



Exigo Admin
Sync Management

v 1.0

Contents

Overview	2
Sync Destinations	3
Summary:	4
Core tables:	4
Managed Reindexing:	5
Data Filtering:	6
SQL DataBase Reporting Dashboard	7
Reporting:	7
Sessions:	8
Processes:	9
Wait Stats:	10
Top 10 Queries:	11
Top 10 SQL Statements:	11
Index Fragmentation:	12
Compression Settings:	13
SQL Configuration:	13
Azure:	14
Azure Database Size:	15
Azure Log Usage:	15
Memory:	16

OVERVIEW

Sync destination performance is crucial to your business. Microsoft provides a full suite of SQL server monitoring features built directly into Exigo Sync that are accessible directly within Exigo Admin.

These features will allow you to get key performance analytics from your sync destinations and will give you the information you need to finely tune each SQL instance.

Driven by industry standards, Exigo Sync Monitoring will collect data points that matter most, some of the included features are listed below.

- Query Performance
 - Most expensive queries
 - Wait statistics
 - Execution plan insights
- Active Workloads
- Database growth (data and log files)
- Resource utilization
- Index Fragmentation
- Availability Group Health
- Backup metrics

SYNC DESTINATIONS

If you have opted out of creating a Shared BI Sync DB via Exigo when initiating your sandbox, you have the option of using your own Server to create a Synced copy of your data.

To add a Sync Connection, go to Company Portal > Sync Management > New Destination

Sync Destinations				NEW DESTINATION	
DESCRIPTION	SERVER	DATABASE	ENABLED		
Sync Destination 1	Server Description 1	Database Description 1	✗	VIEW	⋮
Sync Destination 2	Server Description 2	Database Description 2	✓	VIEW	⋮
Sync Destination 3	Server Description 3	Database Description 3	✓	VIEW	⋮
Sync Destination 4	Server Description 4	Database Description 4	✓	VIEW	⋮
Sync Destination 5	Server Description 5	Database Description 5	✗	VIEW	⋮

From here, enter your credentials for the existing Server you want to sync your data to and click Save Changes.

Manage Sync Destination ⌵ ✕

Destination

Name

Data Source

Initial Catalog

User ID

Password

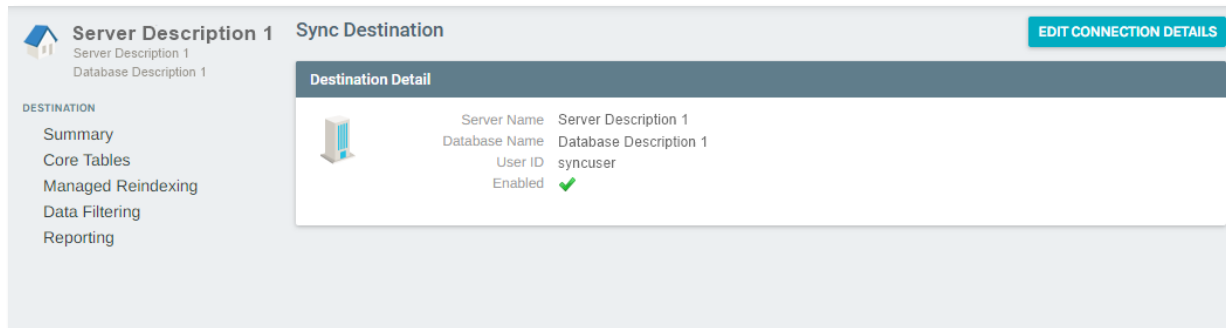
Encrypt connection

SAVE CHANGES
TEST CONNECTION
CANCEL

SUMMARY:

To see the summary of details for your Sync Destinations click on the desired option on the Sync Management page.

From here you will be able to review the details of your Sync Destination and Edit the Connection Details for the connection.



Server Description 1 Sync Destination EDIT CONNECTION DETAILS

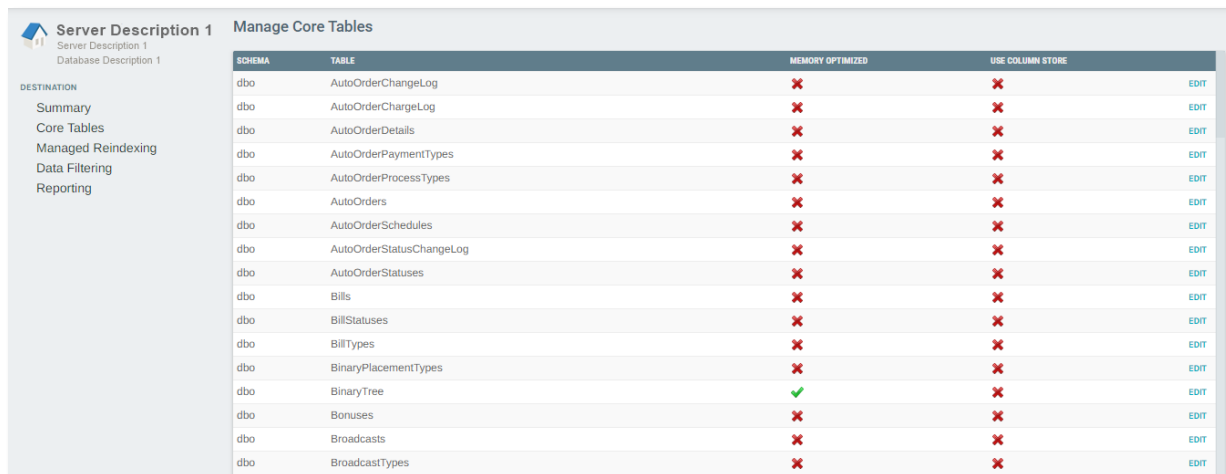
Server Description 1
Database Description 1

