

DBPLUS  
Performance Monitor for PostgreSQL  
description of changes in version 2020.3

Date: 9 october 2020

**Table of contents:**

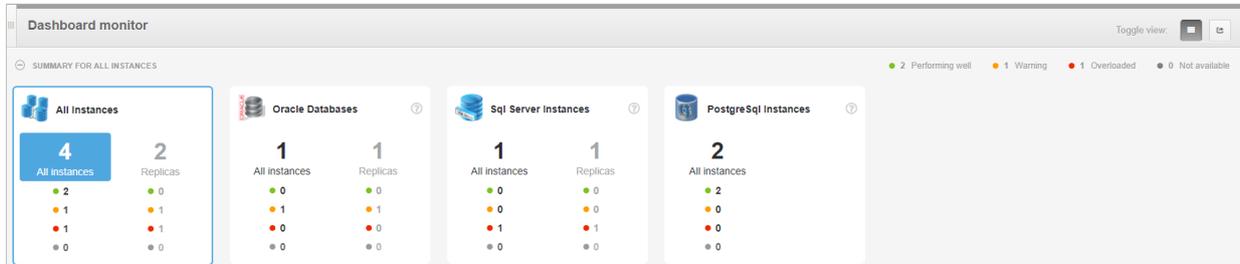
1.1	<i>Shared Dashboard – next edition</i>	3
1.2	<i>Changing the monitoring user privileges</i>	5
1.3	<i>General Improvements</i>	6
1.3.1	<i>Hiding points on the graph</i>	6
1.3.2	<i>SQL FIND search improved</i>	7
1.3.3	<i>Adding a description for waits</i>	7

Below is a list of changes in the DBPLUS Performance Monitor system for monitoring PostgreSQL instances.  
**New in version 2020.3**

### 1.1 Shared Dashboard – next edition

In version 2020.3.1, we improved the Dashboard screen. This screen allows, as before, an online view for all monitored databases for each of the three available platforms (SQL Server, Oracle, PostgreSQL). Information about status of monitored databases is available in the summary in two variants:

- Primary view:



This view allows you to filter the databases by the current SQL PostgreSQL instance status.

- Minimized view



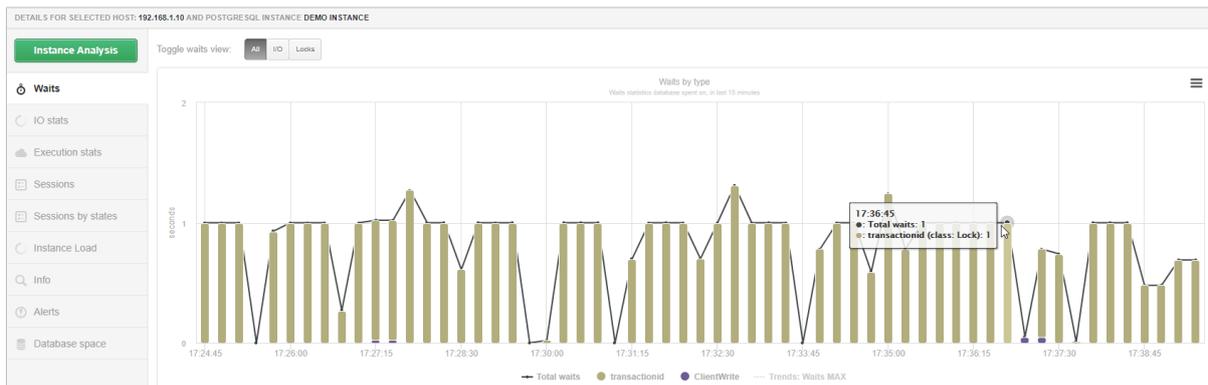
This view is available after clicking the minus icon in the **SUMMARY FOR ALL INSTANCES** bar. It only contains information on the number of databases in a given and group, and allows easy navigation between them.

