

DBPLUS Performance Monitor dla PostgreSQL Description of changes in version 2019.3

Date: August 30, 2019



Table of Contents

1	Nev	w in version 2019.3	3
	1.1	New version of the SQL 3D	3
	1.2	Comparison of the wait level	3
	1.3	A new view shows the process of data archiving	4
	1.4	Preview of the replication process	5
	1.5	Table stats	5
	1.6	General improvements	8
	1.6.	1 Improving the ergonomics of displaying data in the application	8
	1.6.	2 Summary of statistics	8
	1.6.	3 Quick configuration of dates from the calendar	9
	1.6.	4 Addition of statistics presentation broken down into databases	10



Below is a list of changes in the DBPLUS Performance Monitor system for monitoring Microsoft PostgreSQL instances.

1 New in version 2019.3

1.1 New version of the SQL 3D

In the latest version of the browser, the SQL 3D view has been modified. A new library has been used, making the chart easier to navigate and faster in presenting and reading data contained in the chart. The graph can be rotated vertically and horizontally (by holding the left button of the computer mouse), zooming in and out. At the same time, the previous functionality was saved, where after selecting the bar the tooltip will be shown, which includes:

- Query ID,
- Date details,
- Statistics value,
- Query text.

By unchecking the checkbox next to the queries (above the graph), the user can easily remove the query from the graph.



1.2 Comparison of the wait level

In the new version of the application, the function of compare the wait level in the indicated period has been added. Compare is possible from the instance details in the Waits tab. There are two comparison modes:

- Days Compare
- Period Compare

In order to compare the validity level, first select the type of validity to be compared from the list (one or more types), then select individual days for comparison (Days compare tab) or whole day ranges (Period Compare tab).



1.3 A new view shows the process of data archiving

In the new version of the application, a view showing the information contained in the system view pg_stat_archiver has been added. The application presents information about the number of correctly archived WAL files and information about the number of polices where the archiving failed.



Data are available in the division for a given day, after click the indicated row in the table below the chart, detailed information grouped after a snap for a given day will be presented:

The data is available in the IO Stats menu in the Archived WALs tab:

DBPLUS



1.4 Preview of the replication process

In the latest version of the application, the system view support pg_stat_replication has been added. The data is available in the IO Stats menu in the Replication tab.

After entering the tab, first select one in the replicating processes from the list available in the "WAL sender process" field. Note: in the absence of a replicating process in the dropdown field, the message *No process found* will be displayed.

After select the process, information that contain details of the given process will be displayed:

- Logdate date of data collection
- PID sender process ID
- Usename name of the user logged in to this sender process
- Application name the name of the application connected to this sender
- Client IP Client IP connected to this sender
- Backend start Process start time (time stamp)
- State the status of the WAL sender
- Write lag [sec]
- Flush lag [sec]
- Replay lag [sec]
- Sync priority the priority of choosing a standby server
- Sync state synchronous state of the standby server
- Pending lag [Bytes]
- Write lag [Bytes]
- Flush lag [Bytes]
- Replay lag [Bytes]
- Total lag [Bytes]

1.5 Table stats

In the latest version of the application, the ability to view statistics for tables in PostgreSQL databases has been added. To this end, a new menu item has been added at the level of instance details> Table Stats. Notice!! If the application is running in Safe mode (the SECURITY parameter is ON), access to the newly created page should be granted (access is denied by default).

After entering the user page, it is possible to view statistics broken down into all databases or a specific database [Filter tables by database]. After select the database, a graph will be drawn for the default statistics, which can be changed at any time by select a different statistic from the [Show] field and press the [Refresh] button.





After select the statistics, it is possible to verify the percentage share of a given table compared to other tables. To do this, indicate tables (or many tables at the same time) from the list below the graph, and their total value for a given statistic will be presented on the chart in the form of yellow columns. 20 top tables are visible below the chart, it is possible to change the list to present all tables by changing the [Table filter] field.

For detailed analysis, it is possible to add a table for analysis by click the [+] button next to the table name [Table name] in the grid below the chart. The mechanism works in the same way as when choosing query identifiers. Two options to choose:

- [Add to table identifiers list] information about the table is added to the clipboard
- [View table details] go directly to the table details.

If the option to add to clipboard is selected, table identifiers are visible after pressing the green arrow on the left side of the screen. After pressing the button and selecting a table from the list from the clipboard, user will go to the details screen> Table Stats Details.

On the details screen it is possible to analyze detailed statistics for a given table. To analyze the table, first select the database from dropdown and then fill in the schema data and table name manually. If you select a table from the clipboard, after select the table name in the clipboard, the data will be substituted automatically.