Destination Detail

Server Name: Server Description 1
Database Name: Database Description 1
User ID: syncuser
Enabled: ✔

CORE TABLES:

To see a list of all of the Core Exigo DB tables and manage how they are indexed and stored in the DB you can go to Company Portal > Sync Management > Select Sync Destination > Core Tables.



SCHEMA	TABLE	MEMORY OPTIMIZED	USE COLUMN STORE	
dbo	AutoOrderChangeLog	✘	✘	EDIT
dbo	AutoOrderChargeLog	✘	✘	EDIT
dbo	AutoOrderDetails	✘	✘	EDIT
dbo	AutoOrderPaymentTypes	✘	✘	EDIT
dbo	AutoOrderProcessTypes	✘	✘	EDIT
dbo	AutoOrders	✘	✘	EDIT
dbo	AutoOrderSchedules	✘	✘	EDIT
dbo	AutoOrderStatusChangeLog	✘	✘	EDIT
dbo	AutoOrderStatuses	✘	✘	EDIT
dbo	Bills	✘	✘	EDIT
dbo	BillStatuses	✘	✘	EDIT
dbo	BillTypes	✘	✘	EDIT
dbo	BinaryPlacementTypes	✘	✘	EDIT
dbo	BinaryTree	✔	✘	EDIT
dbo	Bonuses	✘	✘	EDIT
dbo	Broadcasts	✘	✘	EDIT
dbo	BroadcastTypes	✘	✘	EDIT

Memory Optimized

Setting tables to memory Optimized can help improve performance of tables. For more information about memory optimized tables, please see the official Microsoft documentation located [here](#).

Column Store

Applying a Column Store index can improve speeds drastically when it comes to Data Analytics. For more information on Column Store indexes, please see the official Microsoft documentation located [here](#).

MANAGED REINDEXING:

To manage your core Exigo DB table indexes navigate to Company Portal > Sync Management > Select Sync Destination > Managed Reindexing.

From here you can control and configure if and when table indexes are rebuilt.

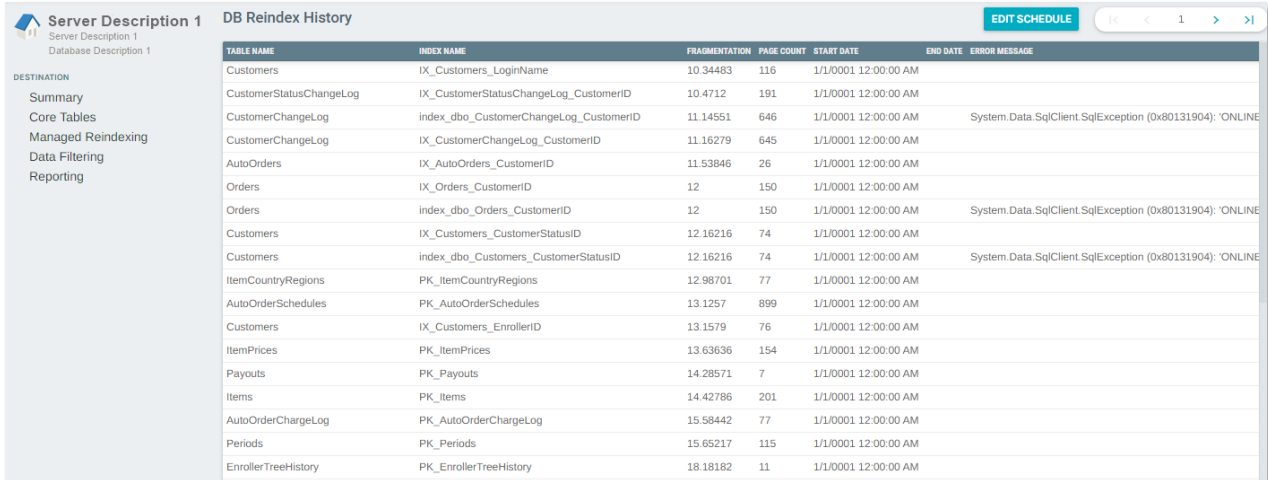
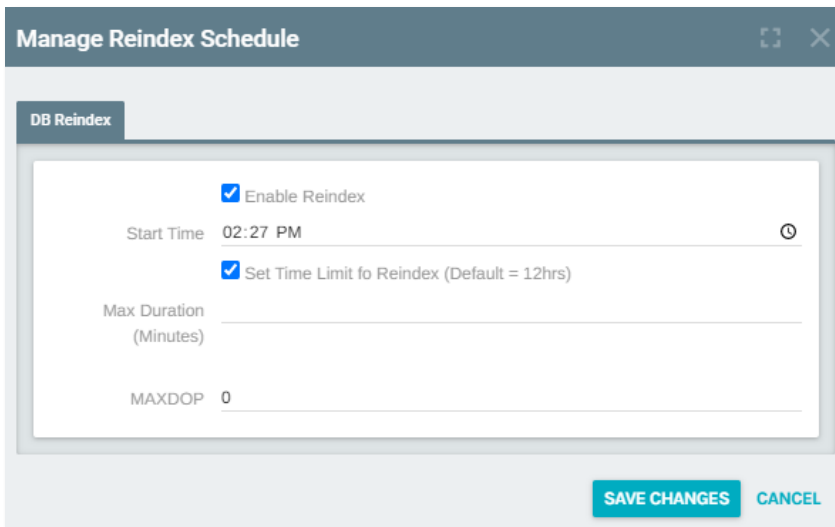


