

DBPLUS Performance Monitor **opis zmian w wersji 2018.1.1**

Data: 15 kwiecień 2018

Spis treści

1	Nowości w wersji 2018.1.1	3
1.1	Plany wykonań	3
1.1.1	Generowanie skryptu PLAN GUIDE	3
1.1.2	Pokazywanie informacji o brakujących indeksach	5
1.1.3	Usprawnienia w ekranie Show Plan Object	6
1.2	Widok sesji	7
1.2.3	Dodanie kolumny hostname	7
1.2.4	Wiele oczekiwań dla pojedynczej sesji (zadania sesji)	7
1.3	Ekran historii aktywnych sesji i sesji używających log-a	8
1.3.1	Dodanie kolumn hostname, context_info	8
1.3.2	Pokazywanie wielu oczekiwań dla pojedynczych sesji	8
1.3.3	Dodanie zbiorczych wskaźników dla monitorowanych sesji	8
1.4	Usprawnienia w opcji SQL Find	8
1.4.1	Wyświetlanie dodatkowych statystyk dla wyszukanych zapytań	9
1.4.2	Usprawnienie w analizie zapytań, które zmieniają plan wykonania	10
1.5	Aktualizacja ekranu Application architecture	11
1.6	Zmiany w usłudze monitoringu DBPLUSCATCHER	12
1.6.1	Dodanie procesu przebudowy tabel używanych w ekranie Dashboard	12
1.6.2	Błąd w monitoringu baz danych w edycji SQL Azure	12
1.6.3	Aktualizacja słownika waitów NIE-performance-owych - CXPACKET	12
1.7	Usprawnienia ogólne	12
1.7.1	Zmiana rozmiaru/wysokości wykresów	12
1.7.2	Wykrywanie problemu z nie działającym monitoringiem	13
1.7.3	Niepoprawne sortowanie w ekranie Slow SQL	13

Poniżej prezentujemy wykaz zmian w systemie DBPLUS Performance Monitor do monitoringu baz danych MSSQL.

1 Nowości w wersji 2018.1.1

1.1 Plany wykonań

W funkcjach umożliwiającym podgląd planu wykonania min.:

- Sessions
- Performance->Instance Load -> wybrany snapshot na wykresie
- Performance->Sql Analyze
- Performance->Sql Details
- Performance->Slow SQL

zostały wykonane poniższe usprawnienia:

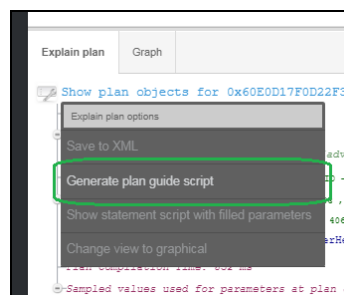
1.1.1 Generowanie skryptu PLAN GUIDE

Na poziomie ustawień planu wykonania pojawiła się opcja z możliwością wygenerowania skryptu na podłożeniu planu wykonania.

Funkcjonalność jest przydatna przede wszystkim w sytuacjach gdy silnik bazy danych zmienia plan wykonania na gorszy. Bardzo często ze zmianą planu, wiąże się wzrost czasu wykonania i wolno działające zapytanie może być źródłem innych problemów wydajnościowych (np. wydłużenie transakcji, wzrost blokad, wzrost odczytów z urządzeń dyskowych).

Podkładając plan mechanizmem PLAN GUIDE-a, ustawiamy na stałe określony plan wykonania z jakim ma działać zapytanie.

Po kliknięciu w ikonę ustawień, wyświetla się okno z opcjami:



Po wybraniu **Generate plan guide scripts**, pokazuje się formularz z opcjami wyboru, od których zależy dalsza postać skryptu.

W formularzu użytkownik ma dostępne następujące informacje, min o:

- Tekście zapytania
- Formie uruchamianie zapytania, do wyboru mamy:
 - Z poziomu aplikacji
 - Z poziomu obiektu bazy (np. procedura, funkcja, trigger)
 - Z bloku T-SQL
- Nazwie bazy danych
- Liście parametrów lub nazwie obiektu z którego zapytanie jest uruchamiane
- Liczbie identyfikatorów zapytania – jeżeli jest więcej niż 1 może to oznaczać, że zapytanie jest z literałem lub jest uruchamiane w kilku bazach danych.

Przed wygenerowaniem skryptu, użytkownik może:

- Określić nazwę planu wykonania

- Zmienić nazwę bazy danych (np. opcja może być przydatna, gdy identyczne zapytania jest uruchamiane w kilku bazach danych)
- Wybrać wskazówki tj. hinty dla skryptu z „*plan guidem*”. Opcja pozwala określić:
 - Własne wskazówki dla indeksów, operatorów złączeń, kolejności operacji, itp
 - Wskazówki wg planu, na który został uruchomiony formularz
 - Wskazówki z parametryzacją planu (jeżeli zapytanie jest literałem)

Opcja **Plan guide HINTS: Use hints from selected plan hash**, oznacza, że skrypt z podłożeniem planu użyje wybranego planu wykonania.

