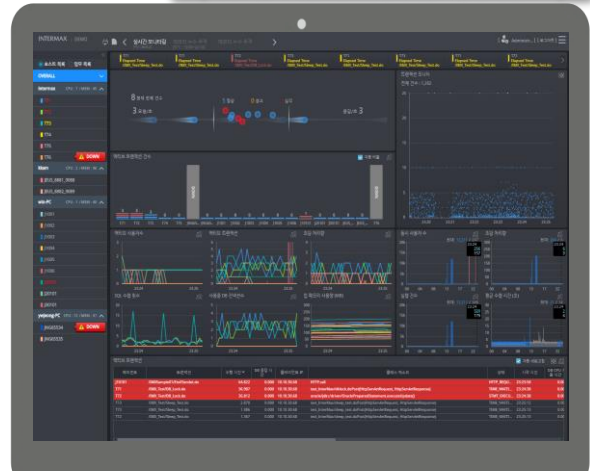
[ Administrator Screen ]



[ WAS-DB Integrated Screen ]



[ WAS Administrator Screen ]



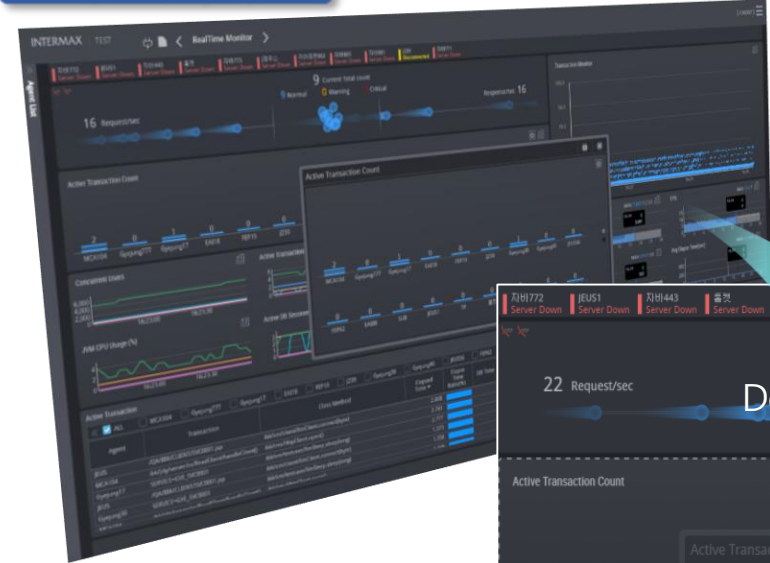


# MFJ E2E Main Features

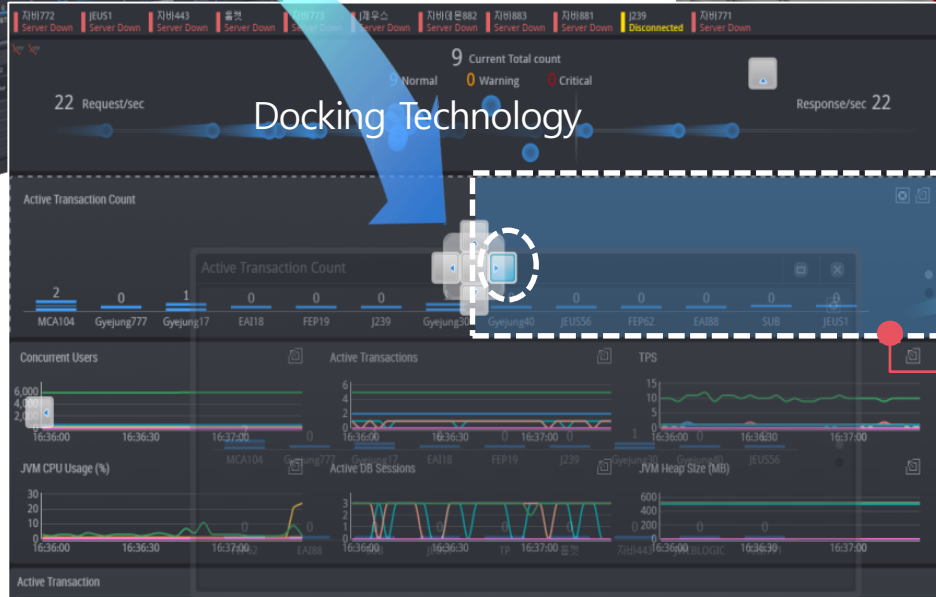
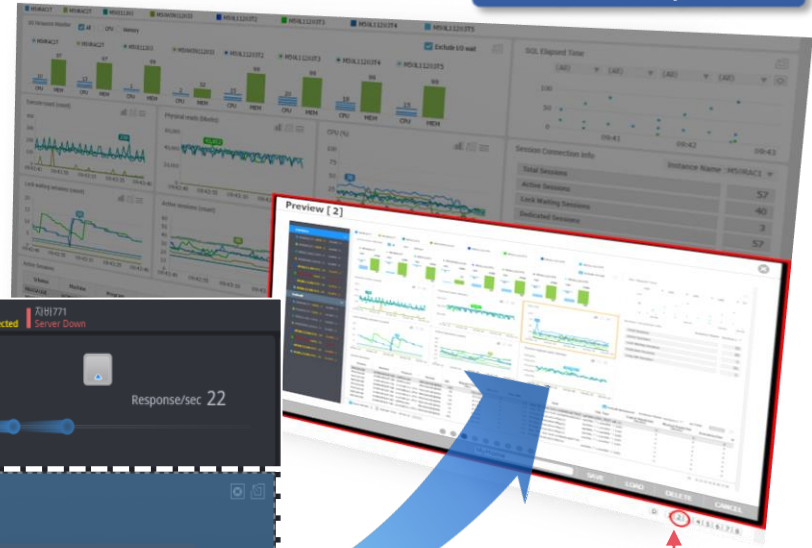
**(Feature) Operation Management – Provides various types of dashboards to meet the needs of the user and can be customized for user convenience. (2/4)**