TABLE NAME	INDEX NAME	FRAGMENTATION	PAGE COUNT	START DATE	END DATE	ERROR MESSAGE
Customers	IX_Customers_LoginName	10.34483	116	1/1/0001 12:00:00 AM		
CustomerStatusChangeLog	IX_CustomerStatusChangeLog_CustomerID	10.4712	191	1/1/0001 12:00:00 AM		
CustomerChangeLog	index_dbo_CustomerChangeLog_CustomerID	11.14551	646	1/1/0001 12:00:00 AM		System.Data.SqlClient.SqlException (0x80131904): 'ONLINE
CustomerChangeLog	IX_CustomerChangeLog_CustomerID	11.16279	645	1/1/0001 12:00:00 AM		
AutoOrders	IX_AutoOrders_CustomerID	11.53846	26	1/1/0001 12:00:00 AM		
Orders	IX_Orders_CustomerID	12	150	1/1/0001 12:00:00 AM		
Orders	index_dbo_Orders_CustomerID	12	150	1/1/0001 12:00:00 AM		System.Data.SqlClient.SqlException (0x80131904): 'ONLINE
Customers	IX_Customers_CustomerStatusID	12.16216	74	1/1/0001 12:00:00 AM		
Customers	index_dbo_Customers_CustomerStatusID	12.16216	74	1/1/0001 12:00:00 AM		System.Data.SqlClient.SqlException (0x80131904): 'ONLINE
ItemCountryRegions	PK_ItemCountryRegions	12.98701	77	1/1/0001 12:00:00 AM		
AutoOrderSchedules	PK_AutoOrderSchedules	13.1257	899	1/1/0001 12:00:00 AM		
Customers	IX_Customers_EnrollerID	13.1579	76	1/1/0001 12:00:00 AM		
ItemPrices	PK_ItemPrices	13.63636	154	1/1/0001 12:00:00 AM		
Payouts	PK_Payouts	14.28571	7	1/1/0001 12:00:00 AM		
Items	PK_Items	14.42786	201	1/1/0001 12:00:00 AM		
AutoOrderChargeLog	PK_AutoOrderChargeLog	15.58442	77	1/1/0001 12:00:00 AM		
Periods	PK_Periods	15.65217	115	1/1/0001 12:00:00 AM		
EnrollerTreeHistory	PK_EnrollerTreeHistory	18.18182	11	1/1/0001 12:00:00 AM		

Edit Schedule

To edit/manage the Index rebuild schedule click on Edit Schedule in the top right.

From here you can enable/disable reindexing of table indexes and control how long a rebuild is to take.



Manage Reindex Schedule

DB Reindex

Enable Reindex

Start Time: 02:27 PM

Set Time Limit for Reindex (Default = 12hrs)

Max Duration (Minutes): _____

MAXDOP: 0

SAVE CHANGES **CANCEL**

- **Enable Reindex** – Ability to enable and disable the automated process of rebuilding table indexes
- **Start Time** – Here you can define the time you want the rebuilding of indexes to occur. Best practice dictates you set these to be rebuilt during non-peak times for your business.
- **Set Time Limit for Reindex (Default = 12hrs)** – Allows you to control the max amount of time a rebuild is allowed to take. When selected you are able to provide a max amount of time in the following field.

- **Max Duration (Minutes)** – Allows you to define how long the process of rebuilding an index is allowed to take. If no value is provided, it will by default allow a maximum of 12 hours.
- **Max DOP** – Max degree of parallelism (MAXDOP) is a setting in SQL Server that controls how many processors may be used for parallel plan execution. Parallel plan execution is good—it lets SQL Server make the best use of all those processors in modern servers. This setting allows you to define how many processors may be used in the rebuilding of your indexes.

DATA FILTERING:

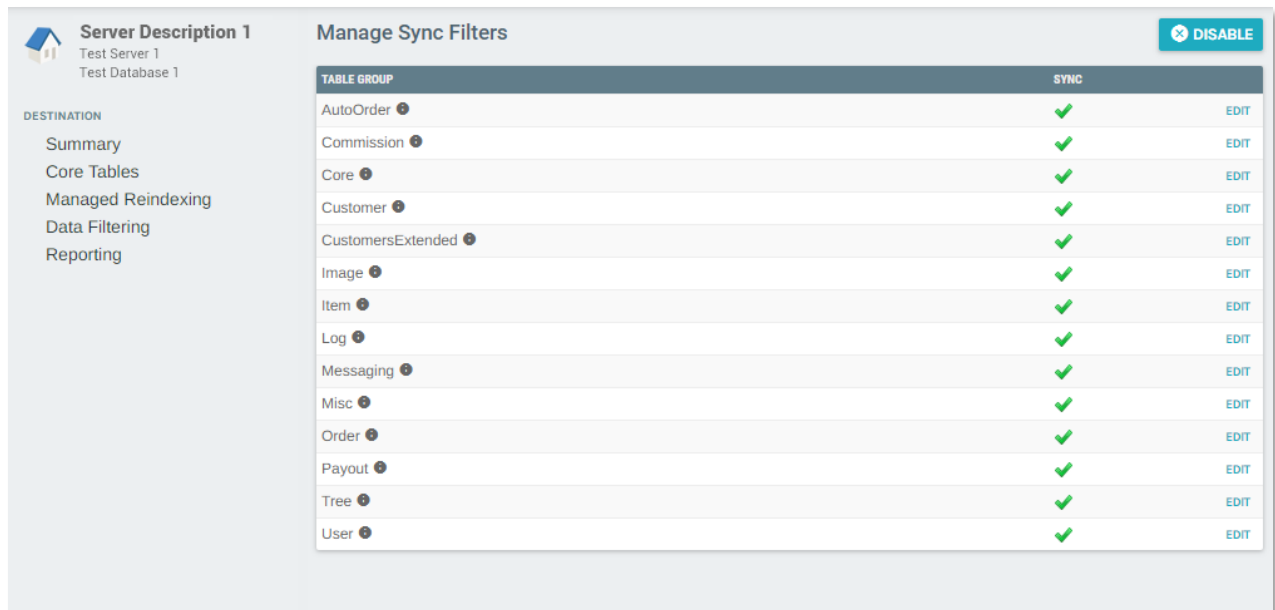
To manage which Exigo Core tables are included in Sync navigate to Company Portal > Sync Management > Select Sync Destination > Data Filtering.

From here you can select which core Exigo Table groups are included in Sync.

To see a list of included tables for each group you may hover over the tool tip located next to the Table Group name.

To disable all tables from being included in Sync, you may use the Disable button located at the top right.

To disable individual tables you can click Edit to the right of a specific Table group, uncheck “Enable Group Sync” and click Save Changes.



Server Description 1
Test Server 1
Test Database 1

DESTINATION
Summary
Core Tables
Managed Reindexing
Data Filtering
Reporting

Manage Sync Filters DISABLE

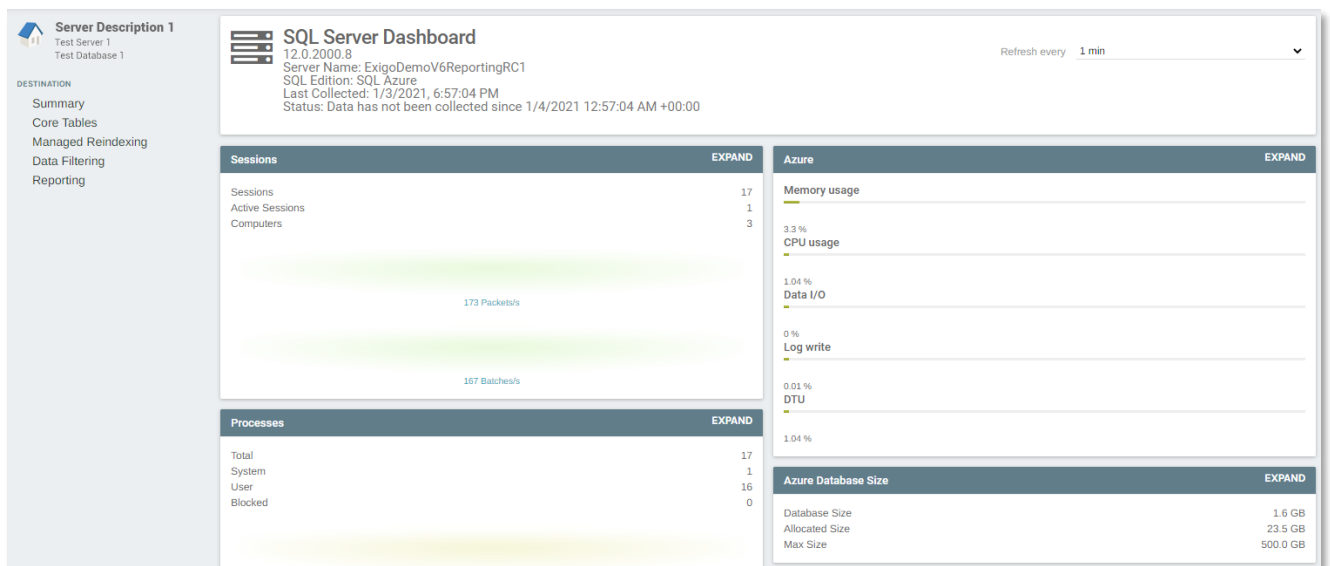
TABLE GROUP	SYNC	
AutoOrder	✓	EDIT
Commission	✓	EDIT
Core	✓	EDIT
Customer	✓	EDIT
CustomersExtended	✓	EDIT
Image	✓	EDIT
Item	✓	EDIT
Log	✓	EDIT
Messaging	✓	EDIT
Misc	✓	EDIT
Order	✓	EDIT
Payout	✓	EDIT
Tree	✓	EDIT
User	✓	EDIT

SQL DATABASE REPORTING DASHBOARD

REPORTING:

This section in the Sync Management menu serves up a robust set of comprehensive tools that allows a user to review the health of their DB as well as various metrics pertaining to the activity taking place on said DB. While more geared towards someone with a technical skillset such as a DBA or IT Team, this section will be covered later in more detail.