Przy opcji **Plan guide HINTS: Use own hints** mamy możliwość podania własnych wskazówek dla planu – poniżej przykład:

Opcję **Use Parametrization Forced hint**, używamy w sytuacji gdy mamy do czynienia z literałami. Np. dla wybranego identyfikatora QueryHash, mamy wiele SQL handli w widoku systemowym ze statystyką zapytań.

Przykładowa postać zapytań, gdzie warto zastosować parametryzację, to:

```
select * from pracownicy where nr=1
select * from pracownicy where nr=2
select * from pracownicy where nr=3
select * from pracownicy where nr=4
```

Poniżej screenshot z przykładowym skryptem na podłożenie planu wykonania:

```

/* Please execute following plan guide script on instance [SQL12-12\NAV_NN] with sysadmin/dbowner rights*/
use master;
declare @query_hash varbinary(8);
declare @query_text nvarchar(max);
declare @query_text_template nvarchar(max);
declare @plan_name nvarchar(max);
declare @plan_name_old nvarchar(max);

declare @plan_type nvarchar(20);
declare @plan_module nvarchar(128);
declare @plan_params nvarchar(max);
declare @plan_hints nvarchar(max);

set @query_hash = 0x52E706C0113715EA;
set @plan_name = N'DBPLUS_0x52E706C0113715EA';
set @plan_type = N'SQL';
set @plan_module = null;
set @plan_params = N'@P1 varchar(20),@P2 int,@P3 int,@P4 varchar(10),@P5 int,@P6 varchar(20)';
set @plan_name_old = null;
set @plan_hints = N'OPTION(TABLE HINT("Navision NN"."dbo"."NNNNNN NN$ANHH Document Header", INDEX(I_225DATEI

select top 1 @query_text =
    (SUBSTRING(qt.text, qs.statement_start_offset / 2 + 1,
        (CASE WHEN qs.statement_end_offset = -1
            THEN LEN(CONVERT(nvarchar(max), qt.text)) * 2
            ELSE qs.statement_end_offset end -
            qs.statement_start_offset
        ) / 2 + 1
    ))
from sys.dm_exec_query_stats qs
CROSS APPLY sys.dm_exec_sql_text(qs.sql_handle) AS qt
where qs.query_hash = @query_hash;

if (@query_text collate database_default is null)
begin

```

W następnym kroku skrypt należy przekopiować do programu SQL Studio, i wykonać na użytkowniku z uprawnieniami administracyjnymi.

Przydatne informacje:

- Opcja z **Generowaniem skryptu na podłożenie planu wykonania** jest dostępna od wersji SQL2008 i nie działa w edycjach EXPRESS, AZURE
- W niektórych przypadkach skrypt na podłożenie planu wykonania może się nie udać. Dotyczy to min. poniższych sytuacji:
 - **Optymalizator baz danych SQL NIE wspiera wszystkich rodzajów komend SQL** (np. dla konstrukcji `update pracownicy set nazwisko=@X where nr_pesel=@y`, podłożenie planu NIE jest możliwe, natomiast dla konstrukcji postaci `update p set p.nazwisko=@X from pracownicy p where p.nr_pesel=@y` już podłożenie planu jest możliwe
 - **Optymalizator baz danych SQL NIE wspiera przypadku gdy zapytanie jest uruchamiane w bloku T-SQL** – podłożenie planu jest możliwe, gdy zapytanie jest uruchamiane z aplikacji lub obiektu bazy danych
- **Należy zachować szczególną ostrożność przy podkładaniu planów wykonania, min. wybierać właściwy plan wykonania lub właściwe wskazówki dla planu**
- Po podłożeniu planu wykonania, można sprawdzić jego istnienie w widoku systemowym `sys.plan_guides`.

1.1.2 Pokazywanie informacji o brakujących indeksach

Na poziomie planu wykonania została dodany nowy węzeł (gałąź) pokazujący informacje o brakującym indeksie lub indeksach. Opcja ta jest dostępna w planie wykonania wyświetlanym w trybie Compact.

Brakujące indeksy są wyznaczone przez optymalizator bazy danych na podstawie statystyk indeksów.