To verify a given statistics, indicate the column in the table – column will be marked and data will be transferred to the chart. It is possible to select multiple statistics at once for a single table. The site can view statistics for tables grouped by:

- snap (15 minutes),
- an hour
- a day



- a month
- no group (No group by period)

According to the current function, the graph and data in the table can be easily exported to a file.

Collecting information about table statistics requires manual change of settings for each monitored PostgreSQL instance (disabled by default) from the settings level in the main menu (Configuration> Settings). First, User need to select the instance in which You want to change the settings, and then for the MONITOR_TABLE_STATS parameter through the [Edit] button we indicate for which databases we want to enable monitoring table statistics. The changes will be visible after another 15 minutes (snap).

Configuration	KEEP_SNAPSHOT_HISTORY_DAYS	30		Number of days how long to keep detail statistics for sql statement executions, waits, latches, performance counters.	Edit								
Soungs Instances References lists	LOCKING_SNAPSHOT_FREQUENCY	60		The interval time in seconds between each snapshol of locks made by DBPLUSPOSTGRESCATCHER service. The parameter can be setup separately for each instance. In a case of thequent locks, seare consider lower value for LOCKING_SNAPSHOT_FREQUENCY. In a case of rarely occured locks, please use bigger value for it.									
Security Alert settings Scheduled outages	LOGGING_MODE	OFF		Parameter used for debugging mode. By default it should be set to OFF.	Edit								
Scheduled works	MONITOR_EXPLAIN_PLANS	ON		Parameter which switch ONOFF the module to estimate explain plans for most heavy statements run on the instance.	Edit								
Version: 2019.3.1	PLANS_TO_GENERATE_PER_SNAP	20		Number of most heavy queries for which system will estimate explain plans - Estimation is done in every snapshot.	Edit								
	SECURITY	OFF		Application can work in SECURITY mode set to ON or to OFF. It means that application uses (or doesn't use) user authentication. Setting the SECURITY to on, it requires at least one user created.									
	List of parameters specific for pa Those params which are marked	rticular s I in light (ql server instance. Below settings ov gray color, are inherited from main co	ervrite Main Application Parameters table. Infiguration	×								
	INSTANCE PARAMETERS - PLEASE SELEC	T AN INST	ANCE Repository instance +										
	Parameter	Value	Description		×								
	LOCKING_SNAPSHOT_FREQUENCY	60	The interval time in seconds between ea LOCKING_SNAPSHOT_FREQUENCY.	ch enaphol of locks made by OBPLUSPOSTORESCATCHER service. The parameter can be setup separately for each instance. In a case of frequent locks, please consider lower value for In a case of rarely occured locks, please use logger values for it.	Edit								
	MONITOR_TABLE_STATS ON Parameter containing list of databases with monitored table statistics.												

We repeat the changes for each monitored instance.

Selection of databases in PostgreSQL instance for which table statistics will be collect.

SELECT DATABASES													
Databases		Monitored databases											
Search by name		a3 doplus rap											
template1 a1 a2	0	postgres											
]		ОК	Cancel									



1.6 General improvements

1.6.1 Improving the ergonomics of displaying data in the application

In the latest version of the application, we have improved the data presentation on individual pages. The correction consisted in the modification of pages where empty sections were visible and could not be minimized or moved.

1.6.2 Summary of statistics

The latest version adds a summary of the data presented in the tables. Summaries are available for the most important screens in the application:

• Session (online sessions) – information about the number of active sessions is available

III Ses	ions	Session with transactions	Sessions history	Active Sess	ion histor	у									
🗹 Activ	Alt databases - Users only Min elapsed time: 0 + Usersame: Username: Username: Refrest														
	Show additional filters														
SELECT	SELECT SESSION (LAST REFRESHED: 13.06.00) Kill session														
Logon tir	Logon time Pid Transaction start Query start Query istart Picture P									Statement					
2019-07-3	1 1 17	52 2019-07-31 13:05:57	2019-07-31 13:05:57		dbplu	postg	eactive	3		pgAdmin			do \$do\$ declare i int; begin for i in 11000000 loop insert into test2 values (random(), i * random()); end loop; end; \$do\$;		
2019-07-3	19-07-31 1 4616 2019-07-31 13.06.00 2019-07-31 13.06.00 2019-07-31 13.06.00 dtptu. dtptus eactive 0 DBPLUS														
Count se	ssion 2												•		
501															
Just															
STATEN	IENT TEX														
select	* from	pg_stat_activity ps	a where 1=1 and ps	sa.state = \$1	L and (p	sa.dati	ld is not n	ull and	psa.use	ename is no:	t null)				
EXPLAI	N PLAN														
Sh	ow pla:	objects													
	sted Lo	op Left Join (cost=1.	263.44 rows=1 width	=440)											
	Join .	Filter: (s.datid =	d.oid)												
	-	lash Join (cost=1.26	2.40 rows=1 width=276												
		-Hash Cond: (u.o	<pre>id = s.usesysid) pg authid u (seese</pre>	0.00 1.09											
		-Filter	: (rolname IS NOT)	NULL)	widta										

 SQL Details (query details) – the information is grouped into total (Total), medium (Avg), minimum (Min), maximum (Max) values.