Additionally, in the latest version in the grid available for each PostgreSQL Instance, apart from the columns presenting basic data about the instance information is also available on:

- CPU Host – not available,
- CPU Instance – not available,
- Waits – total waits [s/1s],
- Waits I/O – I/O related waits [s/1s],
- Locks – level of locks on a given instance [s/1s],
- Active Sessions – number of active sessions,
- Locked Sessions – blocked sessions level,
- Transactions – the number of active transactions in the database,
- Alerts – number of alerts (critical/warning),
- Space info – the level of used space for a given instance [GB] .

Status	Name	Host Name	Version	Startup time	Category	CPU Host [%]	CPU Instance [%]	Waits [s/1s]	Waits I/O [s/1s]	Locks [s/1s]	Active sessions	Locked sessions	Transactions	Alerts	Space Info [GB]
Warning	XE	DESKTOP-HR...	11.2.0.2.0	2020-10-02 14.5...	NOT SPECIFIED	29 %	0 %	0	0	0	1	0	0	0	46 % 12.3
Overloaded	DESKTOP-HR1BE66/SQL_2019	DESKTOP-HR...	2019 (15.0.2070...	2020-10-05 16.0...	NOT SPECIFIED	28 %	0 %	0	0	0	1	0	1	0	25 % 0.4
Performing well	Repository instance	localhost	11.6	2020-10-02 14.5...	NOT SPECIFIED	N/A	N/A	0	0	0	0	0	0	0	100 % 0.1
Performing well	a1	localhost	10.13	2020-10-06 12.1...	NOT SPECIFIED	N/A	N/A	0	0	0	0	0	0	0	0.0

By selecting a specific column for a given PostgreSQL instance, the User receives information on changes in statistics in the form of a graph for the last 15 minutes. In the case below, we get information on top waits for DEMO instances.



The dashboard allows access to other columns that the User can freely add to the dashboard screen after selecting the **Show hidden columns** option in the settings of a given grid (cog icon).

ORACLE DATABASES										Group filter: All selected		Search instance		Hidden columns		Grid options	
Name	Host Name	Version	Startup time	Category	CPU Host [%]	CPU Instance [%]	Waits [s/1s]	Waits I/O [s/1s]	Locks [s/1s]	Active Sessions [% of trend]	Elapsed Time [% of trend]	Waits [% of trend]	Waits statistics	Active sessions [% of trend]	Export grid	Export grid with formatted data	
T1	g4	11.2.0.4.0	2020-08-25 16:22...	TESTING...	62%	38%	1.34	1.31	0	77	49	1					
FK	u4	11.2.0.4.0	2020-09-07 17:52...	TESTING...	46%	44%	2.69	2.28	0	19	0						
T5	u3	11.2.0.4.0	2020-10-01 17:04...	TESTING...	4%	4%	74.16	0.13	49	2	91%	24 267.4					
FK	u4	11.2.0.4.0	2020-08-07 14:37...	TESTING...	33%	8%	1.66	1.01	0	2	91%	5 843.7					
T8	u4	11.2.0.4.0	2020-08-07 19:09...	TESTING...	31%	31%	1.52	0.89	0	27	0						
FK	g4	11.2.0.4.0	2019-06-10 15:35...	TESTING...	22%	6%	1.43	1.13	0	12	0						

### Changing the status of a monitored SQL instance

The change of the instance status is related to exceeding the alarm thresholds for the most important database performance parameters. In the event that the database status changes to **orange** (Warning) or **red** (Critical), the information about the cause is available after pointing the mouse at the given status.

The change in the instance status is related to exceeding the alarm thresholds for the most important performance parameters of PostgreSQL instances:

- Active\_Sessions
- Locked\_Sessions
- Waits I/O
- Waits Locks
- Waits Total

The User can define his own alarm thresholds for each of the main database parameters. This configuration is available in the **Configuration > Settings** menu in the **Dashboard setting** tab. Exceeding a given parameter causes a change of status for a given instance depending on the exceeded threshold. Configuration is available for all or a dedicated PostgreSQL instance.

