DBPLUS
better performance

DBPLUS
Performance Monitor for Oracle
description of changes in version 2021.3

Date: October 8, 2021

Table of Contents

Below is a list of changes in the DBPLUS Performance Monitor system for monitoring Oracle databases.

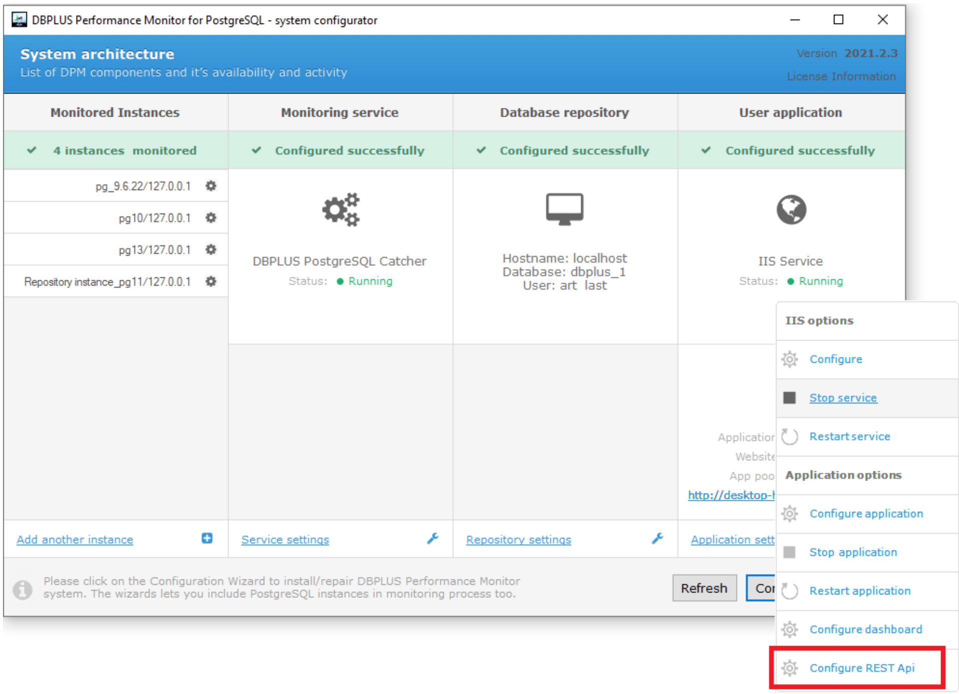
New In version 2021.3

1 REST API – Performance Monitor

The REST API module is available in the DBPLUS Performance Monitor application from version 2021.3.1 for each of the monitored database platforms.

1.1. Configuration

To run REST API functions for a given platform, go to the **Application settings** menu from the configurator and then select **Configure REST Api**.



DBPLUS Performance Monitor for PostgreSQL - system configurator

System architecture
List of DPM components and it's availability and activity

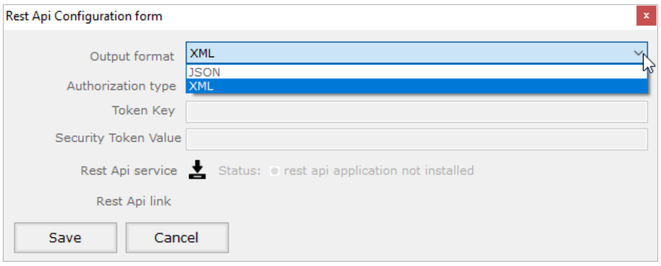
Version 2021.2.3
License Information

Monitored Instances	Monitoring service	Database repository	User application
✓ 4 instances monitored	✓ Configured successfully	✓ Configured successfully	✓ Configured successfully
pg_9.6.22/127.0.0.1 pg10/127.0.0.1 pg13/127.0.0.1 Repository instance_pg11/127.0.0.1	DBPLUS PostgreSQL Catcher Status: ● Running	Hostname: localhost Database: dbplus_1 User: art last	IIS Service Status: ● Running

Please click on the Configuration Wizard to install/repair DBPLUS Performance Monitor system. The wizards lets you include PostgreSQL instances in monitoring process too.

As part of the configuration, the User can set the format - **Output format** in which the information will be returned using the API. Options to choose from:

- JSON
- XML



Rest Api Configuration form

Output format: XML

Authorization type

Token Key

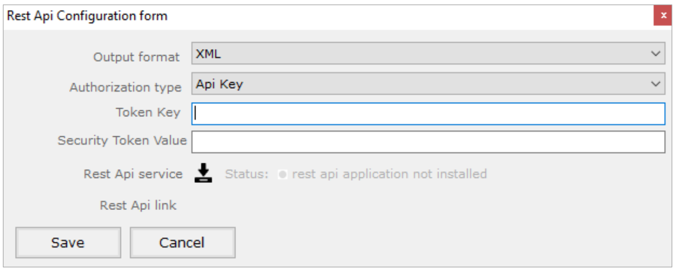
Security Token Value

Rest Api service Status: ○ rest api application not installed

Rest Api link

Then the User can configure the authorization type. Options:

- **Not Set** – no authorization
- **API Key** – key authorization



Rest Api Configuration form

Output format: XML

Authorization type: Api Key

Token Key

Security Token Value

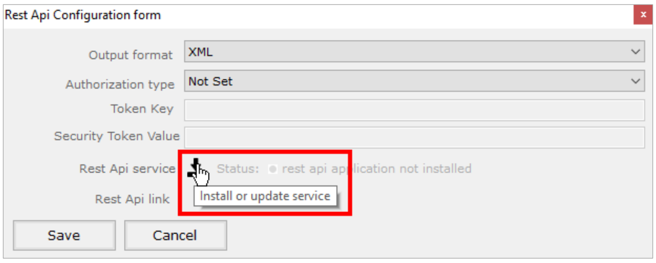
Rest Api service Status: ○ rest api application not installed

Rest Api link

If User choose the **Api Key** option, they must complete additional fields:

- **Token Key** – key name
- **Security Token Value** – the password for the key

For the first configuration or when the DBPLUS Rest API application has been removed, click the **[Install or update service]** button.