- You can freely configure the real-time monitoring screen by using the docking feature to arrange the desired screens.
- Docking configuration of various types of real-time window frames

[ Black Style ]



[ White Style ]



Predefined View & Save Layout

Dock any additional desired screens for monitoring.

**(Feature) Operation Management – Provides various types of dashboards to meet the needs of the user and can be customized for user convenience (3/4)**

- Identify the overall system's problems at a glance through alarms by setting thresholds for the monitoring items.
- Regarding the data collected through APM solutions, customized dashboards can be provided to meet the needs of the client company.

The screenshot shows the INTERMAX monitoring interface with a 'Select Menu' for real-time monitoring. The menu is divided into two columns:

실시간 대시보드	실시간 모니터링
실시간 모니터링 (기본)	실시간 도킹 프레임
실시간 모니터링 (변형)	Activity 모니터
실시간 모니터링 통합 뷰	Activity 그룹 모니터
그룹 모니터링 뷰	액티브 트랜잭션 건수
매니저 뷰	그룹별 액티브 트랜잭션 건수
멀티인스턴스 로드 밸런싱	액티브 트랜잭션
WAS-DB 연계뷰	트랜잭션 모니터
시스템 리소스	최근 서버별 성능 지표
메모리	최근 성능 지표 (합계)
사용자	균일 서비스 지표
트롤로지 뷰	균일 방문자 수
	평균 시간당 수행 건수
	앱별 정보
	알림 발생 내역
	GC 지표
	CPU 사용량
	커넥션 풀 모니터

Real-time integrated dashboard is provided separately so that each user can freely choose the performance stat depending on his or her perspective for configuration and monitoring.