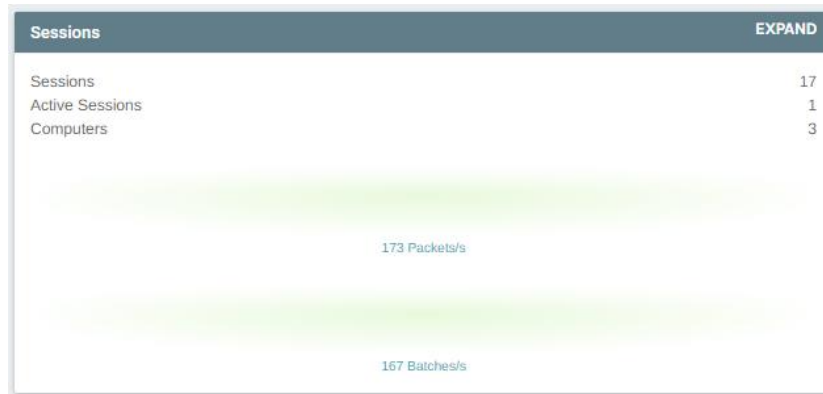
Click [here](#) to jump ahead.



Below we will provide a high-level overview of each section on the dashboard and drill down into each of them where possible.

SESSIONS:

This section provides a high level overview of session related data as well as packet and batch related data.

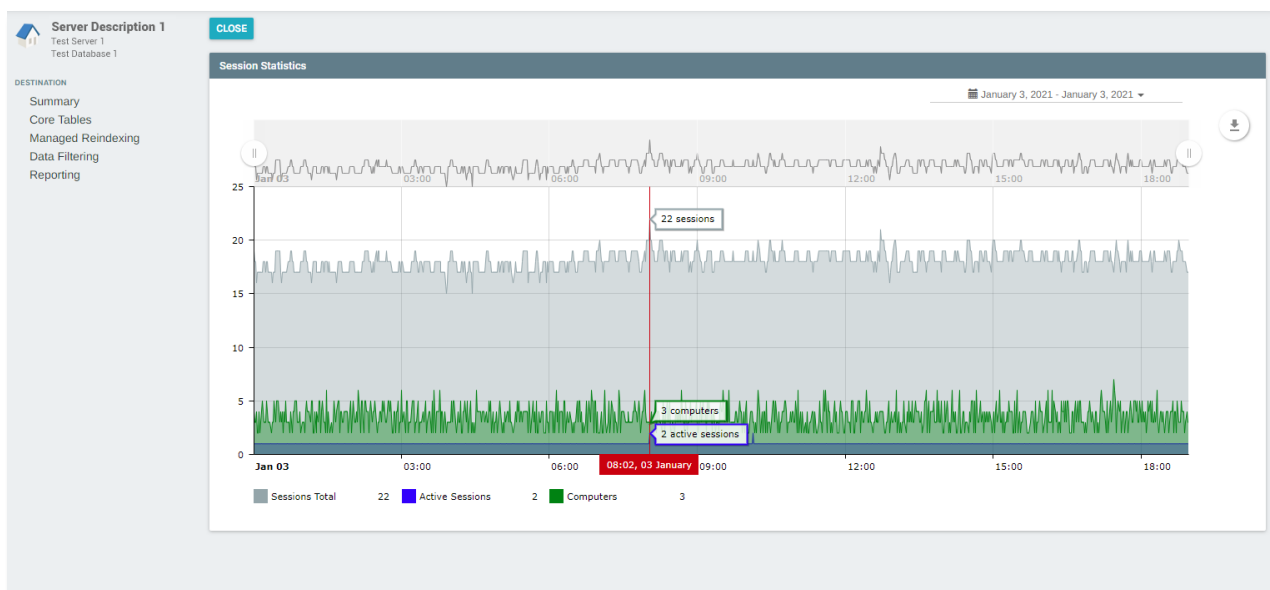


Packets: This flow represents the rate at which network packets are being received by SQL Server from client applications.

Batches: This flow shows the rate at which Batches of SQL statements are being submitted to SQL Server from execution.

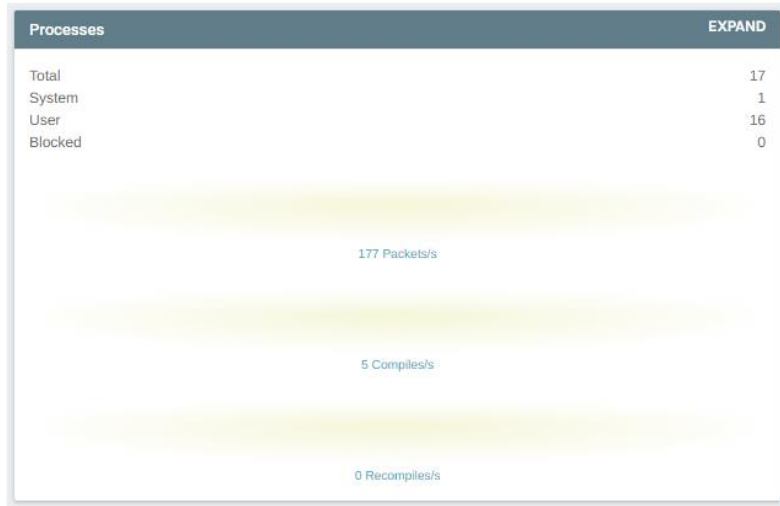
You can click “Expand” to drill down and see more information related the server sessions using an interactive graph as shown below.