Poniżej przykłady:

```

Show plan objects for 0x60E0D17F0D22F3B1
--Database: adv_works
--Missing indexes
/*Missing index impact: 93.77% use [adv_works]; create index [missing_index_SalesOrderHeaderEnlarged_04232018] on [Sales].[SalesOrderHeaderEnlarged] ( [RevisionNumber],[Status],[SalesOrderID] ) include ( [O
--SELECT ( Cost = 24,7761 , Rows = 0 , CPU = 0 , IO = 0 )
--Compute Scalar ( Cost = 24,7761 , Rows = 406392 , CPU = 0,0406392 , IO = 0 )
--Compute Scalar ( Cost = 23,6002 , Rows = 406392 , CPU = 0,129007 , IO = 0 )
--Clustered Index Seek ([SalesOrderHeaderEnlarged].[PK_SalesOrderHeaderEnlarged_SalesOrderID] [h]) ( Cost = 23,4712 , Rows = 406392 , CPU = 1,41923 , IO = 22,052 )
--Plan Compilation Time: 832 ms
--Sampled values used for parameters at plan compilation time
--@h: 100
--@r: 3
--@s: 3
    
```

```

Show plan objects for 0xF02EB8B03976C8D9
--Database: adv_works
--Missing indexes
/*Missing index impact: 97.4518% use [adv_works]; create index [missing_index_TransactionHistory_04232018] on [Production].[TransactionHistory] ( [ProductID] ) include ( [TransactionID],[ReferenceOrderID],[
--SELECT ( Cost = 204,969 , Rows = 0 , CPU = 0 , IO = 0 )
--Top ( Cost = 204,969 , Rows = 1000 , CPU = 0,0001 , IO = 0 )
--Sort ( Cost = 204,969 , Rows = 1000 , CPU = 5,03024 , IO = 0,0112613 )
--Inner Join-Nested Loops ( Cost = 199,928 , Rows = 69244 , CPU = 0,28526 , IO = 0 )
--Index Seek ([TransactionHistory].[IX_TransactionHistory_ProductID] [t]) ( Cost = 0,165074 , Rows = 69244 , CPU = 0,0752254 , IO = 0,0895459 )
--Clustered Index Seek ([TransactionHistory].[PK_TransactionHistory_TransactionID] [t]) ( Cost = 199,978 , Rows = 1 , CPU = 0,0001991 , IO = 0,009123 )
--Plan Compilation Time: 11 ms
--Sampled values used for parameters at plan compilation time
--@r: 1000
--@p: 1
    
```

Należy zachować ostrożność przy zakładaniu indeksów proponowanych przez silnik bazy danych.

1.1.3 Usprawnienia w ekranie Show Plan Object

W oknie **Show Plan Object** pozwalającym na analizę planu wykonania i obiektów uczestniczących w planie pojawiły się dodatkowe informacje dotyczące kolumn dla wybranej tabeli lub indeksu:

- Is identity – czy dana kolumna ma atrybut identyfikatora/sekwencji
- Is nullable – czy kolumna przyjmuje wartości null

Poniżej przykład:

Object columns	DDL info	Properties	Details for table [Production].[TransactionHistory]							
Column	Type	Max Length	Position	Is identity	Is computed	Is nullable	Is sparse	Density	Unique values	Rows sampled
StatusID	int	4	10	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	0.02941176	34	34 062 112
TransactionID	int	4	1	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0.00000000	335 924	335 924
ProductID	int	4	2	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0.00000775	129 024	34 062 112
ReferenceOrderID	int	4	3	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0.00002750	36 363	345 060
ReferenceOrderLineID	int	4	4	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>			
TransactionDate	datetime	8	5	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	0.00000690	144 955	34 062 112

1.2 Widok sesji

W ekranie Sesji pojawiło się kilka niżej opisanych usprawnień.

1.2.3 Dodanie kolumny hostname

W widoku sesji widzimy dodatkową informację o nazwie hosta z którego jest uruchomiona sesja:

Logon time	Sessid Id	Query Hash	Login name	Status	Last request start time	Elapsed Time [seconds]	Windows username	Host name	Program	Context Info	Blocking session	Database	Wait
2018-04-23 10:52:16	54	0x78C283194A2E	dbplusweb	running	2018-04-23 13:51	0		WIN-PVMD4LCT8A	DBPLUS Perform		0	master	
2018-04-23 11:41:43	63	0xDF1D6547		running	2018-04-23 13:51	0		WIN-PVMD4LCT8A	Net SqlClient Dat		0	adv_works	

1.2.4 Wiele oczekiwań dla pojedynczej sesji (zadania sesji)

W widoku sesji, jedną z kolumn tabeli jest oczekiwanie. W przypadku gdy sesja działa wielowątkowo lub wielozadaniowo, każde zadanie może działać z różnym oczekiwaniem np. jeden wątek może być w blokadzie a inny czekać na odpowiedź z macierzy dyskowej.