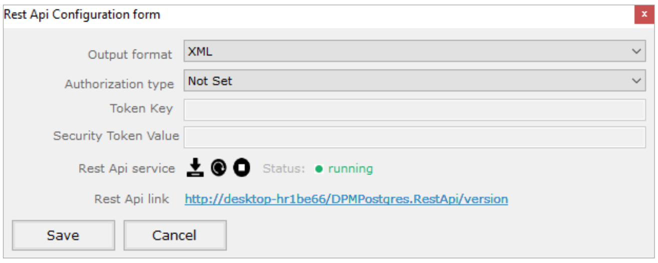
Rest Api Configuration form

Output format	XML
Authorization type	Not Set
Token Key	
Security Token Value	
Rest Api service	Status: rest api application not installed
Rest Api link	

After installing the Rest API application, Status: **running** and the link to the application - Rest Api link should be displayed:

http://hostname/DPMPostgres.RestApi/version

Clicking the link will run the version method that checks the version of the Performance Monitor application currently installed on the Windows server.



Rest Api Configuration form

Output format	XML
Authorization type	Not Set
Token Key	
Security Token Value	
Rest Api service	Status: ● running
Rest Api link	http://desktop-hr1be66/DPMPostgres.RestApi/version

When a port other than the standard 80 is used within the Performance Monitor application, the link will contain, in addition to the hostname, information about the configured port. Below is an example for port 82:

http://hostname:82/DPMPostgres.RestApi/version

1.1.1. Additional information

Domain authentication is not supported in the current version.
Setting the Security option at the Performance Monitor application level is not currently supported in the Rest API.

General description of integration:

- Filters do not support LIKE '% ... %'
- All filters are optional and do not need to be specified in the api call
- Date format: yyyy-mm-dd hh24:mi:ss
- In special cases, the website may return a field that does not apply to a given database platform, e.g. PostgreSQL - we leave it so that there is consistency between different platforms,

Each database platform has a separate REST API. Depending on the platform, the link calling the REST API will be different:

- For Oracle:
  http://hostname/DPMOracle.RestApi/
- For SQL SERVER
  http://hostname/DPM.RestApi/
- For PostgreSQL
  http://hostname/DPMPostgres.RestApi/

1.2. REST API call

In order to call the appropriate method, the appropriate method must be completed in the link that calls the REST API for the platform indicated. For example, below is a call to the **instancelist** method for the SQL SERVER platform.
An example of calling a method:

http://hostname/DPM.RestApi/instancelist

The call will return information about all instances connected to DBPLUS monitoring on the MS SQL platform.

The REST API allows to call a method with additional parameters. For this purpose, User can call the given method adding parameters for the link. For example, calling the **instancelist** method for the ORACLE platform with additional parametersi:

- **Id** – internal database identifier in DBPLUS
- **Category** – category assigned to the database

An example of calling a method:
https://hostname/DPMOracle.RestApi/instancelist?id=70&category=OTHER

1.3. REST API DBPLUS call methods

1.3.1. Version

Method	GET
Database platform	PostgreSQL, Oracle, MS SQL
Address	/version
Action	Gets the version and name of the monitored database platform

Input data: null

Output data:

ProductVersion	The version of the Performance Monitor application for your database platform
ToolName	Name of the DBPLUS application

Example [xml]:

```
<Root xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<ProductVersion>2021.2.3</ProductVersion>
<ToolName>MonitoringPOSTGRES</ToolName>
</Root>
```

Example [JSON]:

```
{"ProductVersion":"2021.2.3","ToolName":"MonitoringPOSTGRES"}
```

1.3.2. Instance list

Method	GET
Database platform	PostgreSQL, Oracle, MS SQL
Address	/instancelist
Action	Retrieves information about instances / databases added to the monitoring configuration (connected and not connected)

Input data:

MonitoringEnabled	Instances / databases monitoring included: ▪ true ▪ false
id	Instance ID
category	DBPLUS category assigned to the instance / database
Hostname	Host name
name	Instance name or connection name or base SID

Output data:

InstanceList	Instance list
InstanceInfoRecord	Instance details
ServerId	Server Ip in the DBPLUS repository
HostName	Host name
InstanceName	Instance name
DisplayedName	Name displayed in the DBPLUS application
InstanceId	Instance ID
Category	DBPLUS category assigned to the instance / database
Version	Instance / database version
MonitoringEnabled	Instances / databases monitoring included: ▪ True ▪ False
DbplusMonitoringUser	User designated for monitoring