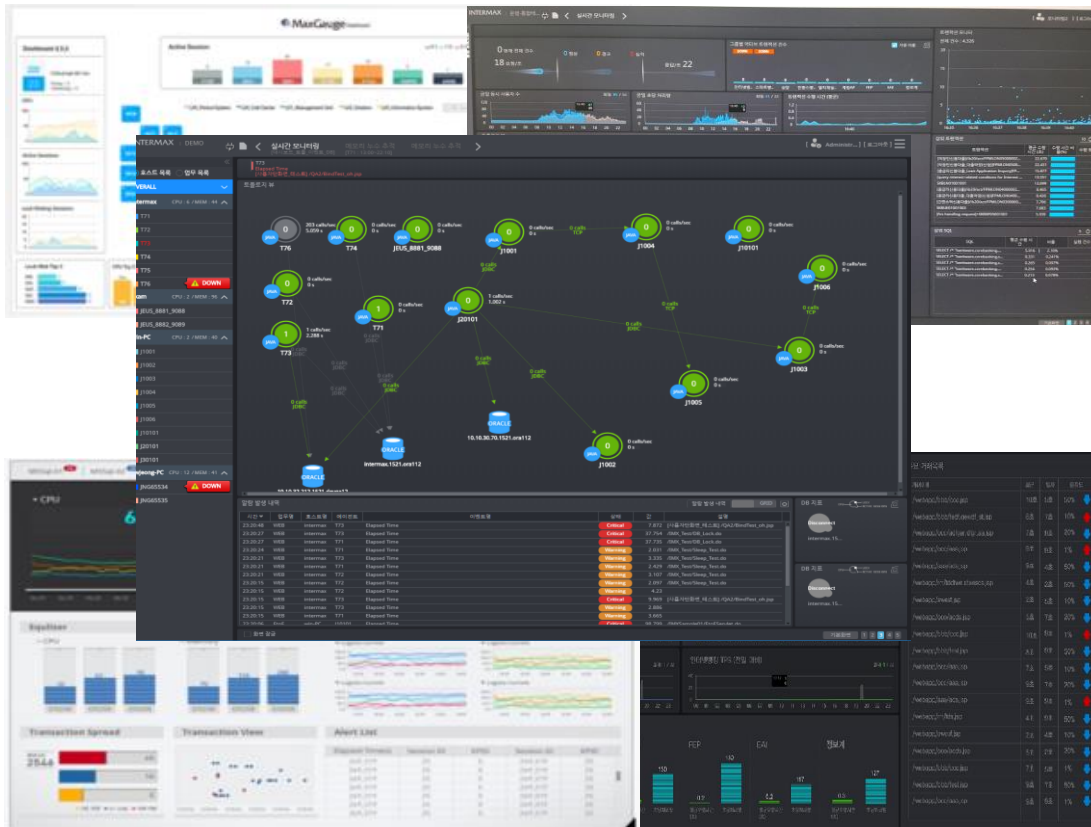


# MFJ E2E Main Features

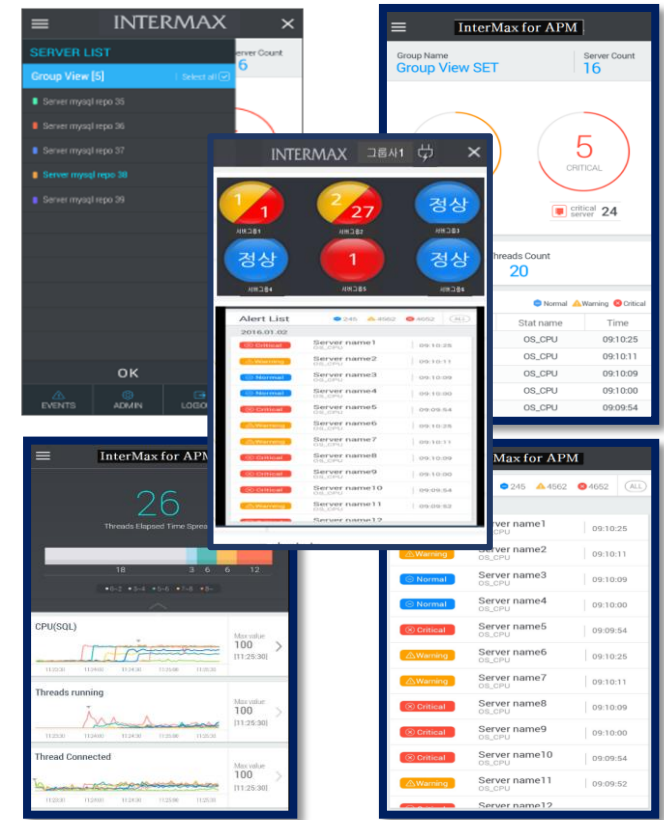
(Feature) Operation Management – Provides various types of dashboards to meet the needs of the user and can be customized for user convenience (4/4)

- Besides the default dashboards, you can also use the mobile version dashboards for urgent response to events. (Expected to release in 2017)

[ PC version Dashboards ] : Provided by default in the solutions.



[ Mobile version Dashboards ] : Mobile dashboards expected to release in 3Q of 2017



## (Feature) Operation Management – Provides various types of reports. (By transactions per user request, time, system, etc. reporting feature) (1/2)

- Individual-based, user-defined dashboard and reports.
- When creating a statistics report, user can select desired target WAS and stats.