W nowej wersji dodano logikę obsługującą tego typu przypadki. W poniższym przykładzie widać jedną sesję, która działa z dwoma oczekiwaniami:

Logon time	Sessid Id	Query Hash	Login name	Status	Last request start time	Elapsed Time [seconds]	Windows username	Host name	Program	Cont Info	Blockin session	Database	Wait	Wait time [seconds]
2018-04-23 13:59:14	55	0x8BD98CF9D17	MAQCHirados	running	2018-04-23 13:59	25	radoslaw	MAQCH	Microsoft SQL	0	0	dbplus_tester	LCK_M_S_CXPACKET	25.49
2018-04-23 14:00:03	62	0xDF543DE9FA0	dbgas	running	2018-04-23 14:00	0		MAQCH	DBPLUS Perf	0	0	DBPLUS		0.00

Po kliknięciu w rekord wybranej sesji widać szczegóły – została tutaj dodana nowa zakładka **Waiting tasks**, w której pokazywane są poszczególne zadania/wątki dla sesji.

Poniżej przykład dla sesji działającej w trybie wielowątkowym:

Exec Context	Wait	Wait Time [seconds]	Blocking Session id	Blocking Exec Context	Resource Description
0	CXPACKET	47.000	55	1	exchangeEvent id=Pipe90162680 WaitType=e_waitPipeGetRow nodeId=0
0	CXPACKET	47.000	55	2	exchangeEvent id=Pipe90162680 WaitType=e_waitPipeGetRow nodeId=0
2	LCK_M_S	47.000	58		keylock hobtid=72057594186301440 dbid=16 id=lock8a946300 mode=X associatedObjectId=7205759418630
1	LCK_M_S	47.000	58		keylock hobtid=72057594186301440 dbid=16 id=lock800b1900 mode=X associatedObjectId=7205759418630

Dla wątków głównych mamy pozycje działające z oczekiwaniem CXPACKET i wartością 0 dla kolumny Exec Context. Kolejne wątki otrzymały identyfikatory większe od zera -> w prezentowanym przykładzie oba wątki są blokowane z waitem LCK_M_S przez inną sesję o identyfikatorze 58.

Poniżej przykład dla sesji wykonującej kopie bezpieczeństwa danych działające w 3 zadaniach:

Exec Context	Wait	Wait Time [seconds]	Blocking Session id	Blocking Exec Context	Resource Description
0	ASYNC_IO_COMPLETION	315.000			
2	BACKUPIO	0.000			
3	BACKUPBUFFER	0.000			

1.3 Ekran historii aktywnych sesji i sesji używających log-a

W ekranie Historii aktywnych sesji oraz sesji używających pliku logów wprowadzono poniższe usprawnienia:

1.3.1 Dodanie kolumn hostname, context_info

W module zostały dodane kolumny hostname oraz context_info, po których dodatkowo można filtrować. Filtry są dostępne po kliknięciu w link Show additional filters:

Kolumny hostname oraz context_info, są widoczne po kliknięciu w snapshot sesji w zakładkach Sessions/Log Usage:

Sessions		Log usage													
Type	Session Id	Program	Nt user name	Host name	Login name	Context Info	Query Hash	Plan Hash	Wait type	Wait time [seconds]	Blocking session id	Command	Database	Elapsed Time [seconds]	Cpu Time [seconds]
Session	55	.Net SqlClient (WIN-PVM04LT			DxDf1D6547F4EC	0xCB74A269DF		0.0	0	SELECT	adv_works	0.016	0.015
Session	58	.Net SqlClient (Administrator	WIN-PVM04LT	ADDBPLUS\A		Dx603C0FBB2	DxD458D13F64E		0.0	0	SELECT	adv_works	0.006	0.016
Session	61	.Net SqlClient (Administrator	WIN-PVM04LT	ADDBPLUS\A		Dx64C102F23329C	DxF02EB880387		0.0	0	SELECT	adv_works	0.110	0.094
Session	66	.Net SqlClient (WIN-PVM04LT			Dx677E3020F4588	Dx60305CE484A		0.0	0	SELECT	adv_works	0.016	0.016

1.3.2 Pokazywanie wielu oczekiwań dla pojedynczych sesji

Podobnie jak w widoku sesji, przy monitorowaniu aktywnych sesji lub sesji używających loga została dodana logika ze sprawdzaniem oczekiwań dla wszystkich zadań sesji. W przypadku gdy takie oczekiwania wystąpią, wówczas pojawiają się w kolumnie Wait odseparowane przecinkiem.