Example [xml]:

```
<Root xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<InstanceList>
<InstanceInfoRecord>
<ServerId>1</ServerId>
<HostName>10.123.212.51</HostName>
<InstanceName>psqlrepo</InstanceName>
<DisplayedName>psqlrepo</DisplayedName>
<InstanceId>psqlrepo</InstanceId>
<Category>DBUX PRODUCTION</Category>
<Version>11.5 (Debian 11.5-1+deb10u1)</Version>
<MonitoringEnabled>true</MonitoringEnabled>
<DbplusMonitoringUser>dbplusrepo</DbplusMonitoringUser>
</InstanceInfoRecord>
<InstanceInfoRecord>
<ServerId>10</ServerId>
<HostName>10.125.1.20</HostName>
<InstanceName>prodbddb01</InstanceName>
<DisplayedName>prodbddb01</DisplayedName>
<InstanceId>prodbddb03</InstanceId>
<ServerType>PRODUCTION DATABASE</ServerType>
<Version>13.1</Version>
<MonitoringEnabled>true</MonitoringEnabled>
<DbplusMonitoringUser>dbplusmon</DbplusMonitoringUser>
</InstanceInfoRecord>
</InstanceList>
</Root>
```

Example [JSON]:

```
{"InstanceList":[{"HostName":"SQL11","InstanceName":"SQL11\\SQLMDR","DisplayedName":"SQL11\\SQLMDR","InstanceId":"SQL11\\SQLMDR","Category":"OTHER","Version":"2012","MonitoringEnabled":true,"DbplusMonitoringUser":"dbplus","ServerId":104,"StringServerId":"104"}]}
```

1.3.3. Dashboard status

Method	GET
Database platform	PostgreSQL, Oracle, MS SQL
Address	/dashboard
Action	Retrieves information about the statistics presented on the DBPLUS Dashboard screen

Input data:

isactive	Connection status
id	Instance ID
category	DBPLUS category assigned to the instance / database
Hostname	Host name
name	Instance name or connection name or base SID

Output data:

InstanceList	Instance list
InstanceInfoRecord	Instance record
ToolName	Name of the DBPLUS application
ServerId	Server ID
HostName	Host Name
InstanceName	Instance name
Category	DBPLUS category assigned to the instance / database
Version	Instance / database version
ProcesorsNumber	Number of processors * for PostgreSQL it returns null
InstanceProcesorsNumber	The number of processors assigned to the instance * for PostgreSQL it returns null
IsActive	Active connection to the DBPLUSCATCHER service: ▪ True ▪ False
IsOutage	Is the instance / database currently in Outage: ▪ True ▪ False
ActiveStatus	Instance / database status based on DBPLUS: ▪ -1 – Not connected ▪ 0 – instance in Outage status ▪ 1 – Performing Well ▪ 2 - Warning ▪ 3 - Critical
AlwaysOn_ActiveStatus	Always On status based on DBPLUS: ▪ -1 – Not connected ▪ 0 – instance in Outage status ▪ 1 – Performing Well ▪ 2 - Warning ▪ 3 – Critical ▪ Null – returned for Oracle, PostgreSQL versions
FailoverCluster_ActiveStatus	FailoverCluster status based on DBPLUS: ▪ -1 – Not connected ▪ 0 – instance in Outage status ▪ 1 – Performing Well ▪ 2 – Warning ▪ 3 – Critical ▪ Null – returned for Oracle, PostgreSQL versions
StandBy_ActiveStatus	Status Standby based on DBPLUS: ▪ -1 – Not connected ▪ 0 – instance in Outage status ▪ 1 – Performing Well ▪ 2 – Warning ▪ 3 – Critical ▪ Null – returned for MS SQL, PostgreSQL versions

UtilizationCPUServer	Server CPU usage [s/1s] * for PostgreSQL it returns null
UtilizationCPUInstance	Instance CPU usage [s/1s] * for PostgreSQL it returns null
UtilizationWaits	Waits Level [s/1s]
UtilizationWaitsIO	Waits IO Level [s/1s]
UtilizationWaitsLock	Lock level [s/1s]
UtilizationWaitsPercent	Waits Level [%] * for PostgreSQL it returns null
UtilizationWaitsIOPercent	Waits IO Level [%] * for PostgreSQL it returns null
UtilizationWaitsLockPercent	Lock level [%] * for PostgreSQL it returns null
UtilizationWaitsOther	Waits Other level [s/1s]
UtilizationCPUServerPercent	Server CPU usage [%] * for PostgreSQL it returns null
UtilizationCPUInstancePercent	Instance CPU usage [%] * for PostgreSQL it returns null
UtilizationSessionsActive	The number of active sessions
UtilizationSessionsLocked	The number of blocked sessions
UtilizationTransactions	the number of transactions
SpaceInfo	Information about the use of disk space
Total	Total Usage
Used	Occupied space
Free	Free space
ReasonAlertCritical	Number of Critical alerts for the last 2 hours
ReasonAlertWarning	Number of Warning alerts for the last 2 hours
Logdate	Date when the data was generated
ContainsAlwaysOn	Includes Always On: ▪ True ▪ False ▪ Null - returned for Oracle, PostgreSQL versions
ContainsFailoverCluster	Includes Failover Cluster: ▪ True ▪ False ▪ Null - returned for Oracle, PostgreSQL versions
ContainsStandBy	Includes Standby: ▪ True ▪ False ▪ Null - returned for MS SQL, PostgreSQL versions
ErrorInfo	Error information.