From here you can view and download session data.



PROCESSES:

This section provides a high level overview of the processes being run against the Server as well as packet and compilation overviews.



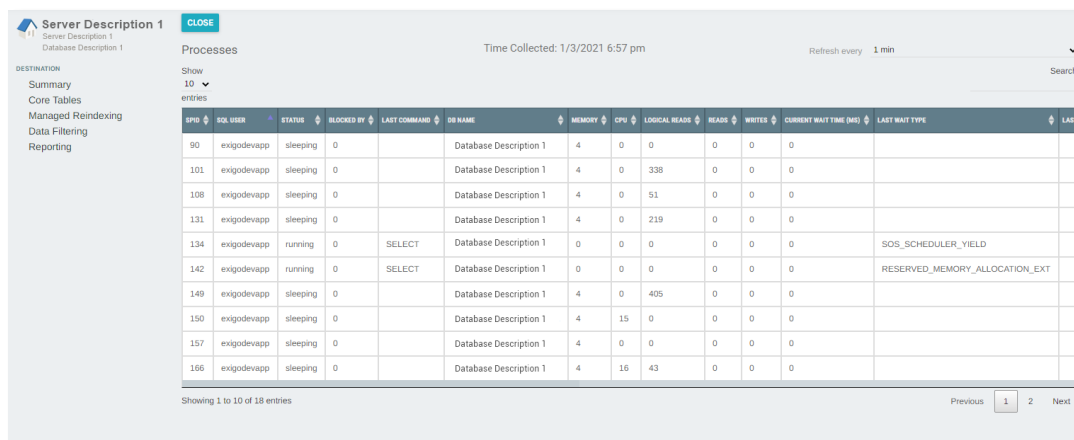
Packets: This flow represents the rate at which network packets are being sent from SQL Server to client applications.

Compiles: This flow shows the rate of SQL compilations.

Recompiles: This flow shows the rate of SQL recompilations.

You can click “Expand” to drill down and see more information related the server processes using an interactive Kendo Grid to sort and filter.

This section helps to provide a high level overview of the current SQL Server connections and their state. In turn this provides DBAs with data needed to troubleshoot application timeouts, high CPU time, and high disk IO.



SPID	SQL_USER	STATUS	BLOCKED BY	LAST COMMAND	DB NAME	MEMORY	CPU	LOGICAL READS	READS	WRITES	CURRENT WAIT TIME (MS)	LAST WAIT TYPE	LAST
90	exigodevapp	sleeping	0		Database Description 1	4	0	0	0	0	0		
101	exigodevapp	sleeping	0		Database Description 1	4	0	338	0	0	0		
108	exigodevapp	sleeping	0		Database Description 1	4	0	51	0	0	0		
131	exigodevapp	sleeping	0		Database Description 1	4	0	219	0	0	0		
134	exigodevapp	running	0	SELECT	Database Description 1	0	0	0	0	0	0	SOS_SCHEDULER_YIELD	
142	exigodevapp	running	0	SELECT	Database Description 1	0	0	0	0	0	0	RESERVED_MEMORY_ALLOCATION_EXT	
149	exigodevapp	sleeping	0		Database Description 1	4	0	405	0	0	0		
150	exigodevapp	sleeping	0		Database Description 1	4	15	0	0	0	0		
157	exigodevapp	sleeping	0		Database Description 1	4	0	0	0	0	0		
166	exigodevapp	sleeping	0		Database Description 1	4	16	43	0	0	0		

Showing 1 to 10 of 18 entries

WAIT STATS:

Wait stats are captured and recorded by SQL Server and all this captured information is referred to as wait statistics.


The promise of wait stats is to help quickly identify where the most pressing bottlenecks are building within SQL Server. This data provides additional insight that can aid a DBA or technical resource in resolving problems that are related to SQL Server performance.

Wait Stats
EXPAND

Top 5 wait types

- HADR_WORK_QUEUE
- XE_LIVE_TARGET_TVF
- HADR_FILESTREAM_IOMGR_IOCOMPLETION
- PWAIT_EXTENSIBILITY_CLEANUP_TASK
- HADR_FABRIC_CALLBACK

You can click "Expand" to drill down and see more information related the Wait statistics using an interactive Kedo Grid to sort and filter data.