To see summary of statistics, select the new Show Summary checkbox.

III Instance	Load Waits	SQL Analyze	QL Details Load	Trends Compare	Top SQL SQL 3D	Top Day Slow S	QLs								
78958181	7895818160065593477 From: 2019/07/24 00:00 for 2019/07/21 22:59 2-3 Coup by plan														
STATEMEN	STATEMENT TEXT														
select \$ sum(numb) sum(xact sum(xact sum(blks)	<pre>lelect #1 as datid, #2 as datname, yum(numbackends) as mubackends, yum(xact_comit) as xact_comit, yum(xact_cliback) as xact_collback, uum(biks_read) as biks_read,</pre>														
SQL STATI	SQL STATISTICS 🔲 Show values per 1 executions 🐼 Show summary														
Date	Plan Id	Elapsed Time	Blks read time	Blks write time	Executions	Rows	Blks hit	Blks read	Blks dirtied	Blks written	Buffer Quality	Temp blks read	Temp blks write	Elapsed Time per 1 Exec	
		[Seconds]	[Seconds]	[Seconds]		[Rows]	[Blocks]	[Blocks]	[Blocks]	[Blocks]	[%]	[Blocks]	[Blocks]	[Seconds]	
2019-07-30	3873484607	23.1	0	0	1 196	1 196	1 196	0	0	0	100.0	0	0	0.0193 🔺	
2019-07-31	3873484607	22.4	0	0	958	958	958	0	0	0	100.0	0	0	0.0234	
2019-07-26	3873484607	48.3	0	0	1 974	1 974	1 974	0	0	0	100.0	0	0	0.0245	
2019-07-29	3873484607	47.7	0	0	1 795	1 795	1 795	0	0	0	100.0	0	0	0.0266	
2019-07-24	3873484607	30.5	0	0	1 137	1 137	1 137	0	0	0	100.0	0	0	0.0268	
2019-07-25	3873484607	44.2	0	0	1 496	1 496	1 496	0	0	0	100.0	0	0	0.0296 🔻	
Total	-	216.2	0	0	8 556	8 556	8 556	0	0	0	-	0	0	-	
Avg	-	36.0	0	0	1 426	1 426	1 426	0	0	0	100.0	0	0	0.0250	
Max		48.3	0	0	1 974	1 974	1 974	0	0	0	100.0	0	0	0.0296	
Explain p	an Graph		<u></u>								<u></u>	<u></u>		3873484607 👻	
Data	plan obje base: dbplu egate (cosp= Seg Scan or	ects for 387348 s_rep 1.201.24 rows=1 wi a pg_database d (c	84607 doh=512) cost=0.001.02 rows	=2 width=4)											

 Load Trends – the information is grouped into total (Total), medium (Avg), minimum (Min), maximum (Max) values.