Example [xml]:

```
<Root xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<InstanceList>
<InstanceInfoRecord>
<ToolName>MonitoringMSSQL</ToolName>
<ServerId>70</ServerId>
<HostName>CRMSQL31</HostName>
<InstanceName>CRMSQL31</InstanceName>
<Category>OTHER</Category>
<Version>2014 (12.0.5207.0)</Version>
<ProcesorsNumber>16</ProcesorsNumber>
<InstanceProcesorsNumber>16</InstanceProcesorsNumber>
```

```
<CPUMachineSupported>true</CPUMachineSupported>
<CPUInstanceSupported>true</CPUInstanceSupported>
<IsActive>true</IsActive>
<IsOutage>false</IsOutage>
<ActiveStatus>1</ActiveStatus>
<AlwaysOn_ActiveStatus>-1</AlwaysOn_ActiveStatus>
<FailoverCluster_ActiveStatus>-1</FailoverCluster_ActiveStatus>
<StandBy_ActiveStatus>-1</StandBy_ActiveStatus>
<Replication_ActiveStatus>-1</Replication_ActiveStatus>
<UtilizationCPUServer>2.08</UtilizationCPUServer>
<UtilizationCPUInstance>2.08</UtilizationCPUInstance>
<UtilizationWaits>2.4</UtilizationWaits>
<UtilizationWaitsIO>0.14</UtilizationWaitsIO>
<UtilizationWaitsLock>0</UtilizationWaitsLock>
<UtilizationWaitsPercentToCpu>15</UtilizationWaitsPercentToCpu>
<UtilizationWaitsIOPercentToCpu>1</UtilizationWaitsIOPercentToCpu>
<UtilizationWaitsLockPercentToCpu>0</UtilizationWaitsLockPercentToCpu>
<UtilizationWaitsOther>2.26</UtilizationWaitsOther>
<UtilizationCPUServerPercent>13</UtilizationCPUServerPercent>
<UtilizationCPUInstancePercent>13</UtilizationCPUInstancePercent>
<UtilizationWaitsPercent>20</UtilizationWaitsPercent>
<UtilizationSessionsActive>4</UtilizationSessionsActive>
<UtilizationSessionsInactive>0</UtilizationSessionsInactive>
<UtilizationSessionsLocked>0</UtilizationSessionsLocked>
<UtilizationTransactions>0</UtilizationTransactions>
<SpaceInfo>
<ServerId>70</ServerId>
<Total>2206.3</Total>
<Used>1550.7</Used>
<Free>655.6</Free>
</SpaceInfo>
<ReasonAlertCritical>0</ReasonAlertCritical>
<ReasonAlertWarning>0</ReasonAlertWarning>
<Logdate>2021-08-23 11:20:30</Logdate>
<ContainsAlwaysOn>false</ContainsAlwaysOn>
<ContainsFailoverCluster>false</ContainsFailoverCluster>
<ContainsStandBy>false</ContainsStandBy>
<ContainsReplica>false</ContainsReplica>
<ErrorInfo/>
</InstanceInfoRecord>
</InstanceList>
</Root>
```

Example [JSON]:

```
{"InstanceList":[{"IsActive":true,"IsOutage":false,"ActiveStatus":1,"AlwaysOn_ActiveStatus":-
1,"FailoverCluster_ActiveStatus":-
1,"StandBy_ActiveStatus":null,"UtilizationCPUServer":3.2,"UtilizationCPUInstance":1.12,"UtilizationWaits":5.39,"
UtilizationWaitsIO":1.26,"UtilizationWaitsLock":0.01,"UtilizationWaitsOther":4.12,"UtilizationCPUServerPercent":
20,"UtilizationCPUInstancePercent":7,"UtilizationWaitsPercent":34,"UtilizationWaitsIOPercent":8,"UtilizationWai
tsLockPercent":0,"UtilizationSessionsActive":4,"UtilizationSessionsLocked":0,"UtilizationTransactions":0,"Logdate
":"2021-09-01
14:52:15","ContainsAlwaysOn":false,"ContainsFailoverCluster":false,"ContainsStandBy":null,"InstanceName":"CR
MSQL31","HostName":"CRMSQL31","ProcesorsNumber":16,"InstanceProcesorsNumber":16,"Version":"2014
(12.0.5207.0)","ServerId":70,"ToolName":"MonitoringMSSQL","Category":"OTHER","SpaceInfo":{"Total":2207.1,"
Used":1493.1,"Free":714.0,"UsedPercent":68,"FreePercent":32},"ReasonAlertCritical":0,"ReasonAlertWarning":0
,"ErrorInfo":""}]}
```

1.3.4. Alert Information

Method	GET
Database platform	PostgreSQL, Oracle, MS SQL
Address	/alerts
Action	Gets information about alerts in the monitored instance

Input data:

server_id	Instance ID
date_from	Date from which the alerts will be downloaded In format [YYYY:RR:DD HH:MM:SS]
date_to	Date to which alerts will be downloaded In format [YYYY:RR:DD HH:MM:SS]

No parameters on the input means that information about alerts for the last 2 hours is being downloaded

Output data:

ProblemsList	List of problems
ProblemInfoRecord	Details of the problem
ReasonId	The ID of the problem
ServerId	Instance / database ID
Class	Problem class
Name	Name of the event related to the problem
AlertsList	List of alerts
Alert	Information about the alert
AlertType	Alert type
AlertStatus	Alert status: ▪ Critical ▪ Warning
AlertId	Internal DBPLUS identifier of the alert
AlertStatisticName	The name of the statistics associated with the alert
Message	Alert message
QueryHashIdentifier	ID of the query associated with the alert
IsQueryAlert	Is an alert associated with the query: ▪ True ▪ False

