DBPLUS
better performance

DBPLUS
Performance Monitor for PostgreSQL
description of changes in version 2022.2

Date: July 10, 2022

Table of contents:

1 Session Trace profiler 3
2 REST API – Performance Monitor 4
2.1. IO Stats 5
3 Anomaly monitor improvements 7
4 Google Cloud and AWS Cloud support 7
5 Bug fixes and improvements 10
5.1. Improved SQL Parser 10
5.2. Implementation of TLS1.1 and TLS1.2 support 11

Below is a list of changes to the DBPLUS Performance Monitor system for PostgreSQL database monitoring.

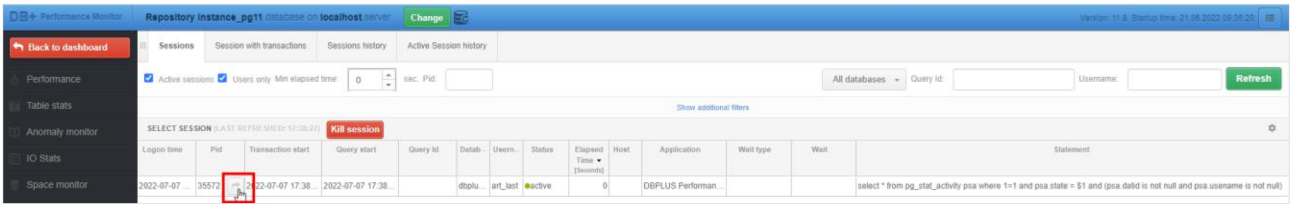
New in 2022.2 version

1 Session Trace profiler

In the latest version of the application, we have added the functionality of session monitoring using Trace Profiler. This option is available from the level of each monitored instance from the ***Sessions*** menu. The functionality allows you to monitor the users' session regardless of the functionality available in the application by default.

Session monitoring can be started in two ways:

- by clicking the button in the ***Pid*** column for a dedicated session on the online session screen:

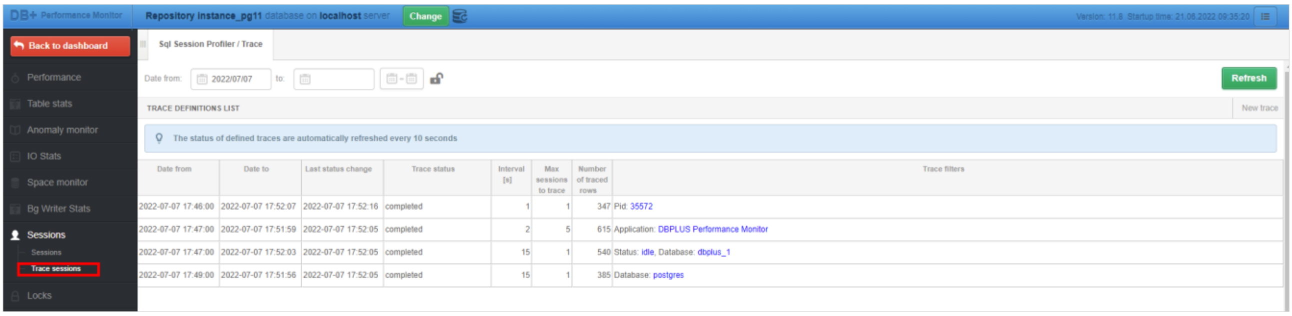


☑ Active sessions ☑ Users only Min elapsed time: 0 sec. Pid: All databases Query Id: Username:

SELECT SESSION (LAST REFRESHED: 17:38:27)

Logon time	Pid		Transaction start	Query start	Query Id	Datab...	Usern...	Status	Elapsed Time [Seconds]	Host	Application	Wait type	Wait	Statement
2022-07-07 ...	35572		2022-07-07 17:38...	2022-07-07 17:38...		dbplu...	art_last	● active	0		DBPLUS Performan...			select * from pg_stat_activity psa where 1=1 and psa.state = $1 and (psa.datid is not null and psa.usename is not null)

- from the **Sessions> Trace profiler** menu by clicking **[New trace]:**



Date from: 2022/07/07 to:

TRACE DEFINITIONS LIST

The status of defined traces are automatically refreshed every 10 seconds

Date from	Date to	Last status change	Trace status	Interval [s]	Max sessions to trace	Number of traced rows	Trace filters
2022-07-07 17:46:00	2022-07-07 17:52:07	2022-07-07 17:52:16	completed	1	1	347	Pid: 35572
2022-07-07 17:47:00	2022-07-07 17:51:59	2022-07-07 17:52:05	completed	2	5	615	Application: DBPLUS Performance Monitor
2022-07-07 17:47:00	2022-07-07 17:52:03	2022-07-07 17:52:05	completed	15	1	540	Status: idle, Database: dbplus_1
2022-07-07 17:49:00	2022-07-07 17:51:56	2022-07-07 17:52:05	completed	15	1	385	Database: postgres

In both cases, after clicking, a dedicated window will appear in which we can set the conditions with which the session monitoring is to be started.