[ Create Analysis Report ]

보고서 양식 > 주간 에이전트 보고서

● 지난 주 2017-02-27 ~ 2017-03-05 Time Window 09 ~ 18 30

에이전트 목록

에이전트 찾기

- Intermax
  - T71
  - T72
  - T73
  - T74
  - T75
- win-PC
  - J1001
  - J1002
  - J1003
  - J1004
  - J1005
  - J1006
  - J10101
  - J20101
  - J30101
- ktam
  - JEU5\_8881\_9088
  - JEU5\_8882\_9089
- ywjeong-PC
  - JNG65534
  - JNG65535

지표 목록

지표 찾기

- 지표명
- 실행 건수
- 평균 수할 시간
- 초당 입력 처리 건수
- 초당 처리량
- 악티브 트랜잭션
- 동시 사용자 수
- OS CPU 사용률 (%)
- JVM CPU 사용률 (%)
- OS 메모리 사용량 (MB)
- 힙 메모리 사용량 (MB)
- 힙 메모리 크기 (MB)
- 사용자 DB 연결 건수
- 전체 DB 연결 건수
- 대기 DB 연결 건수

선택된 지표

- 초당 입력 처리량
- OS CPU 사용률 (%)
- OS 메모리 사용량 (MB)
- 힙 메모리 크기 (MB)
- 힙 메모리 사용량 (MB)

이전 | 엑셀로 저장

[ Daily Report ]

기간: 2017-02-27 ~ 2017-03-05

총합: 실행 건수 19300, 평균 수할 시간 09:00 - 17:30, 에이전트 85335

1. 에이전트 성능 요약

에이전트	실행 건수	평균 수할 시간	평균 처리량	OS CPU 사용률 (%)	OS 메모리 사용량 (MB)	힙 메모리 사용량 (MB)	힙 메모리 크기 (MB)
Intermax	1275	0.00	0.00	0.00	0.00	0.00	0.00
win-PC	16525	0.00	0.00	0.00	0.00	0.00	0.00
ktam	0	0.00	0.00	0.00	0.00	0.00	0.00
ywjeong-PC	0	0.00	0.00	0.00	0.00	0.00	0.00

2. 에이전트 별 상세 성능

Report

[ Weekly Report ]

주간 시스템 성능

기간: 2017-02-27 ~ 2017-03-05

총합: 실행 건수 19300, 평균 수할 시간 09:00 - 17:30, 에이전트 85335

1. 에이전트 성능 요약

2. 에이전트 별 상세 성능

시간당 실행 건수, TPS, TPS(MIN), TPS(MAX), 수행시간(AVG), 수행시간(MAX)

시간	실행 건수	평균 수할 시간	평균 처리량	OS CPU 사용률 (%)	OS 메모리 사용량 (MB)	힙 메모리 사용량 (MB)	힙 메모리 크기 (MB)
0:00	0	0.00	0.00	0.00	0.00	0.00	0.00
1:00	1	0.00	1.00	12.34	0.00	0.00	0.00
2:00	2	0.00	2.00	13.11	0.00	0.00	0.00
3:00	3	0.00	3.00	12.34	0.00	0.00	0.00
4:00	4	0.00	4.00	11.11	0.00	0.00	0.00
5:00	5	0.00	5.00	0.00	0.00	0.00	0.00

[ Monthly Report ]