1.3.3 Dodanie zbiorczych wskaźników dla monitorowanych sesji

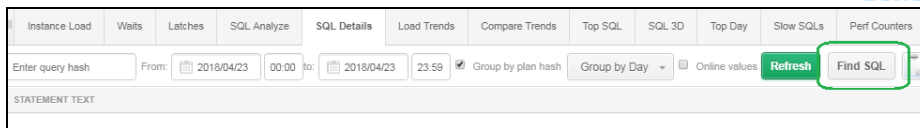
W głównym ekranie historii sesji, dodano poniższe informacje:

- Liczbę sesji używających pliki logów
- Liczbę rekordów/zmian we wszystkich plikach logów
- Rozmiar zmian ze wszystkich plików logów

Logdate	Active Sessions	Log Usage Sessions	Log Usage Record Count	Log Space Used [MB]
2018/04/23 04:07:24	2	1	26	0.1 MB
2018/04/23 04:06:53	1	1	26	0.1 MB
2018/04/23 04:06:23	1	1	26	0.1 MB
2018/04/23 04:05:52	1	1	26	0.1 MB
2018/04/23 04:05:21	2	1	200423	1 596.1 MB
2018/04/23 04:04:51	2	1	14	0.0 MB
2018/04/23 04:04:20	3	1	48024878	11 786.5 MB
2018/04/23 04:03:50	2	1	42934417	10 857.1 MB
2018/04/23 04:03:19	2	1	37537395	9 884.8 MB

1.4 Usprawnienia w opcji SQL Find

Moduł wyszukiwania zapytań SQL Find znajduje się w ekranie SQL Details.



W opcji tej wprowadzono poniższe usprawnienia:

1.4.1 Wyświetlanie dodatkowych statystyk dla wyszukanych zapytań

W każdej zakładce dodano informacje o:

- Czasie trwania zapytania / Elapsed Time
- Czasie użycia procesorów wirtualnych / Cpu Time
- Liczbie wykonań
- Liczbie czytanych bloków:
 - z urządzeń dyskowych
 - z pamięci
- Liczbie zapisywanych bloków
- Liczbie zwracanych rekordów

Poniżej wynik przykładowego wyszukiwania po tekście zapytania:

Statement by text

Plan Flip-Flop Statements

New statements

Enter query hash:

Date from: Date to: Max. returned statements:

CLICK ON [ADD TO SQL DETAILS] BUTTON (ICON WITH +) TO ADD QUERY IDENTIFIER TO QUERY HASHES TOOLBAR LIST

Query Hash	Last execution date	Elapsed Time [Seconds]	Cpu Time [Seconds]	Executions	Disk reads [MB]	Buffer gets [Blocks]	Buffer writes [Blocks]	Rows processed	Query text
0x165730E7D548B0AB	2018-04-23	24.77	20.85	394	0 MB	455 858	0	18 518	select * from dbo.ProductWithTransactio
0x60300FB248FF9E94	2018-04-23	2 637.67	2 261.19	46 500	9 MB	53 800 500	0	46 500	SELECT COUNT(*) FROM [dbo].[Produ
0x64C102F23329DC98	2018-04-23	14 818.37	8 970.26	86 530	124 MB	4 916 333 563	0	1 155 521 620	select top(@v) t.* from Production.Trans
0x677E3020F458809A	2018-04-23	6 816.98	5 671.19	445 436	8 MB	391 092 808	0	445 436	select top ? * from Production.Transacti
0xC234CB9C6A04DF5D	2018-04-23	4.19	3.73	3 915	0 MB	0	0	3 915	select @tc = COUNT(*) FROM sys.dm_!
0xCBE0580B79F27DC6	2018-04-23	276.83	240.54	19 613	0 MB	3 443 399	0	156 904	select p.ProductNumber, p.Name, p.Ma
0xD54F433E1DEA7406	2018-04-23	1.16	1.15	40 280	0 MB	161 120	0	40 280	SELECT COUNT(*) FROM [Production]
0xD9C52915FFD07EF2	2018-04-23	18.54	15.76	3 945	0 MB	8 594	0	33	select SessionTrans.session_id, s.status
0xDB32CA35859FA837	2018-04-23	91.28	73.76	7 138	0 MB	1 253 174	0	57 104	select p.ProductNumber, p.StandardCos

Poniżej wynik przykładowego wyszukiwania zapytaniu uruchamianych w określonym dniu a nie uruchamianych w dniu poprzedzającym:

Statement by text

Plan Flip-Flop Statements

New statements

Statement executed in period

Date from: Date to: Min. elapsed time (sec):

And statement not executed in the period range

Date from: Date to:

CLICK ON [ADD TO SQL DETAILS] BUTTON (ICON WITH +) TO ADD QUERY IDENTIFIER TO QUERY HASHES TOOLBAR LIST

Query Hash	Elapsed Time [Seconds]	Cpu Time [Seconds]	Executions	Disk reads [MB]	Buffer gets [Blocks]	Buffer writes [Blocks]	Rows processed	Query text
0x64C102F23329DC98	15 181.59	9 178.58	88 532	124 MB	5 030 077 193	0	1 182 256 326	select top(@v) t.* from Production.TransactionHistory t where t.Produ
0x677E3020F458809A	6 816.98	5 671.19	445 436	8 MB	391 092 808	0	445 436	select top ? * from Production.TransactionHistory t where t.Transactio
0x60300FB248FF9E94	2 686.68	2 303.32	47 311	9 MB	54 738 827	0	47 311	SELECT COUNT(*) FROM [dbo].[ProductWithTransaction] WHERE [
0x2F5EE731FCEDF74A	293.73	228.99	859	233 MB	25 665 202	0	25 770	select h.CustomerID, h.SalesOrderID, h.OrderDate, h.ShipDate, h.Se
0x4CB9B8522ADAD20F	120.10	118.87	47 311	1 MB	6 055 808	0	47 311	select count(*) from Sales.vIndividualCustomer where BusinessEntity