ALERTS CONFIGURATION					
Name	Alert Enabled	Value for Alert	Warning Enabled	Value for Warning	Description
ACTIVE_SESSIONS	NO		NO		Alert if Active Sessions is equal or greater than value specified in the parameter. Alert is calculated every 15 seconds.
INSTANCE_CPU	YES	80	YES	50	Alert if PostgreSQL instance process utilization is equal or greater than specified in the parameter. Alert is calculated every 15 seconds.
LOCKED_SESSIONS	NO		NO		Alert if Locked Sessions is equal or greater than value specified in the parameter. Alert is calculated every 15 seconds.
SERVER_CPU	YES	80	YES	50	Alert if server cpu utilization is equal or greater than specified in the parameter. Alert is calculated every 15 seconds.
WAITS_IO	YES	80	YES	50	Alert if I/O waits in reference to cpu number is equal or greater than specified in the parameter. Alert is calculated every 15 seconds.
WAITS_LOCKS	YES	80	YES	50	Alert if locking waits in reference to cpu number is equal or greater than specified in the parameter. Alert is calculated every 15 seconds.
WAITS_TOTAL	YES	80	YES	50	Alert if total waits in reference to cpu number is equal or greater than specified in the parameter. Alert is calculated every 15 seconds.

### Alerts on the Dashboard screen

In the latest version, Alerts are displayed on the Dashboard screen. Depending on exceeding the alarm threshold, the user receives information about the criticality of the alert and the number of occurrences. Alerts

are visible on the screen for a period of 2 hours after their occurrence, for example, if the alert occurred at 09:20, the information on the Dashboard about the alert will be visible until 11:20.

ALL INSTANCES														Group filter: All selected		Alerts		
Status	Name	Host Name	Version	Startup time	Category	Repl. status	CPU Host [%]	CPU Instance [%]	Waits [s/r/s]	Waits I/O [s/r/s]	Locks [s/r/s]	Active sessions	Locked sessions	Transactions	Space info [GB]			
●	KAD	u4u	11.2.0.4.0	2020-08-07 19:...	PRODUCTIO...		61 %	12 %	0.01	0.01	0	3	0	0	56 %	113.8	7	1
●	KADT1	e1w	11.2.0.4.0	2020-08-10 13:...	TESTING DAT...		55 %	2 %	0.03	0.01	0	2	0	0	56 %	113.5		

The visibility time of alerts on the Dashboard screen can be configured separately for each platform, using the **DASHBOARD\_ALERTS\_TIME\_MINUTES** parameter available in the **Configuration> Settings** menu.

Settings	Waits settings	Dashboard setting	Dashboard Tv Parameters	Object Size Settings
CURSOR_SNAPSHOT_FREQUENCY	30	The interval time in seconds between each snapshot of FETCH API open cursor statements, made by DBPLUS CATCHER service. The parameter can be setup separately for each instance. In a case of high utilization by FETCH API statements, please consider lower value for CURSOR_SNAPSHOT_FREQUENCY. In a case of rarely executed cursors, please use bigger value for it.		Edit
DASHBOARD_ALERTS_TIME_MINUTES	120	Specify how long alerts will be visible on dashboard. The parameter uses minutes units.		Edit
DASHBOARD_ANIMATE_PARAMETERS	ON	Setting is valid for DPM dashboard displayed in television mode. Based on it each sql server icon can toggle/animate automatically its parameters like (server cpu, sql instance cpu, waits, sessions, etc.)		Edit