DBPLUS better performance

III Instance	Load Wai	ts SQL Ar	alyze SQ	L Details	Load Trends	Compar	re Top S	QL SQL 3	D Top Da	y Slow SC	QLs																
Date from:	2019/0	17/02 to	201	9/07/29															SI	now statistic:	s for: Al	l databases		Group	by Day 👻	Refres	h
POSTGRE	SQL TRENDS	LOAD																							Chart type:	Line	-
(PostgreS	QL trend stal	istics												=	
Time											\checkmark		-														
2019-07-42 2019-07-45 2019-07-10 2019-07-16 2019-07-19 2019-07-24 2019-07-24													9-07-29														
	fine Finsed Time																										
	- Elapsed Time																										
POSTGRESQL TRENDS STATISTICS Clear selection														ction													
Logdate •	Elapsed Time	Executions	Rows	Blks hit	Blks read	Blks dirtied	Blks written	Temp blks read	Temp blks written	Wait time	IO time	Lock time	Active	Sessions	Connecti	Commits	Rollbacks	Tuples returned	Tuples fetched	Tuples inserted	Tuples updated	Tuples deleted	Conflicts	No of temp files	Data writen to temp	Deadlocks	BI
	[Seconds]		[Rows]	[Blocks]	[Blocks]	[Blocks]	[Blocks]	[Blocks]	[Blocks]	[Seconds]	[Seconds]	[Seconds]													[MB]		[\$
2019-07-02	41	24 530	20 315	79 591	38	1 648	0	0	0	0	(0	()	8 186	25 845	163	3 056 102	196 879	8 245	4 225	5 841	0	0	. 0	0	4
2019-07-03	41	30 427	21 630	86 591	37	1 799	0	0	0	0	0	0	()	8 198	27 787	174	3 338 527	208 864	8 801	4 502	6 221	0	0	. 0	0	1
2019-07-04	60	33 766	27 164	107 396	50	2 192	0	0	0	0	0	0	() ;	8 246	34 696	217	4 217 521	261 703	10 968	5 561	7 722	0	0	0	0	1
2019-07-05	73	36 072	138 088	430 584	69	2 339	0	0	0	0	0	0	()	9 281	36 802	226	4 542 793	277 776	11 562	6 050	8 098	0	0	0	. 0	1
2019-07-08	57	26 222	36 174	157 835	308	7 300	0	0	0	0	(0	()	9 211	26 721	163	3 301 211	211 767	8 628	4 469	5 788	0	0	0	. 0	1
2019-07-09	80	55 832	65 952	245 077	235	6 352	0	0	0	0	(0	()	8 354	50 497	310	6 321 085	393 249	16 090	8 487	11 123	0	0	0	0	1
2019-07-10	94	59 550	72 659	293 873	314	7 982	0	0	0	0	(0	()	8 414	59 140	357	7 637 790	461 507	18 837	9 839	12 770	0	0	0	0	-
2019-07-11	45	29 130	83 304	267 344	437	3 318	0	0	0			0		,	8 198	28 126	1/4	3 655 238	222 243	9 104	4 /24	6 219	0	0	0	0	-
2019-07-12	20	10 203	203 211	124 509	340	2 000	0	0	0						0 1/4	47.050	140	3 244 400	142 425	6 000	4 205	3 702	0	0	0	0	-
2019-07-17	71	32 631	49 820	159 685	114	3 661	0	0	0						8 210	30.006	185	4 011 290	244 291	9 737	5 205	6 595	0	0		0	
4		02 001	10 020	100 000						1 ~	1	1 *			0 210	00000	100	1011200	LITEOT	0101	0 200					•	
Total	1 117	643 529	1 140 950	3 514 605	4 270	89 148	0	0	0	0	0	0	-		- 3 995	559 435	4 264	73 531 1	5 803 81	181 665	95 617	130 740	0	0	0	0	
Avg	59	33 870	60 050	184 979	225	4 692	0	0	0	0	0	0	0		3 210	29 444	224	3 870 060	305 464	9 561	5 032	6 881	0	0	0	0	
Max	94	70 263	203 211	430 584	630	11 203	0	0	0	0	0	0	0		9 414	59 140	357	7 637 790	990 616	18 837	9 839	12 770	0	0	0	0	

In case when a single row is presented in the table, the summary will not be presented.

1.6.3 Quick configuration of dates from the calendar

In the new version, we've added a new improvement in the form of speed dialing on most major screens. After click the button, user can choose from several defined options:

- Today
- Yesterday
- Last Week
- Last 2 Weeks
- Last 30 days
- This Month the scope of the current month from 1 to the last day of the month,
- Prev Month the range of the entire previous month from 1 to the last day of the month,
- Custom Range selection of date range manually.

After select any range from the list, click the [Refresh] button to refresh the page. An example screen below:



To select the **Custom Range** date range, first select the start date, then indicate the end date. The selection is accepted via the [**Apply**] button.

DBPLUS

III Database Load Waits Latches SQL Analyze SQL	Details SQL Plan	Load Trend	ds (Compare	e	Top SQI	. so	2L 3D	Top E)ay	Slow	SQLs	Perf	f Counte	s (OS Stat
Date from: 2019/03/02 to: 2019/03/31																
ORACLE DATABASE LOAD	Today	<	K Mar 2019 Apr 2019										9 >			
Q Click on the chart at specified snapshot time to view	Yesterday	Мо	Tu	We	Th	Fr	Sa S	Su	Мо	Tu	We	Th	Fr	Sa	Su	
	Last Week	25	26	27	28	1	2	3	25	26	27	28	29	30	31	
	Last 2 Weeks	4	5	6	7	8	9	10	1	2	3	4	5	6	7	
125	Leet 20 Dave	11	12	13	14	15	16	17	8	9	10	11	12	13	14	
	Last 30 Days	18	19	20	21	22	23 2	24	15	16	17	18	19	20	21	
100	This Month	25	26	27	28	29	30 3	31	22	23	24	25	26	27	28	
	Prev Month	1	2	3	4	5	6	7	29	30	1	2	3	4	5	
तु 75	Custom Range															
50 50				,				201	19/03/0	6 - 20	19/03/1	5 (Cancel	Ар	ply	

1.6.4 Addition of statistics presentation broken down into databases

In the new version, the **option** related to browsing queries broken down by a specific database in the instance has been added. After select the database that interest us, the data is available for the selected database, in relation to all databases for PostgreSQL instances. This option has been added on the following pages:

- Top SQL,
- Instance Load,



Load trends,