1.4.2 Usprawnienie w analizie zapytań, które zmieniają plan wykonania

Dla zapytań zmieniających plan wykonania, zostały dodane dodatkowe informacje zgrupowane wg poniższych obszarów:

- Statystyki z podsumowaniem dla wszystkich planów wykonania, na którym działało zapytanie
- Statystyki z podsumowaniem dla najwolniejszego planu
- Statystyki z podsumowaniem dla najszybszego planu
- Porównanie najwolniejszego i najszybszego
- Estymacje o możliwej redukcji czasu trwania zapytania

Poniżej przykład z wynikiem wyszukania tych zapytań które zmienił plan wykonania w okresie 2 tygodni:

Widok na obszary *Total statistics*, *Slowest plan statistics*

Statement by text		Date from:	2018/04/10	00:00	Date to:	2018/04/24	23:59				
Plan Flip-Flop Stateme...											
New statements											
Search											
CLICK ON [ADD TO SQL DETAILS] BUTTON (ICON WITH +) TO ADD QUERY IDENTIFIER TO QUERY HASHES TOOLBAR LIST											
		Total statistics				Slowest plan statistics					
Query Hash	Query text	Elapsed Time [Seconds]	Cpu Time [Seconds]	Executions	Number of plans	Plan Hash	Elapsed Time [Seconds]	Cpu Time [Seconds]	Executions	Elapsed Time Per 1 exec [Seconds]	
Dx64C102F23329DC98	select top(@v)	486 925.70	394 926.91	426 821	2	Dx31F805092B25	442 606.41	367 606.00	156 096	2.8355	
DxA86C6E5BE207D6E8	select max(Err	70.20	24.68	43	2	Dx397376A5E330	52.39	19.52	21	2.4946	
Dx25B65C61193863C4	select * from Pr	11 726.99	10 221.35	1 176 774	3	DxD445611DDDBA	420.77	138.82	1 773	0.2373	
DxE95D16F7F24BD1F3	SELECT DB_I	68.70	60.64	6 695	2	Dx2370E781E95E	25.13	22.26	1 339	0.0188	
Dx24BFF45573B477FD	select convert(i	98.46	88.74	1 343	2	Dx89C31130AB10	26.72	24.42	343	0.0779	
Dx89EB3EE49C2797CF	select * as rec	16.09	15.89	20 742	2	Dx43B435618BC6	7.77	7.68	6 612	0.0012	

Widok na obszary *Fastest plan statistics*, *Slowest vs. Fastest*, *Estimation statistics*.

		Fastest plan statistics				Slowest vs Fastest			Estimation statistics	
Plan Hash	Elapsed Time Per 1 exec [Seconds]	Plan Hash	Elapsed Time [Seconds]	Cpu Time [Seconds]	Executions	Elapsed Time Per 1 exec [Seconds]	Times faster	Elapsed Time Per 1 exec difference [Seconds]	Elapsed Time to reduce [Seconds]	Cpu Time to reduce [Seconds]
Dx02E8B03876	2.8355	Dx02E8B03876	44 319.29	27 320.91	270 725	0.1637	17	2.6718	417 052.5628	351 853.1681
DxFE2C0C637B8	2.4946	DxFE2C0C637B8	17.82	5.16	22	0.8098	3	1.6848	35.3808	14.5963
Dx90B998ECB7C	0.2373	Dx90B998ECB7C	8 388.18	7 310.25	1 169 949	0.0072	33	0.2301	3 289.8811	2 868.4574
Dx43E66D931657	0.0188	Dx43E66D931657	43.57	38.38	5 356	0.0081	2	0.0106	14.2387	12.6670
Dx2B459523C16C	0.0779	Dx2B459523C16C	71.73	64.33	1 000	0.0717	1	0.0062	2.1186	2.3535
DxC99C4CF8765	0.0012	DxC99C4CF8765	8.32	8.21	14 130	0.0006	2	0.0006	3.8777	3.8421

Istotny obszar ekranu **Plan Flip-Flop Statements** to *Estymacja statystyk*. Kolumny **Elapsed Time to reduce** oraz **Cpu Time to reduce**, to wyliczenie o możliwej redukcji czasu dla przypadku kiedy zapytanie działało by wyłączenie na najszybszym planie wykonania.

Przydatna wskazówka:

Posortowanie po jednej z tych kolumn pozwoli znaleźć te zapytania, których optymalizacja przyniesie największą poprawę wydajności.