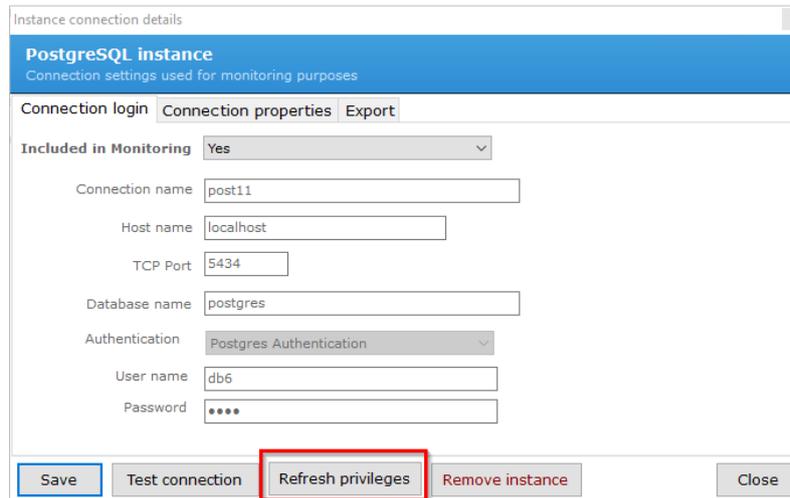
## 1.2 Changing the monitoring user privileges

In the latest version of the application, we have modified the method of collecting queries execution plans. In previous versions, the database user indicated for monitoring had to have high privileges due to the execution of the EXPLAIN command (creating execution plans).

In the latest version, a schema with the same name as the monitoring user name will be created on each monitored database. The explain\_dpm function will be added to the newly created schema. The monitoring user will be authorized to perform functions, which will allow to reduce the privileges in the database.

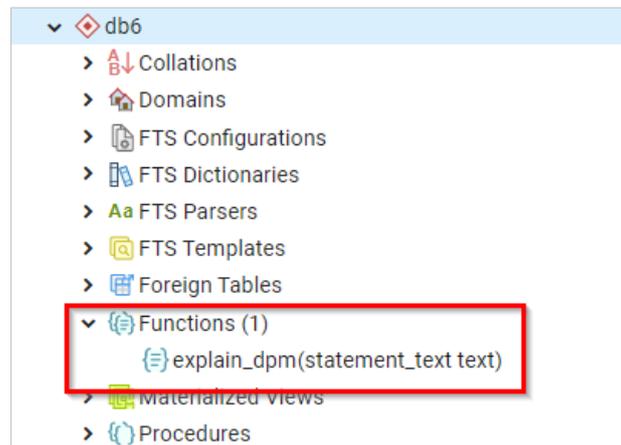
In order to change the current privileges, refresh the monitoring user's rights after updating from the Configuration Wizard. To do this, click the cog icon, as in the example below:

Then click the **[Refresh privileges]** button:



In order to refresh the privileges for the monitoring user (in the example: db6), enter the Administrator's data once and click the **[Refresh privileges]** button. The result will be the creation of a new schema with the same name as the monitoring username (in our case db6) in each database available in the monitored PostgreSQL instance. This schema creates the explain\_dpm function.

Below is an example of a schema named db6:



Thanks to this, the monitoring user no longer executes the EXPLAIN command, but calls the above function to retrieve information about the query execution plan. Repeat the instructions for each PostgreSQL instance available for monitoring.

## 1.3 General Improvements

### 1.3.1 Hiding points on the graph

In the latest version, we have added the ability to hide the selected point on the chart. To hide a given point, hold down the **CTRL** key on the keyboard and at the same time **indicate the point** on the graph that is to be hidden. This functionality will significantly facilitate the performance of the analysis performed over a longer period. Hiding a point in the graph does not change the data available in the grid under the graph. The functionality is intended only to facilitate reading the data available in the chart. Changes have been made to the charts available on the tabs:

- Load trends
- SQL Details
- Bg Writer Stats
- I/O Stats.

### 1.3.2 SQL FIND search improved

In the latest version, we have improved the query search using the SQL FIND functionality, available in the SQL Details tab. Saving previously searched queries / objects has been improved. Capitalization does not matter.

### 1.3.3 *Adding a description for waits*

In the latest version of the application we have added additional information about waits occurring in a given PostgreSQL instance. At the level of details of a given instance, after entering the Waits tab, a graph of wait occurrence for a given instance is presented. Below the graph, information on the level of each wait in a given period of time is presented. Currently, it will also be possible to read additional information (description) what the wait is related to.