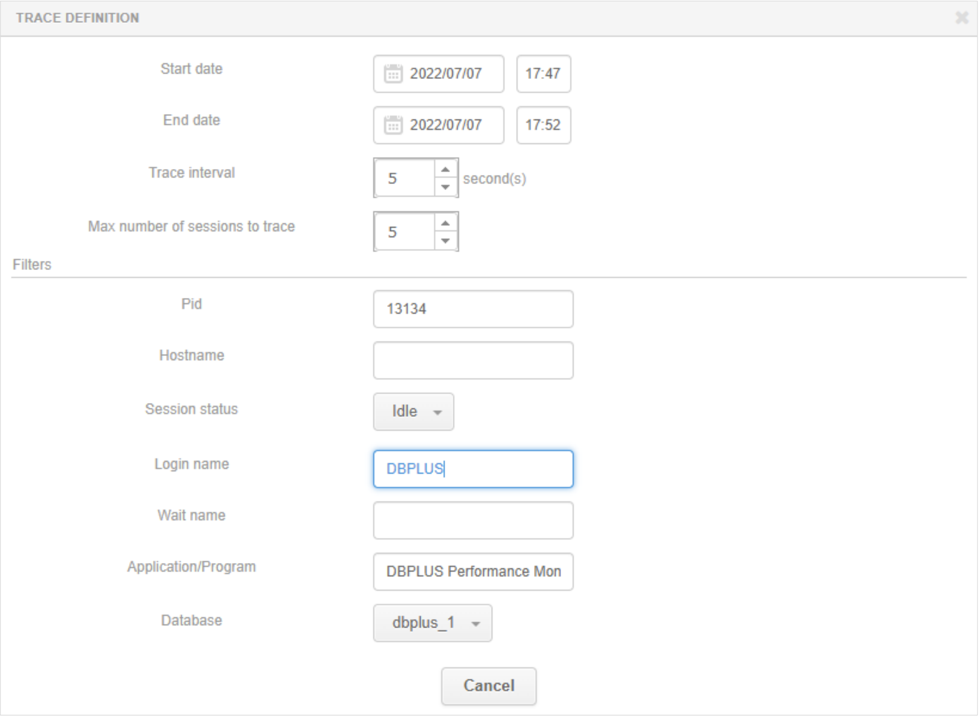
The basic settings include:

- Start date, End date - start and end time of session monitoring,
- Trace interval - the interval of retrieving information about sessions,
- Max number of session to trace - the maximum number of monitored sessions.

The minimum interval for session monitoring is 1 second. It means that every second a command with given conditions will be executed checking information about the session.

By setting up session monitoring, we can configure filters for:

- Pid – session ID
- Hostname
- Session status
- Login name
- Wait name
- Application/Program – the name of the application / program
- Database - database name



TRACE DEFINITION

Start date: 2022/07/07 17:47

End date: 2022/07/07 17:52

Trace interval: 5 second(s)

Max number of sessions to trace: 5

Filters

Pid: 13134

Hostname:

Session status: Idle

Login name: DBPLUS

Wait name:

Application/Program: DBPLUS Performance Mon

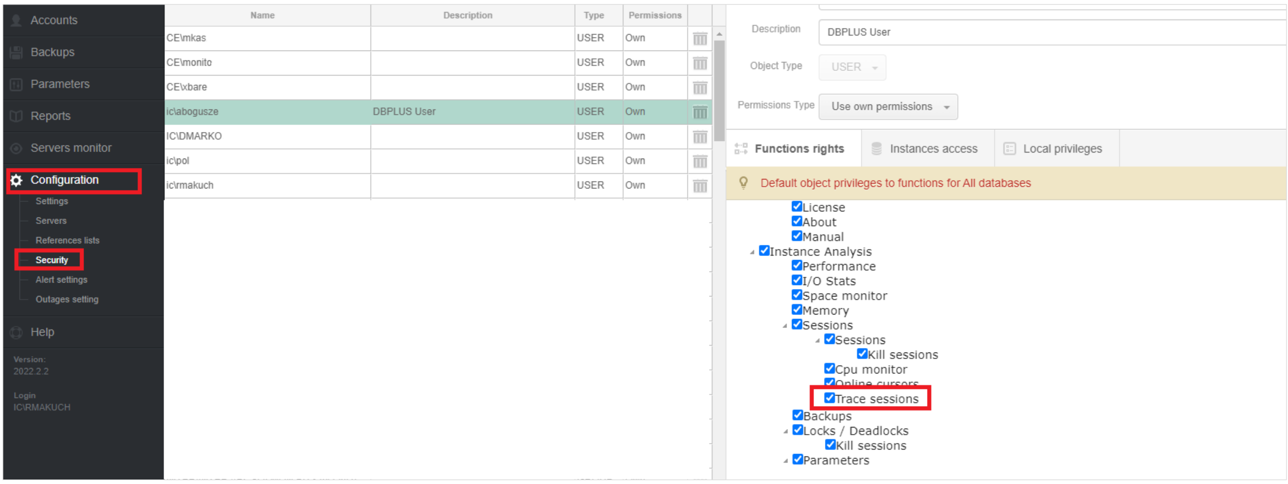
Database: dbplus_1

After starting the session monitoring, the session data will be presented at the bottom of the screen after clicking on the selected row. The monitoring screen refreshes automatically once every 10 seconds. During the session monitoring, the user has the option to modify the filters or stop the monitoring currently being performed.
When the session monitoring is completed, the (Trace status) will change to completed.

The user can delete previously performed monitoring sessions, or wait for them to be deleted automatically based on the parameter set for the length of data storage in the repository in the menu ***Configuration> Settings> History settings section, Session / Locks statistics***.

Access to the Trace Profiler menu

If the **Trace profiler** option is not visible in the **Sessions** menu, please verify that the access to this menu has been granted (by default the Trace profiler menu is invisible). To grant access, on the Dashboard screen in the main menu, select Configuration> Security and then grant appropriate permissions.



Name	Description	Type	Permissions
CE\mkas		USER	Own
CE\monito		USER	Own
CE\xbare		USER	Own
ic\abogusze	DBPLUS User	USER	Own
IC\DMARKO		USER	Own
ic\pol		USER	Own
ic\rmakuch		USER	Own

Description: DBPLUS User

Object Type: USER

Permissions Type: Use own permissions

Default object privileges to functions for All databases

- ☑License
- ☑About
- ☑Manual
- ☑Instance Analysis
  - ☑Performance
  - ☑I/O Stats
  - ☑Space monitor
  - ☑Memory
  - ☑Sessions
    - ☑Sessions
      - ☑Kill sessions
    - ☑Cpu monitor
    - ☑Online cursors
    - ☑Trace sessions
  - ☑Backups
  - ☑Locks / Deadlocks
    - ☑Kill sessions
  - ☑Parameters

2 REST API – Performance Monitor

In the latest version of the application, we have added new method to the REST API:
- get information about IO Stats disk array statistics.

2.1. IO Stats

Method	GET
Database platform	PostgreSQL, Oracle, MS SQL (from version 2022.2)
URL	/iostats
Action	Gets information about IO statistics

Input data:

View	▪ last_snapshot ▪ history * in the case of the history option, additional filters must be completed (group_time, date_from, date_to)
server_id	Server identifier in the DBPLUS repository
group_time	Grouping of returned data (return date format): ▪ year (YYYY) ▪ month (YYYY-MM) ▪ day (YYYY-MM-DD) ▪ hour (YYYY-MM-DD HH24) ▪ snap (YYYY-MM-DD HH24:MI:SS)
group_type	Group type: ▪ Database ▪ Tablespace ▪ File * no field completed means options without grouping
date_to	Date by which statistics will be downloaded *In format YYYY-MM-DD HH24:MI:SS
date_from	Date from which statistics will be downloaded *In format YYYY-MM-DD HH24:MI:SS
database	Database name * used in MSSQL, PostgreSQL
tablespace	Tablespace name *used in Oracle
file_name	File name *used in MSSQL, Oracle

Output data:

IODataList	List of IO statistics
IOStatRecord	Record of IO statistics
Database	Database name
Tablespace	Tablespace name
File	File name
Logdate	Date for the given statistic
NumberOfReads	Number of reads
NumberOfWrites	Number of writes
BytesReads	The number of bytes read
BytesWrites	Number of bytes written
MBytesReads	The number of reads in [MB]
MBytesWrites	The number of writes in [MB]
BlockReads	The number of blocks read
BlockWrites	Number of blocks written
ReadTime	Reading time
WriteTime	Writing time
SingleMByteReadTime	Single MB read time
SingleMByteWriteTime	Single MB write time
SingleBlockReadTime	Single block read time
SingleBlockWriteTime	Single block write time

Example output data [xml]:

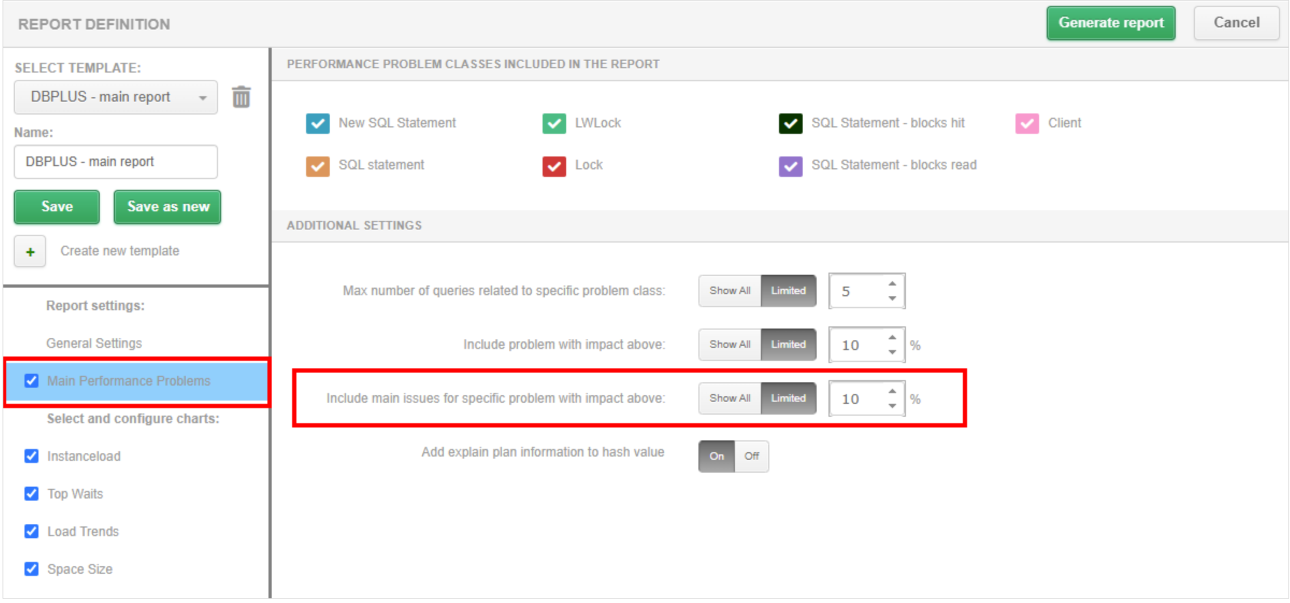
```xml
<?xml version="1.0" encoding="utf-16"?>
<Root xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <IODataList>
        <IOStatRecord>
            <Database>All databases</Database>
            <Tablespace>N/A</Tablespace>
            <File>All files</File>
            <Logdate>2022-06-15 11:28:50</Logdate>
            <NumberOfReads>739</NumberOfReads>
            <NumberOfWrites>637</NumberOfWrites>
            <BytesReads>48234496</BytesReads>
            <BytesWrites>5242880</BytesWrites>
            <MBytesReads>46</MBytesReads>
            <MBytesWrites>5</MBytesWrites>
            <BlockReads>5888</BlockReads>
            <BlockWrites>640</BlockWrites>
            <ReadTime>1.61</ReadTime>
            <WriteTime>2.112</WriteTime>
            <SingleMByteReadTime>0.035</SingleMByteReadTime>
            <SingleMByteWriteTime>0.4224</SingleMByteWriteTime>
            <SingleBlockReadTime>0.000273</SingleBlockReadTime>
            <SingleBlockWriteTime>0.0033</SingleBlockWriteTime>
        </IOStatRecord>
    </IODataList>
    <Response>
        <Status>OK</Status>
        <Message />
    </Response>
</Root>
```

Example output data [JSON]:

```json
"IOStatList": [
        {
            "Database": "All databases",
            "Tablespace": "N/A",
            "File": "%C:\\Program Files\\Microsoft SQL Server\\MSSQL15.SQL_2019\\MSSQL\\DATA\\ABCD%%",
            "Logdate": "2022-06-28 14:17:10",
            "NumberOfReads": 85,
            "NumberOfWrites": 25875,
            "BytesReads": 4194304,
            "BytesWrites": 238026752,
            "MBytesReads": 4,
            "MBytesWrites": 227,
            "BlockReads": 512,
            "BlockWrites": 29056,
            "ReadTime": 0.343,
            "WriteTime": 69.303,
            "SingleMByteReadTime": 0.08575,
            "SingleMByteWriteTime": 0.305300,
            "SingleBlockReadTime": 0.000670,
            "SingleBlockWriteTime": 0.002385
        }
    ],
    "Response": {
        "Status": "OK",
        "Message": ""
    }
}
```

3 Anomaly monitor improvements

In the latest version of the application, we have made changes to the process of generating the Anomaly Monitor report containing the performance anomalies detected by the Performance Monitor applications. The changes consist in including in the report only the largest cases of a given problem in the analyzed period of time. The change consists in adding a dedicated parameter which is responsible for filtering out the occurrences of a given problem in the period for which the report is generated and taking into account only those occurrences that exceed the threshold indicated in the parameter. The parameter value is set to **10%** by default. This will allow the Anomaly Monitor report not to include information about problem occurrences that are not the main performance problem in the monitored instance.