Example [xml]:

```
<Root xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
<ProblemsList>
<ProblemInfoRecord>
<ReasonId>5329529</ReasonId>
<Logdate>2021-08-23 12:32:17</Logdate>
<ServerId>100</ServerId>
<Class>Lock</Class>
<Name>High LCK_M_IX event</Name>
<AlertsList>
<Alert>
<AlertType>Sql Query</AlertType>
<AlertStatus>Critical</AlertStatus>
<AlertId>query_et</AlertId>
<AlertStatisticName>Elapsed Time</AlertStatisticName>
<Message>Alert Type: Sql Query, The measured statistic value is 27,1 times higher than allowed maximum ,
Statement query hash: 0xCAA8349B9AB73044, Statistics: Elapsed Time, Last value: 304,0 s, History value: 10,8
s </Message>
<QueryHashIdentifier>0xCAA8349B9AB73044</QueryHashIdentifier>
<IsQueryAlert>true</IsQueryAlert>
</Alert>
<Alert>
<AlertType>Sql Query</AlertType>
<AlertStatus>Critical</AlertStatus>
```

```
<AlertId>query_et1</AlertId>
<AlertStatisticName>Elapsed Time per 1 exec</AlertStatisticName>
<Message>Alert Type: Sql Query, The measured statistic value is 17,7 times higher than allowed maximum , Statement query hash: 0xCAA8349B9AB73044, Statistics: Elapsed Time per 1 exec, Last value: 0,3436 s, History value: 0,0183 s </Message>
<QueryHashIdentifier>0xCAA8349B9AB73044</QueryHashIdentifier>
<IsQueryAlert>true</IsQueryAlert>
</Alert>
</AlertsList>
</ProblemInfoRecord>
</ProblemsList>
</Root>
```

Example [JSON]:

```
{"ProblemsList":[{"ReasonId":5335636,"Logdate":"2021-09-01 17:07:02","ServerId":203,"Class":"Lock","Name":"High LCK_M_U event","AlertsList":[{"AlertType":"Sql Query","AlertStatus":"Critical","AlertId":"query_et","AlertStatisticName":"Elapsed Time","Message":"Alert Type: Sql Query, The measured statistic value is 2,3 times higher than allowed maximum , Statement query hash: 0xD388D40A35DB4D8F, Statistics: Elapsed Time, Last value: 671,9 s, History value: 205,8 s ","QueryHashIdentifier":"0xD388D40A35DB4D8F","IsQueryAlert":true},{"AlertType":"Sql Query","AlertStatus":"Critical","AlertId":"query_et1","AlertStatisticName":"Elapsed Time per 1 exec","Message":"Alert Type: Sql Query, The measured statistic value is 5,5 times higher than allowed maximum , Statement query hash: 0xD388D40A35DB4D8F, Statistics: Elapsed Time per 1 exec, Last value: 0,0070 s, History value: 0,001077 s ","QueryHashIdentifier":"0xD388D40A35DB4D8F","IsQueryAlert":true}]}]}
```

2 Anomaly Monitor

As part of the latest version of Anomaly monitor, we have introduced a number of fixes and improvements in the anomaly search module in the monitored database. The most important changes are presented below:

- Improved search and presentation of information about the session being the main blocker.

The change is to improve the search for the session that is the root cause of the lockout problem. Information is always presented about the session blocking other sessions whose sum of waiting time is the highest in a given time interval.

- Improved New Statements detection - support for the same query versions.

The Anomaly Monitor feature alerts users to new database queries that affect performance. When a new query (with a new identifier) starts in the database, the application will analyze the content of the query and verify that a similar query has not been run before.

- Problem with the presentation of the Space Size chart on the Anomaly report.
- New detection that informs about high CPU load.

When a CPU is assigned to a database it is heavily disposed of. This can slow down some business processes. This is due to the higher wait for access to the CPU. This is true even if the CPU on the machine is not fully used. Detection verifies the CPU load level. In the case of high utilization, the CPU verifies that it does not adversely affect orders that use the CPU. When the performance of queries related to higher CPU expectations decreases, such information will be presented in Anomaly Monitor, in addition, appropriate communication will be sent to users via e-mail (note: only in the case of a configured mail server in the Performance Monitor application).

- New detection to increase the wait trend without affecting database performance.

The latest version of Anomaly Monitor verifies and checks the trends of top waits in the monitored database. When there is an increase in the duration of one of the top waits, Users will receive information about it and about queries related to the wait.

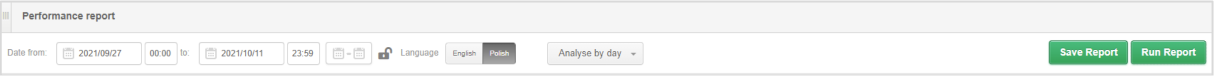
- Improved detection for queries that have constant performance issue.

For queries for which performance degradation occurs daily at certain times or at fixed intervals (e.g., it performs a constant business process), for such cases, additional logic is performed, performance is compared over a longer period of time, and if no performance degradation is found, no problem is reported efficient.

3 Performance Report

The ability to generate a new report has been added to the latest version of the application. The report contains information about the most important performance statistics for a given database. Contains data on top queries performed in the database in a selected period of time.
The report is available at the detail level of a given database in the **Reports - Performance report menu**.
The previous report, which contained information about top queries, is available to be generated in the **Top heavy queries menu**.