1.5 Aktualizacja ekranu Application architecture

Ekran dostępny z poziomu menu głównego **Servers monitor -> Application architecture** został przebudowany i uzupełniony o dodatkowe informacje dotyczące aktywności procesu monitoringu.

W obszarze z lewej strony są dostępne instancje SQL, gdzie widać:

- Kiedy był wykonany ostatni snapshot dla monitorowanej instancji SQL
- Kiedy była ostatnia aktywność instancji SQL (połączenie z serwisu monitoringu z instancją SQL)

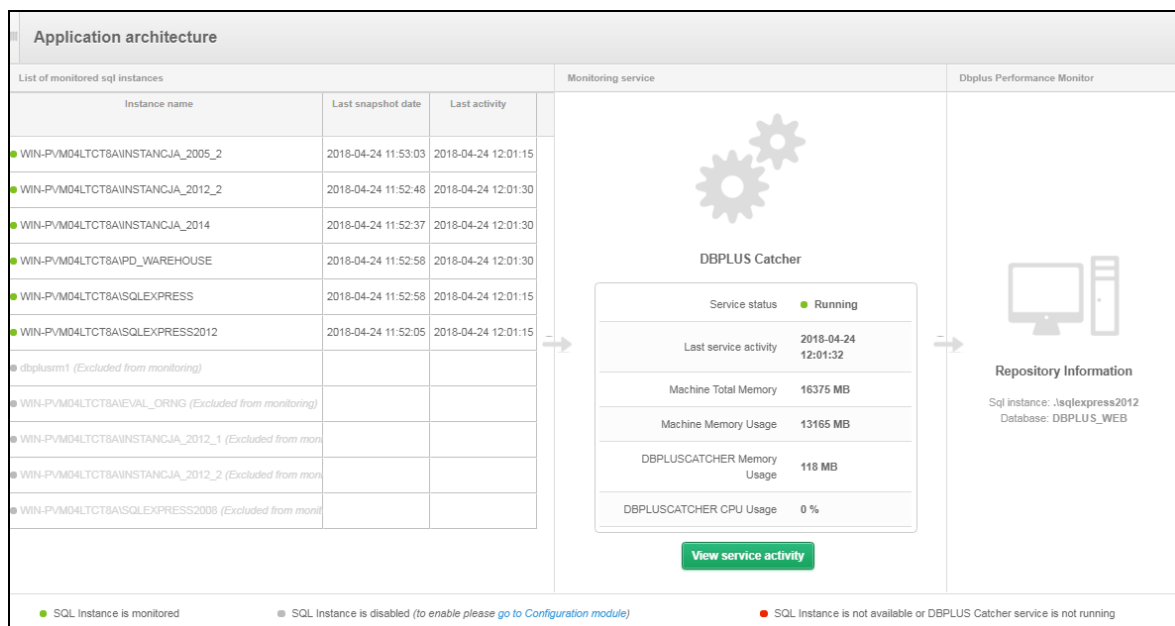
W obszarze środkowym znajduje się informacja o aktualnym stanie usługi monitoringu DBPLUSCATCHER, min. są dostępne takie informacje jak:

- Czy service jest uruchomiony
- Ostatnia aktywność usługi
- Wykorzystanie pamięci na maszynie gdzie działa usługa monitoringu
- Użycie procesora przez usługę monitoringu

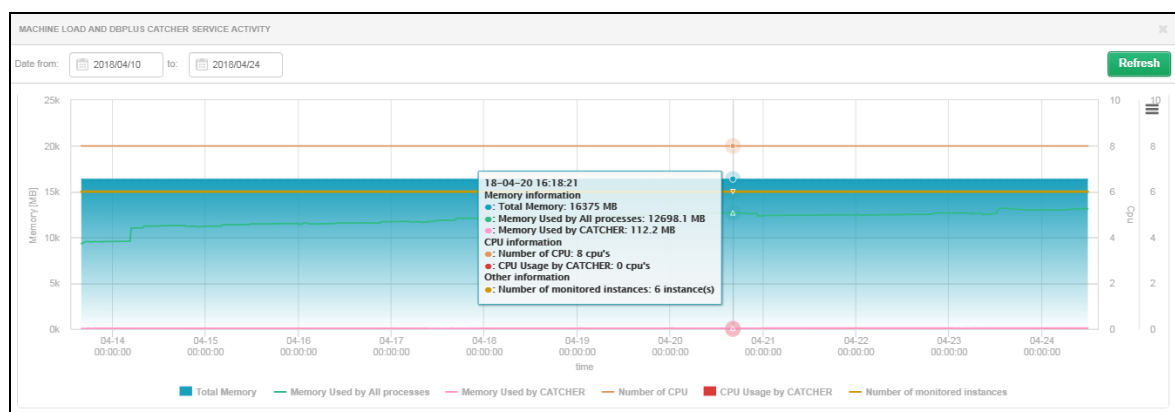
Poniżej statystyk można sprawdzić historyczny stan usługi w zadanym okresie czasu.