REPORT DEFINITION

SELECT TEMPLATE: DBPLUS - main report

Name: DBPLUS - main report

Report settings:

General Settings

☑ Main Performance Problems

Select and configure charts:

☑ Instanceload

☑ Top Waits

☑ Load Trends

☑ Space Size

PERFORMANCE PROBLEM CLASSES INCLUDED IN THE REPORT

☑ New SQL Statement ☑ LWLock ☑ SQL Statement - blocks hit ☑ Client

☑ SQL statement ☑ Lock ☑ SQL Statement - blocks read

ADDITIONAL SETTINGS

Max number of queries related to specific problem class: Show All [✓ Limited] 5

Include problem with impact above: Show All [✓ Limited] 10 %

Include main issues for specific problem with impact above: Show All [✓ Limited] 10 %

Add explain plan information to hash value [✓ On] Off

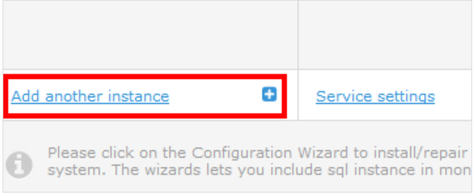
If the user would like the report to describe all performance problems and all occurrences of the problem, it is enough to change the value for a given parameter to **Show All**, which will mean that all problems will be described in the report.

4 Google Cloud and AWS Cloud support

In the latest version, we have added support for monitoring PostgreSQL instances on the Google Cloud and AWS Cloud platforms. The range of functionalities available in the DBPLUS Performance Monitor application is the same as for the on premise version. Below is a shortened scenario of adding a PostgreSQL instance to DBPLUS monitoring.

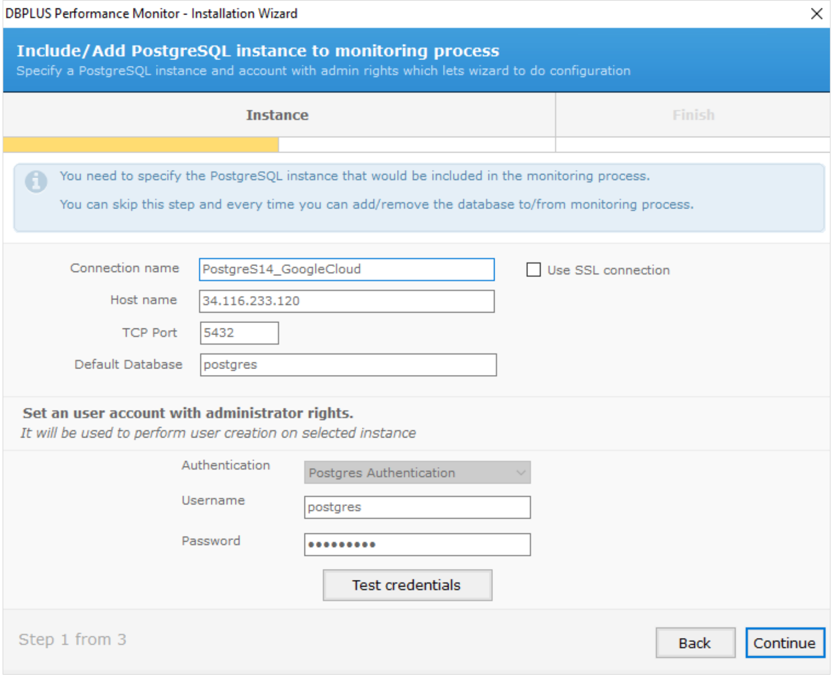
Adding a PostgreSQL instance on the Google or AWS Cloud platform

PostgreSQL instances installed on cloud platforms for DBPLUS monitoring can be added in the same way as for the on premise version, by clicking the **[Add another instance]** button from the DBPLUS Configuration Wizard.



Please click on the Configuration Wizard to install/repair system. The wizards lets you include sql instance in mon

On the next screen, complete the connection data of the PostgreSQL instance and the user name. In a PostgreSQL instance on cloud platforms, we do not have access to a user with **SUPERUSER** privileges, so we fill in the user data on the screen, which will allow to create a new user as well as to grant the privileges needed to monitor the instance.



DBPLUS Performance Monitor - Installation Wizard

Include/Add PostgreSQL instance to monitoring process
Specify a PostgreSQL instance and account with admin rights which lets wizard to do configuration

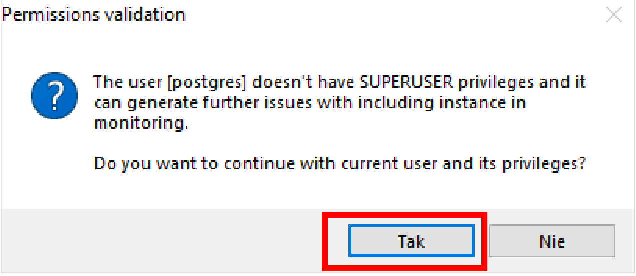
You need to specify the PostgreSQL instance that would be included in the monitoring process.
You can skip this step and every time you can add/remove the database to/from monitoring process.