월별 시스템 성능

기간: 2016-10 ~ 2016-11

총합: 실행 건수 19300, 평균 수할 시간 09:00 - 17:30, 에이전트 85335

1. 에이전트 성능 요약

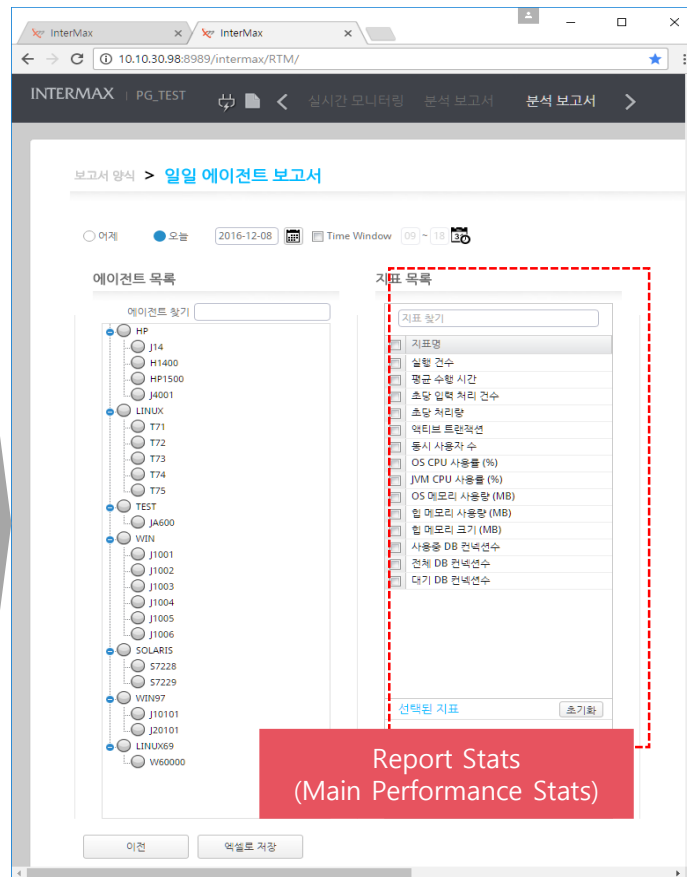
2. 에이전트 별 상세 성능

시간당 실행 건수, TPS, TPS(MIN), TPS(MAX), 수행시간(AVG), 수행시간(MAX)

User can select desired target WAS and performance stats to create a report.

**(Feature) Operation Management – Provides various types of reports. (By transactions per user request, time, system, etc. reporting feature) (2/2)**

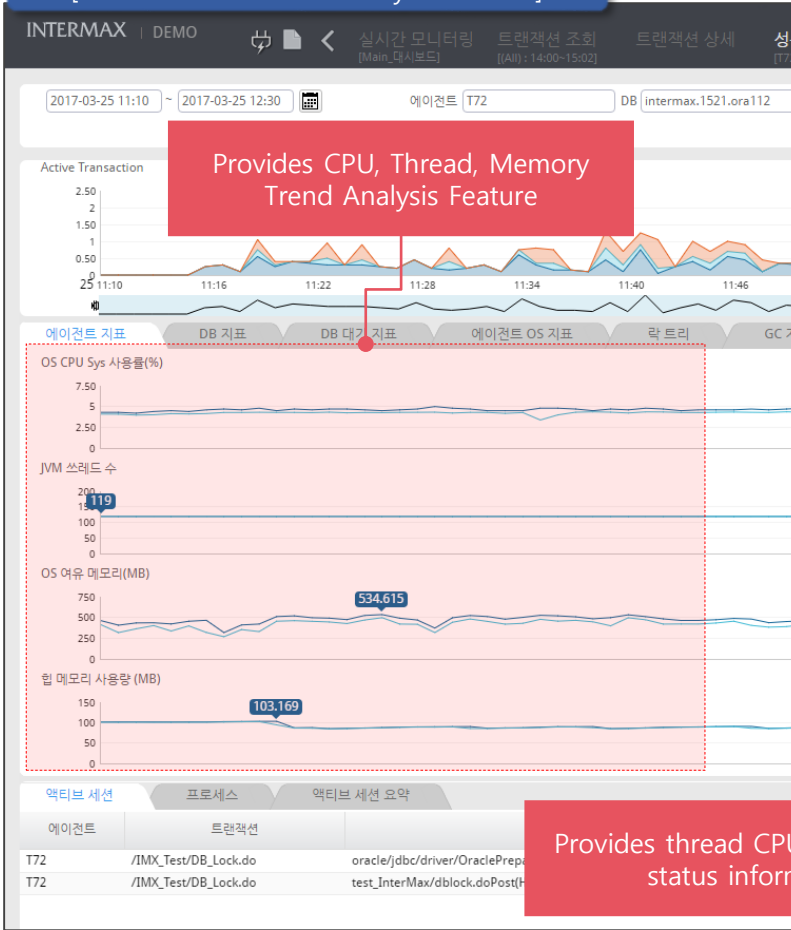
- Provides daily/ weekly/ monthly reporting on main performance stats of the entire system.
- Reporting can be customized based on the kinds of statistics data of the client company.