The report can be generated for the selected period, as well as there is option to choose the language version. Additionally, we can set the accuracy of the generated charts (option by day and by hour). The report generation starts after clicking **Run Report**. The report is visible in the application. Saving the report as * .docx is available after clicking the **Save Report** button.



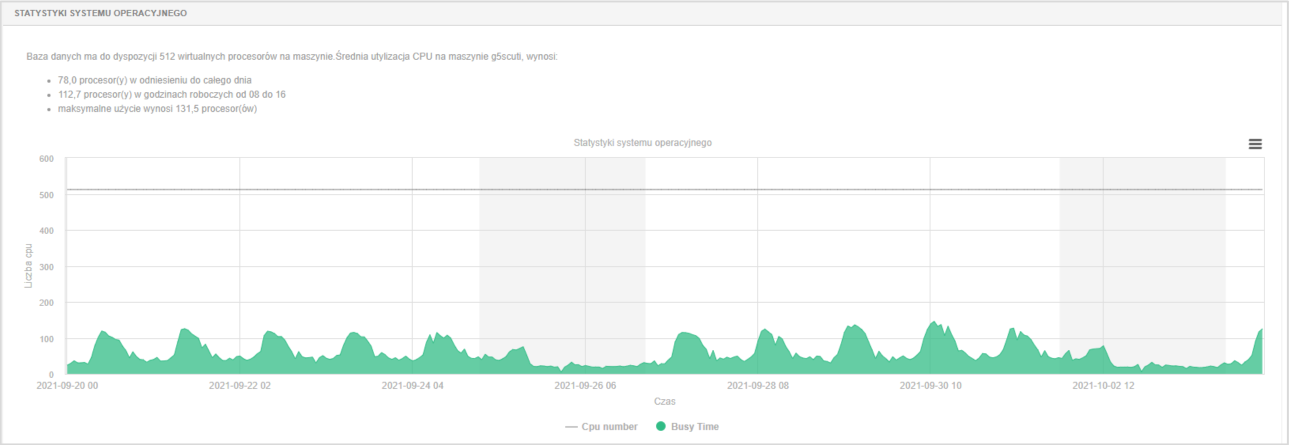
Performance report

Date from: 2021/09/27 00:00 to: 2021/10/11 23:59 Language English Polish Analyse by day

The report contains information about:

- General database description (includes basic database parameters)
- Performance description

- operating system statistics that present the database CPU load

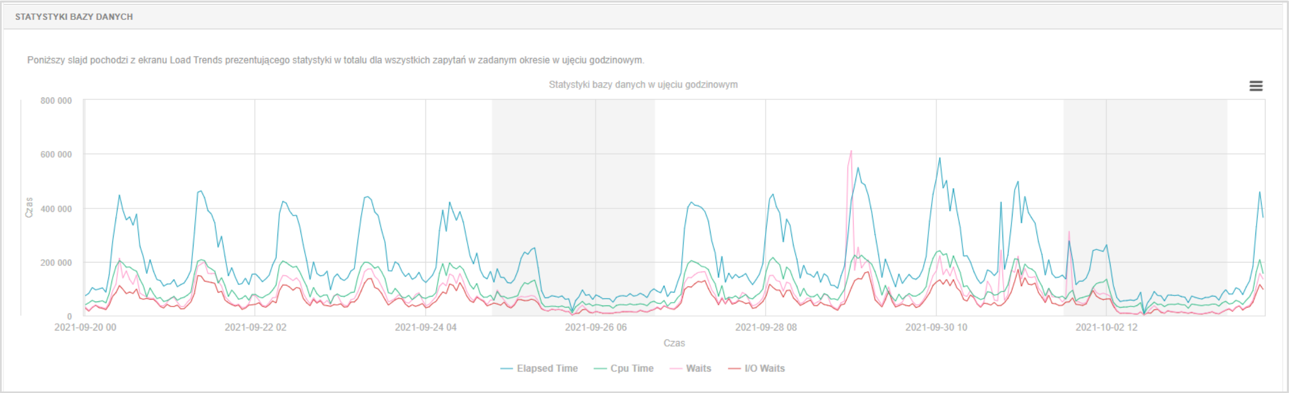


STATYSTYKI SYSTEMU OPERACYJNEGO

Baza danych ma do dyspozycji 512 wirtualnych procesorów na maszynie.Średnia utylizacja CPU na maszynie g5scuti, wynosi:

- 78,0 procesor(y) w odniesieniu do całego dnia
- 112,7 procesor(y) w godzinach roboczych od 08 do 16
- maksymalne użycie wynosi 131,5 procesor(ów)

- database statistics that show the total duration of queries, waiting time, IO waiting, locks and Latches.



STATYSTYKI BAZY DANYCH

Poniższy slajd pochodzi z ekranu Load Trends prezentującego statystyki w totalu dla wszystkich zapytań w zadanym okresie w ujęciu godzinowym.