Connection name: PostgreS14_GoogleCloud
□ Use SSL connection
Host name: 34.116.233.120
TCP Port: 5432
Default Database: postgres

Set an user account with administrator rights.
It will be used to perform user creation on selected instance

Authentication: Postgres Authentication
Username: postgres
Password: •••••••••

After clicking the **[Continue]** button, accept the selection and proceed to the next stage of configuration.



Permissions validation

The user [postgres] doesn't have SUPERUSER privileges and it can generate further issues with including instance in monitoring.

Do you want to continue with current user and its privileges?

In the next step, we choose the option to create a new one (recommended option) or to select an existing user for monitoring.

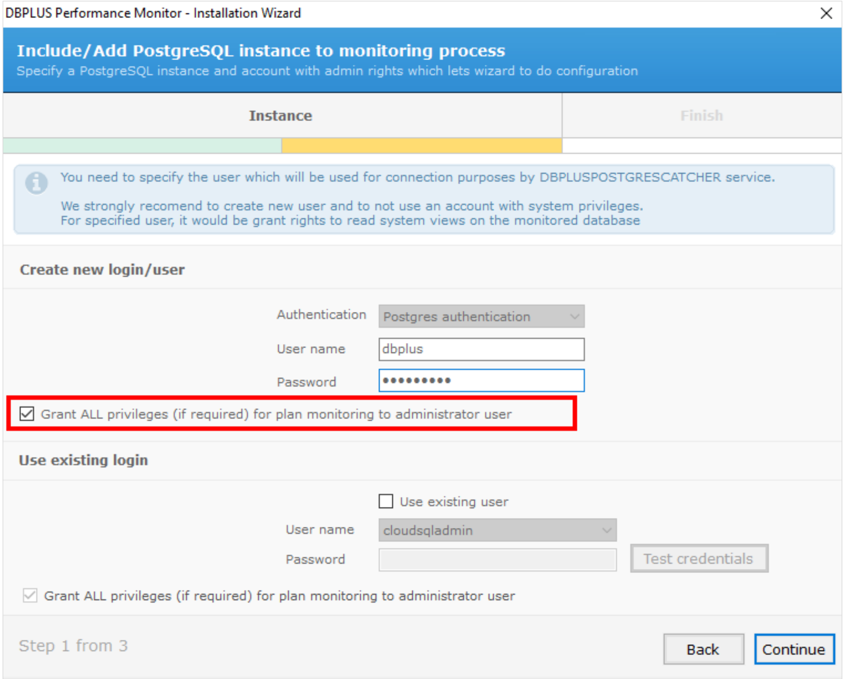
Monitoring of execution plans and statistics for the objects participating in the query in the PostgreSQL instance on the cloud platform requires additional authorizations for the monitoring user. For this purpose, the user indicated for monitoring [monitoring_user] must have the following rights:

grant [owners] to [monitoring_user];

Where [owners] refers to all owners who own properties on the PostgreSQL instance.

Additionally, in order to perform the function that generates execution plans for queries, administrator rights are granted to all databases in the PostgreSQL instance.

Grant all on database [databases_name] to [administartor_user];



DBPLUS Performance Monitor - Installation Wizard

Include/Add PostgreSQL instance to monitoring process

Specify a PostgreSQL instance and account with admin rights which lets wizard to do configuration

You need to specify the user which will be used for connection purposes by DBPLUSPOSTGRESCATCHER service.

We strongly recomend to create new user and to not use an account with system privileges.
For specified user, it would be grant rights to read system views on the monitored database

Create new login/user

Authentication: Postgres authentication

User name: dbplus

Password: •••••••••

☑ Grant ALL privileges (if required) for plan monitoring to administrator user

Use existing login

□ Use existing user

User name: cloudsqladmin

Password:

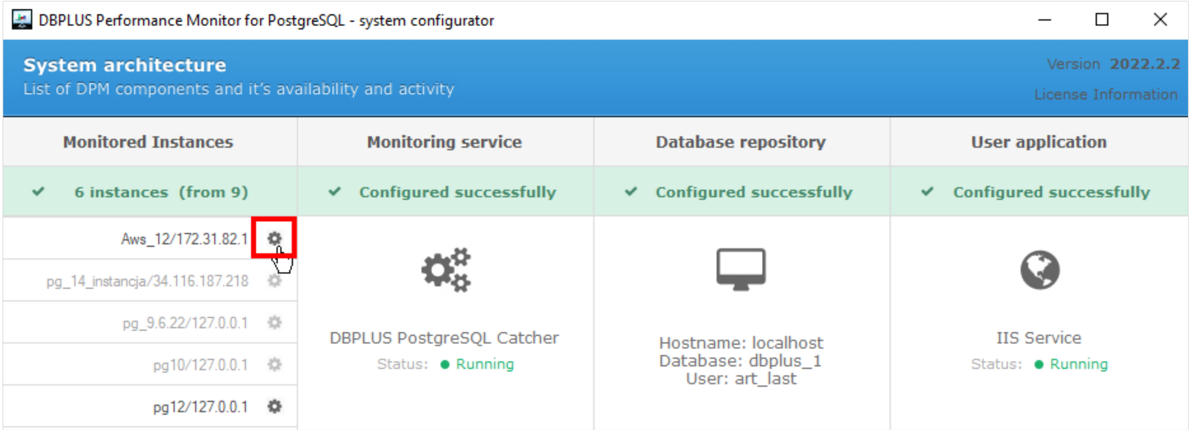
☑ Grant ALL privileges (if required) for plan monitoring to administrator user

Step 1 from 3

Finally, we accept the configuration and finish the installation by clicking the **[Finish]** button.

Monitoring user privileges refreshing

If a new database has been added on the PostgreSQL instance, it is necessary to refresh the monitoring user's permissions. To do this, run the DBPLUS Configuration Wizard program located on the Windows server with the DBPLUS application. Then go to the settings for the dedicated PostgreSQL instance:



DBPLUS Performance Monitor for PostgreSQL - system configurator

System architecture

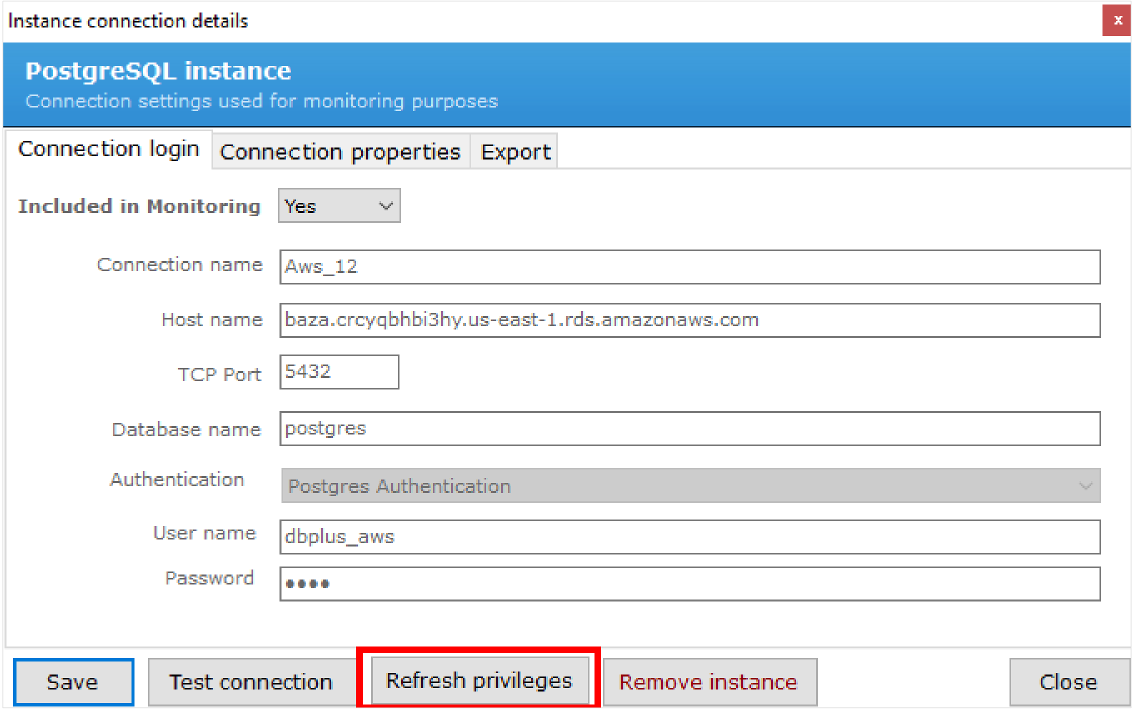
List of DPM components and it's availability and activity

Version 2022.2.2

License Information

Monitored Instances	Monitoring service	Database repository	User application
✓ 6 instances (from 9)	✓ Configured successfully	✓ Configured successfully	✓ Configured successfully
Aws_12/172.31.82.1 pg_14_instancja/34.116.187.218 pg_9.6.22/127.0.0.1 pg10/127.0.0.1 pg12/127.0.0.1	DBPLUS PostgreSQL Catcher Status: ● Running	Hostname: localhost Database: dbplus_1 User: art_last	IIS Service Status: ● Running

Select the **[Refresh privileges]** option:



Instance connection details

PostgreSQL instance
Connection settings used for monitoring purposes

Included in Monitoring: Yes

Connection name: Aws_12

Host name: baza.crcyqbhbi3hy.us-east-1.rds.amazonaws.com

TCP Port: 5432

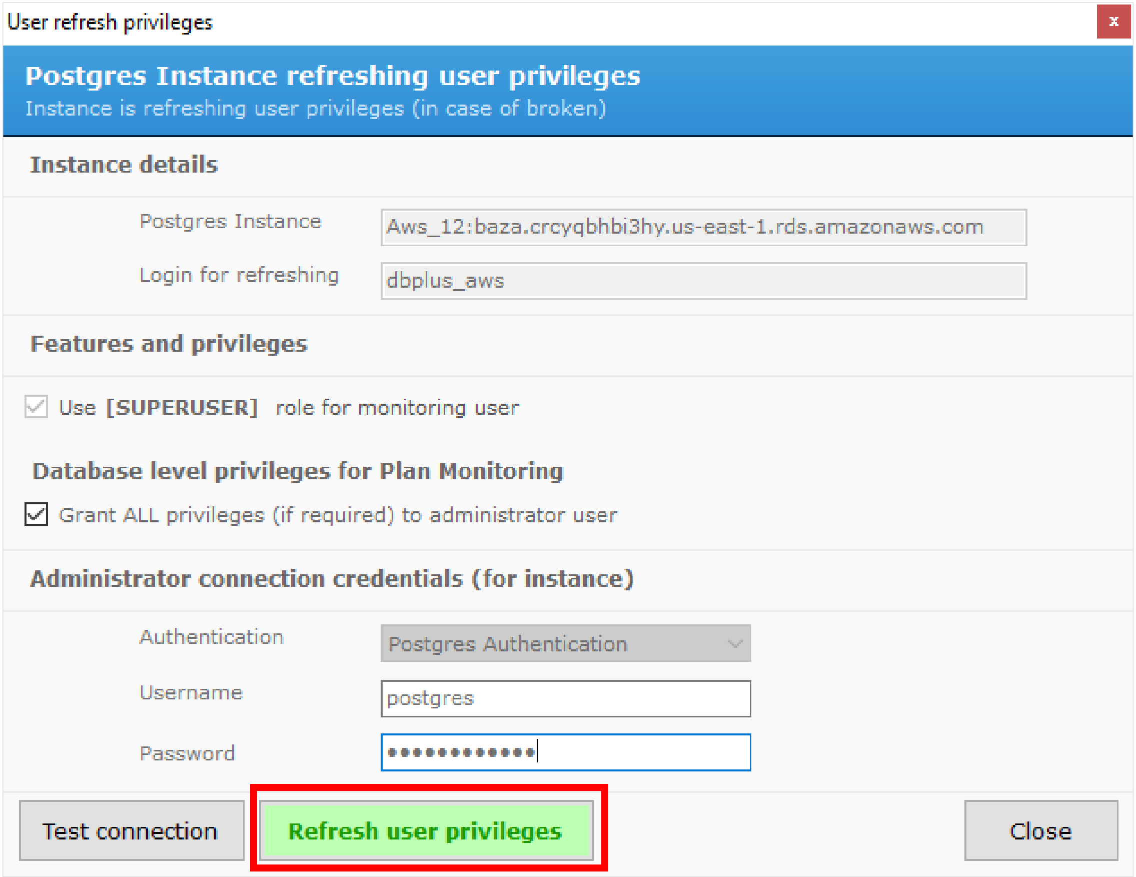
Database name: postgres

Authentication: Postgres Authentication

User name: dbplus_aws

Password: ••••

Complete the login and password of the user with administrator privileges once (user data are not saved anywhere), and then accept the refresh by clicking **[Refresh user privileges]**.



User refresh privileges

Postgres Instance refreshing user privileges
Instance is refreshing user privileges (in case of broken)

Instance details

Postgres Instance: Aws_12:baza.crcyqbhbi3hy.us-east-1.rds.amazonaws.com

Login for refreshing: dbplus_aws

Features and privileges

☑ Use [SUPERUSER] role for monitoring user

Database level privileges for Plan Monitoring

☑ Grant ALL privileges (if required) to administrator user

Administrator connection credentials (for instance)

Authentication: Postgres Authentication

Username: postgres

Password: ••••••••••••

5 Bug fixes and improvements

5.1. Improved SQL Parser

In the latest version we have added a fix pack related to SQL Parser parsing and highlighting of objects in queries. DBPLUS SQL Parser is a functionality that allows the user to analyze queries more easily by selecting the analyzed objects in the plan and the query content.

5.2. Implementation of TLS1.1 and TLS1.2 support

Support for TLS 1.1 and TLS1.2 (Transport Layer Security) has been added to the latest version of the application. The latest version of the application has been coded in .Net 4.7.2.