W obszarze z prawej strony jest informacja o instancji SQL, na której znajduje się repozytorium systemu DBPLUS Performance Monitor.

Poniżej przykładowy slajd:



W poniższym slajdzie historia aktywności usługi DBPLUSCATCHER, po kliknięciu w przycisk **View service activity**:



1.6 Zmiany w usłudze monitoringu DBPLUSCATCHER

1.6.1 Dodanie procesu przebudowy tabel używanych w ekranie Dashboard

Dla wybranych tabel używanych przez usługę monitoring wdrożono mechanizm przebudowy tabel uruchamiany 1 raz w tygodniu.

1.6.2 Błąd w monitoringu baz danych w edycji SQL Azure

W instancji SQL Azure występował błąd w monitoringu w przypadku dodania nowych baz danych w instancji Azure. Problem był wynikiem braku uprawnień użytkownika monitoringu w określonej bazie danych.

Problem został naprawiony. W ekranie Logs pojawiają się informacje o bazach, do których usługa monitoringu nie ma dostępu.

1.6.3 Aktualizacja słownika waitów NIE-performance-owych - CXPACKET

W słowniku system DBPLUS Performance Monitor zostało dodane kolejne oczekiwanie tj. CXPACKET, które **NIE będzie należeć do grupy waitów wydajnościowych**.

W wielu przypadkach poziom waita CXPACKET jest na tyle duży, że nie widać stanu innych oczekiwań i na jakiego rodzaju operacjach instancja SQL, spędza czas.

Stąd eliminacja waita CXPACKET pozwoli na szybszą analizę problemów wydajnościowych.

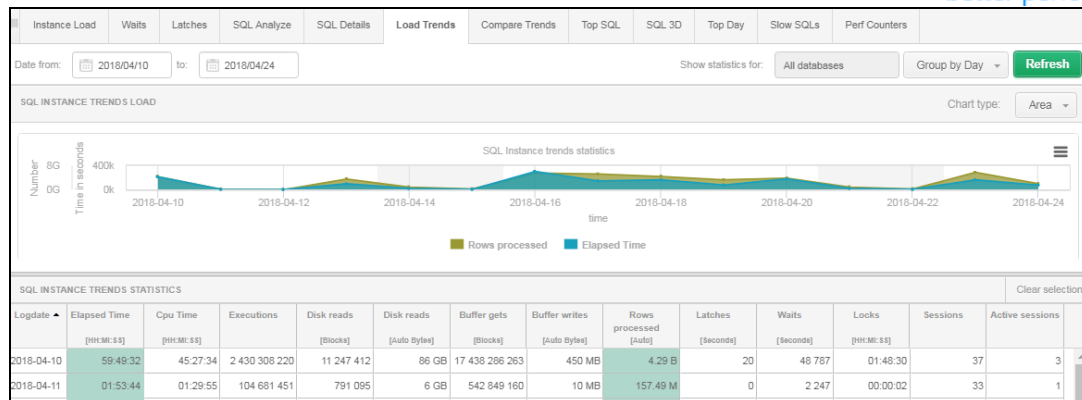
1.7 Usprawnienia ogólne

1.7.1 Zmiana rozmiaru/wysokości wykresów

W ekranach min.

- Instance Load
- Waits
- Latches
- Sql Analyze
- Load Trends
- Performance Counters
- IO Stats
- Space Monitor

Pojawiła się możliwość zmiany wysokości wykresu – poniżej przykład z przeskalowanym wykresem w ekranie Load Trends:



1.7.2 Wykrywanie problemu z niedziałającym monitoringiem

W systemie poprawiono mechanizm wykrywania problemów w działaniu usługi monitoringu – bardzo częstym przypadkiem był brak miejsca w bazie repozytorium i wszelkie informacje o błędach w działaniu usługi łądowały w pliku logów na maszynie z oprogramowaniem DBPLUS Performance Monitor.

W nowej wersji, na ekranie dashboardu pokazuje się alert w przypadku

- problemu z brakiem miejsca w repozytorium
- zatrzymanej usłudze monitoringu.
- braku zasobów na maszynie z usługą monitoringu

Dodatkowo bieżące informacje z aktywności usługi DBPLUSCTCHER można wyświetlić w ekranie **Servers monitor->Application architectre**.

1.7.3 Niepoprawne sortowanie w ekranie Slow SQL

W systemie poprawiono błąd z domyślnie używanym sortowaniem w funkcjonalności Slow SQL. W nowej wersji po wyświetleniu statystyki zapytań z zadanyim okresem, tabela jest domyślnie uporządkowana wg kolumny Elapsed Time (sortowanie malejące)