Server Description 1

Server Description 1

Database Description 1

Wait Statistics

Time Collected: 1/3/2021 6:57 pm

Refresh every 1 min

Search:

Show 25 entries

WAIT TYPE	CATEGORY	WAITING TASKS RATE (TASKS/S)	WAIT TIME RATE (MS/S)	SIGNAL WAIT TIME RATE (MS/S)	WAITING TASKS COUNT	WAIT TIME (MS)	SIGNAL WAIT TIME (MS)	WAITING
HADR_WORK_QUEUE	Idle	470225.144	54933524.671	153323.112	522664652	61059711342	170421705	49.01
XE_LIVE_TARGET_TVF	Idle	276.096	2108720.355	80.334	306886	2343884849	89293	0.029
HADR_FILESTREAM_IOMGR_IOCOMPLETION	Idle	2090.296	1054424.614	346.609	2323406	1172014047	385263	0.218
PWAIT_EXTENSIBILITY_CLEANUP_TASK	MISC	3.515	1054262.735	1054262.735	3907	1171834115	1171834115	0.000
HADR_FABRIC_CALLBACK	AlwaysOn	3.000	1054247.987	1.835	3335	1171817723	2040	0.000
HADR_TIMER_TASK	Idle	17388.346	1054164.789	1117.318	19327494	1171725246	1241921	1.813
PVS_PREALLOCATE	MISC	2774.393	1053333.806	1960.944	3083793	1170801592	2179629	0.288
SOS_SCHEDULER_YIELD	SOS	39347.863	27027.077	27020.109	43735937	30041137	30033391	4.102
BACKUPIO	Backup	9147.077	19731.923	2652.666	10167159	21932427	2948491	0.954
WRITELOG	IO	13456.039	18357.809	10433.332	14956657	20405205	11596857	1.402

TOP 10 QUERIES:

This section provides an overview of what queries are being run against your server.

Top 10 queries
EXPAND

You can click "Expand" to drill down further on the top 10 queries and their performance metrics using an interactive Kedo Grid to sort and filter data.

Server Description 1 CLOSE
Time Collected: 1/3/2021 6:57 pm
Refresh every 1 min

DESTINATION

- Summary
- Core Tables
- Managed Reindexing
- Data Filtering
- Reporting

Top 10 queries

EXECUTION COUNT	TOTAL WORKER TIME	AVERAGE CPU TIME	TOTAL READS	TOTAL WRITERS	TEXT
No data available in table					

Showing 0 to 0 of 0 entries

TOP 10 SQL STATEMENTS:

This section provides an overview of the performance details for queries that are being run against your server.

Top SQL Statements
EXPAND

You can click "Expand" to drill down further on the top 10 queries and their performance metrics using an interactive chart. Clicking each bar will further drill into time slices, a day, an hour, a 10 minute segment, etc.