- top wait stats

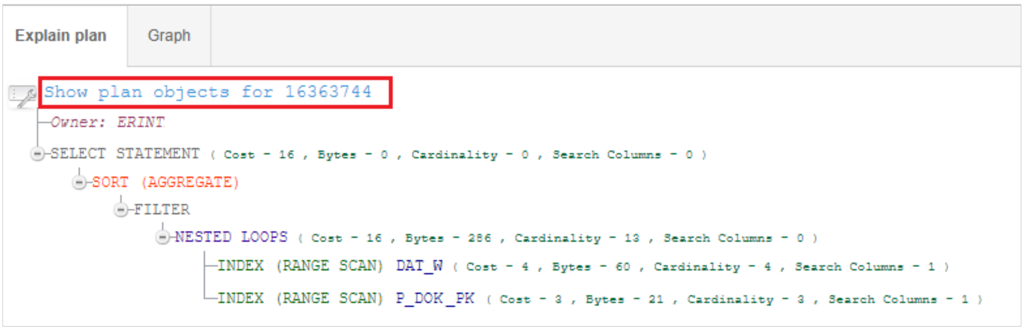
- Performance description for the last 6 months
- Top queries

This chapter presents information on the most burdensome queries in the analyzed period:

- the text of the query,
- characteristics,
- summary of statistics for the selected period,
- graph of duration statistics for a selected period,
- performance plans or plans.

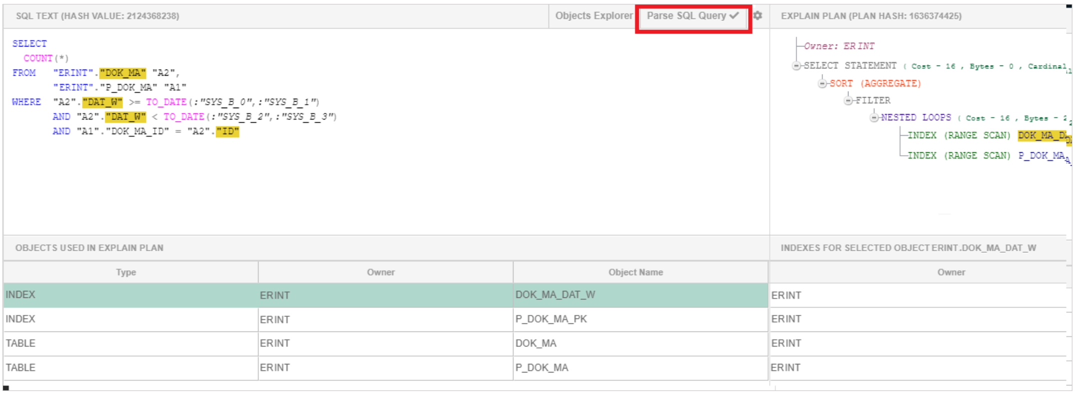
4 DBPLUS Query Advisor

In the latest version of the application, we have added the function of suggesting changes to improve query performance - **Query advisor**. The function is available on the **Plan Objects** screen to analyze queries in detail.
The website can be accessed from any query for which an execution plan has been collected.



```
Show plan objects for 16363744
 Owner: ERINT
 SELECT STATEMENT ( Cost - 16 , Bytes - 0 , Cardinality - 0 , Search Columns - 0 )
   SORT (AGGREGATE)
     FILTER
       NESTED LOOPS ( Cost - 16 , Bytes - 286 , Cardinality - 13 , Search Columns - 0 )
         INDEX (RANGE SCAN) DAT_W ( Cost - 4 , Bytes - 60 , Cardinality - 4 , Search Columns - 1 )
         INDEX (RANGE SCAN) P_DOK_PK ( Cost - 3 , Bytes - 21 , Cardinality - 3 , Search Columns - 1 )
```

The Query advisor function is available only after manual query parsing using the [**Parse SQL Query**] button available on the Plan Objects page.


SQL TEXT (HASH VALUE: 2124368238)
Objects Explorer
Parse SQL Query ✓
EXPLAIN PLAN (PLAN HASH: 1636374425)

```
SELECT
  COUNT(*)
FROM   "ERINT"."DOK_MA" "A2",
       "ERINT"."P_DOK_MA" "A1"
WHERE  "A2"."DAT_W" >= TO_DATE(:"SYS_B_0",:"SYS_B_1")
       AND "A2"."DAT_W" < TO_DATE(:"SYS_B_2",:"SYS_B_3")
       AND "A1"."DOK_MA_ID" = "A2"."ID"
```

```
 Owner: ERINT
 SELECT STATEMENT ( Cost - 16 , Bytes - 0 , Cardinal
   SORT (AGGREGATE)
     FILTER
       NESTED LOOPS ( Cost - 16 , Bytes - 2
         INDEX (RANGE SCAN) DOK_MA_D
         INDEX (RANGE SCAN) P_DOK_MA
```

OBJECTS USED IN EXPLAIN PLAN

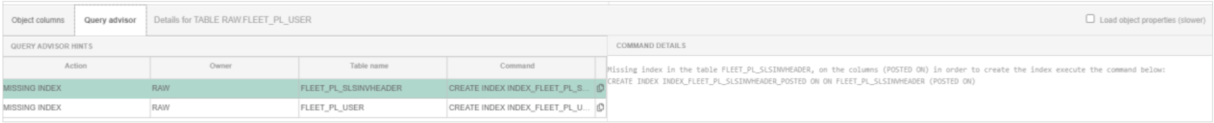
Type	Owner	Object Name
INDEX	ERINT	DOK_MA_DAT_W
INDEX	ERINT	P_DOK_MA_PK
TABLE	ERINT	DOK_MA
TABLE	ERINT	P_DOK_MA

INDEXES FOR SELECTED OBJECT ERINT.DOK_MA_DAT_W

Owner
ERINT
ERINT
ERINT
ERINT

After parsing the query, depending on the selected object in the **Objects Used in Explain Plan** table, the query fragments that contain the data that are part of the given object (index or table) are highlighted.