## (Feature) Operation Management – CPU, Thread, Memory, Coredump Analysis Features

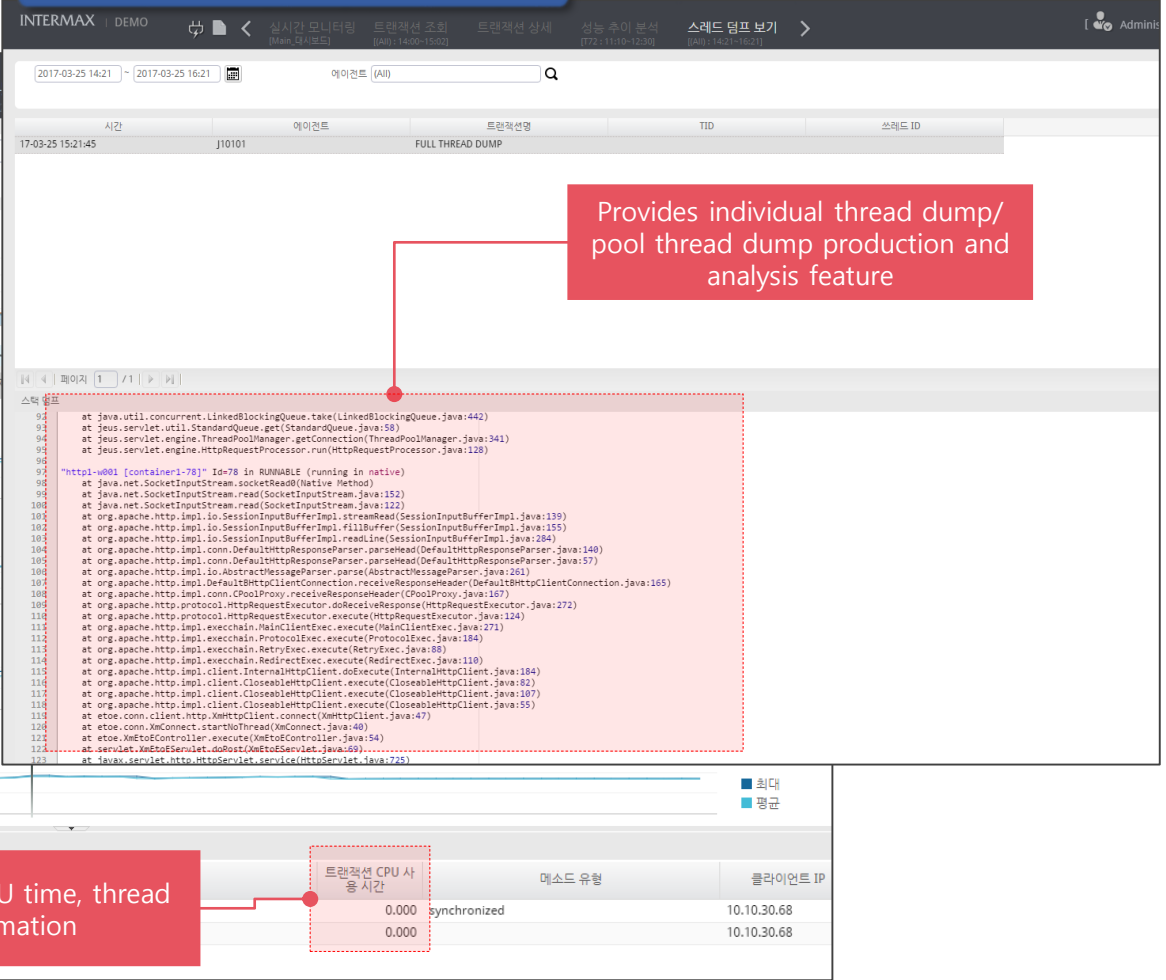
- Provides OS CPU Analysis, Thread CPU Time, Thread Count, OS Memory/Heap Memory usage trend, Thread Dump analysis features.

### [ Performance Trends Analysis Feature ]



Provides CPU, Thread, Memory Trend Analysis Feature

### [ Thread Dump Analysis Feature ]



Provides individual thread dump/pool thread dump production and analysis feature

Provides thread CPU time, thread status information

Thank you for working with



Exem Co., LTD is Korea's leading performance management specialist group.

Thank you