Information containing a hint to improve performance will be visible in the new **Query Advisor** tab at the bottom of the page.
In the current version, the Query advisor function verifies whether each of the tables involved in the query has the optimal index for the execution of the analyzed query. If the index is missing, the information will be presented in the aforementioned tab i.


Object columns
Query advisor
Details for TABLE RAW.FLEET_PL_USER
□ Load object properties (slower)

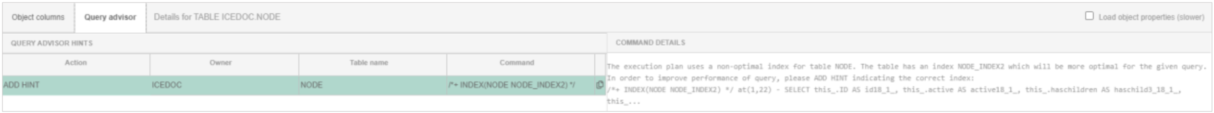
QUERY ADVISOR HINTS

Action	Owner	Table name	Command
MISSING INDEX	RAW	FLEET_PL_SLSINVHEADER	CREATE INDEX INDEX_FLEET_PL_S...
MISSING INDEX	RAW	FLEET_PL_USER	CREATE INDEX INDEX_FLEET_PL_U...

COMMAND DETAILS

```
Missing index in the table FLEET_PL_SLSINVHEADER, on the columns (POSTED_ON) in order to create the index execute the command below:
CREATE INDEX INDEX_FLEET_PL_SLSINVHEADER_POSTED_ON ON FLEET_PL_SLSINVHEADER (POSTED_ON)
```

As part of the query analysis, the currently used indexes for the verification of the execution plan analysis are also verified. When Users find that a given table has an index that is more optimal for a given query, but it is not currently used in the execution plan, in this case, information about the possibility of adding HINT to the query content will be presented.



Object columns | Query advisor | Details for TABLE ICEDOC.NODE

□ Load object properties (slower)

QUERY ADVISOR HINTS

Action	Owner	Table name	Command
ADD HINT	ICEDOC	NODE	/*+ INDEX(NODE NODE_INDEX2) */

COMMAND DETAILS

```
The execution plan uses a non-optimal index for table NODE. The table has an index NODE_INDEX2 which will be more optimal for the given query.
In order to improve performance of query, please ADD HINT indicating the correct index:
/*+ INDEX(NODE NODE_INDEX2) */ at(1,22) - SELECT this_.ID AS id18_1_, this_.active AS active18_1_, this_.haschildren AS haschild3_18_1_,
this_...
```

For some queries, a message may be displayed that informs that there is no recommendation and that the query is not supported in the current version. In each subsequent release of the application version, information on the possibilities of query optimization will be supplemented and updated.
Before introducing a change, we always ask for verification and additional tests in the test environment whether the changes proposed by the tool (index assumption or a new hint) have a positive effect on the performance of the query for which the hint was presented.

5 Small fixes and improvements

5.1. Improvement of the charts in the Top heavy queries report

We have corrected the problem of presenting some graphs while the report is being generated. The graphs in the report were not generating correctly in some versions of browsers. The problem has been fixed in the latest version.
In the latest version of the application, due to the addition of a new report, the current name of the report has been changed from **Performance Report** to **Top heavy queries.**

5.2. Call, exec, execute queries are moved to procedure monitoring

In the latest version of the application, information about query statistics that call procedures or functions such as:

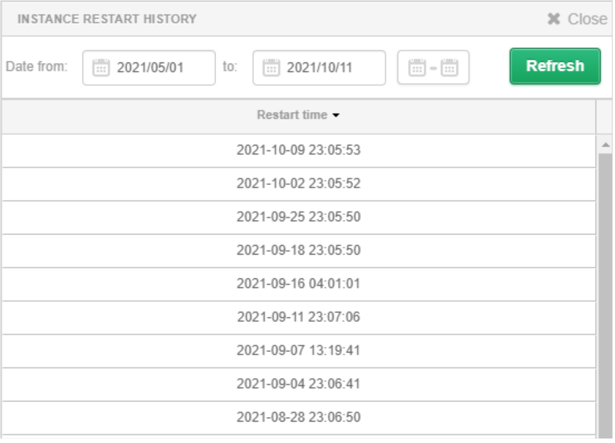
- Call ...
- Exec...
- Execute...

have been moved to monitoring procedures. Information that includes statistics for procedures can be found at the detail level of a given database on the pages:

- Database Load in the Procedure tab
- TopSQL / SQL 3d with report type: Top 20 procedures
- Top Day by checking the Procedures checkbox
- SQL Details by viewing the detailed statistics for a given query.

5.3. Database Restart Browser

From the latest version, the Performance Monitor application will collect information about restarting the database. This information can be obtained by clicking on the icon in the upper right corner of the screen on the level of database details.



INSTANCE RESTART HISTORY

Date from: 2021/05/01 to: 2021/10/11

Restart time ▾
2021-10-09 23:05:53
2021-10-02 23:05:52
2021-09-25 23:05:50
2021-09-18 23:05:50
2021-09-16 04:01:01
2021-09-11 23:07:06
2021-09-07 13:19:41
2021-09-04 23:06:41
2021-08-28 23:06:50