Server Description 1 CLOSE
ExigoDemoV6ReportingRC1
12/30/2020 - 1/1/2021

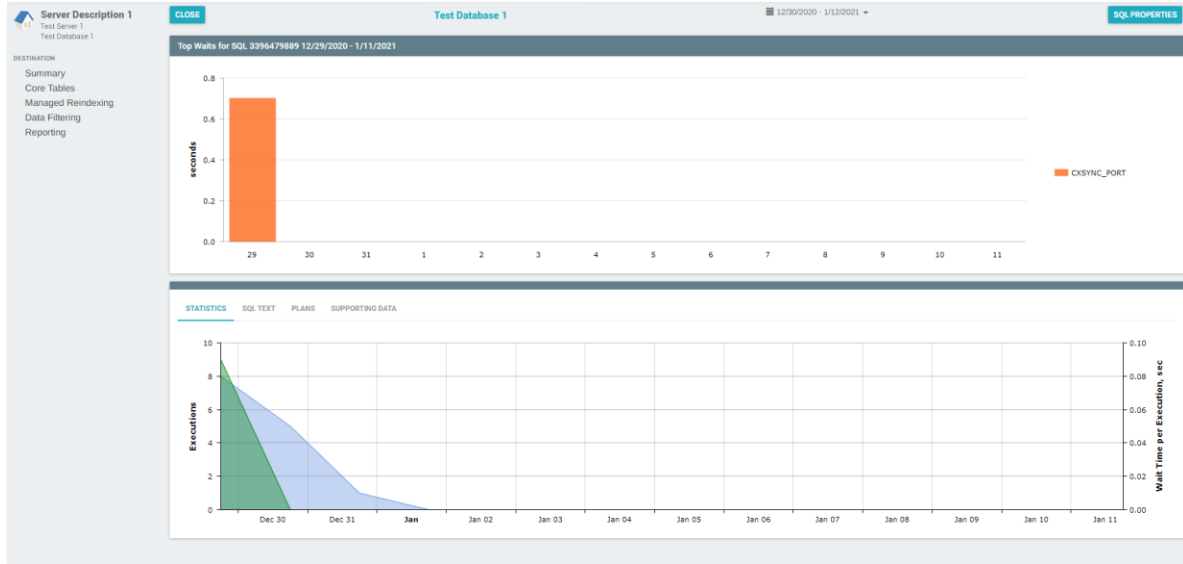
DESTINATION

- Summary
- Core Tables
- Data Filtering
- Reporting

Top SQL Statements 12/29/2020 - 1/1/2021

SQL STATEMENT	Execution Count	Approx. Seconds
update t set PayoutID=tt.PayoutID,BillID=tt.BillID,ModifiedDate=tt.ModifiedDate from PayoutBills t inner join Temp_PayoutBills tt on t.TransactionID = tt.TransactionID insert PayoutBills(TransactionID,PayoutID,BillID,ModifiedDate) select tt.TransactionID,tt.Pay...	29	~2.0
update t set PayoutID=tt.PayoutID,BillID=tt.BillID,ModifiedDate=tt.ModifiedDate from PayoutBills t inner join Temp_PayoutBills tt on t.TransactionID = tt.TransactionID insert PayoutBills(TransactionID,PayoutID,BillID,ModifiedDate) select tt.TransactionID,	30	~1.5
(@1 IN (@2 varchar(3000)UPDATE [sync].[MergeMultiSettings] set [RemoteSequence] = @@1 WHERE [SchemaName]=@2	31	~0.8
update t set CustomerID=tt.CustomerID,BillStatusID=tt.BillStatusID,DueDate=tt.DueDate,BillTypeID=tt.BillTypeID,CurrencyCode=tt.CurrencyCode,Amount=tt.Amount,CommissionRunID=tt.CommissionRunID,IsOtherIncome=tt.IsOtherIncome,Refer...	1	~0.2
update t set CustomerID=tt.CustomerID,OrderStatusID=tt.OrderStatusID,OrderDate=tt.OrderDate,CurrencyCode=tt.CurrencyCode,WarehouseID=tt.WarehouseID,ShipMethodID=tt.ShipMethodID,OrderTypeID=tt.OrderTypeID,PriceTypeID=tt.PriceTypeID,FirstName...	2	~0.5
update t set CustomerID=tt.CustomerID,SubmitDate=tt.SubmitDate,MailTo=tt.MailTo,Subject=tt.Subject,BroadCastID=tt.BroadCastID,DeliveryStatus=tt.DeliveryStatus from EmailOutLog t inner join Temp_EmailOutLog tt on t.OutMailID = tt.OutMailID insert EmailOu...	3	~0.2

Additionally, you may click on the SQL Hash Name located at the bottom of the page to drill down further into each of the individual queries and see performance metrics related to each of them. The SQL Text tab on the bottom will show the actual query statement being run. The Supporting Data tab will show which Hosts, Users and Applications have been running the query for the time slice. Clicking on the SQL Properties button at the top right will allow you to name the SQL Statement so you can easily identify it over time.



INDEX FRAGMENTATION:

This section provides an overview of details pertaining to table index fragmentation.

Heavily fragmented indexes can degrade query performance and cause the application accessing it to respond slowly at times. This tool serves to help identify those instances and aid a user in determining how they should schedule index rebuilds for a specific table.

Index fragmentation
EXPAND

You can click "Expand" to drill down further on the top 10 queries and their performance metrics using an interactive chart.

The screenshot shows the 'Index Fragmentation' table with the following data:

SCHEMA	TABLE	INDEX	INDEX TYPE DESCRIPTION	FRAGMENTATION %	PAGE COUNT
dbo	Bills	PK_Bills	IN_ROW_DATA	99.30	5710
dbo	CustomerChangeLog	PK_CustomerChangeLog	IN_ROW_DATA	82.45	2006

Showing 1 to 2 of 2 entries

COMPRESSION SETTINGS:

This section provides a on overview of data compression settings related to the tables in your database.

Compression Settings
EXPAND

You can click "Expand" to drill down further on the top 10 queries and their performace metrics using an interactive chart.

Server Description 1

Test Server 1
Test Database 1

Summary
Core Tables
Managed Rerouting
Data Filtering
Reporting

Table Compression Settings of Test Database 1

Time Collected: 1/3/2021 12:30 am

Search

DATABASE	SCHEMA	TABLE	INDEX ID	INDEX NAME	INDEX TYPE	ROWS	PARTITION	COMPRESSION	STATUS	PROCESSED	TOTALMB	OBJECTID	PARTITIONID	INDEXPROPERTYID	INDEXID	INDEXPROPERTYID
Test Database 1	dbo	EnvelopeTree	0		HEAP	26293	1	NONE	0	0	73632523	7205766116179680		true		0
Test Database 1	dbo	LHLevelTree	0		HEAP	26742	1	NONE	0	0	787844319	7205766300786600		true		0
Test Database 1	dbo	BinaryTree	0		HEAP	15539	1	NONE	0	0	1472232099	7205766444547970		true		0
Test Database 1	sync	ReplayErrorLog	1	PK_ReplayErrorLog	CLUSTERED	7733	1	NONE	60.76	0.11	60.87	1091780359	7205761390110310	false		1
Test Database 1	sync	MergeTables	1	PK_MergeTables	CLUSTERED	137	1	NONE	0.02	0.05	0.07	1059780245	72057638895050460	false		0
Test Database 1	sync	IdentifyRange	1	PK_IdentifyRange	CLUSTERED	126	1	NONE	0.07	0.13	0.2	1139786300	72057638896271360	false		0
Test Database 1	sync	MergeMailSettings	1	PK_MergeMailSettings	CLUSTERED	40	1	NONE	0.02	0.05	0.07	1011780074	72057613900040960	false		0
Test Database 1	sync	Settings	1	PK_Settings	CLUSTERED	3	1	NONE	0.02	0.05	0.07	1237079447	72057694043564030	false		0
Test Database 1	dbo	AA	0		HEAP	1	1	NONE	0.02	0.05	0.07	531813389	72057653145042940	false		0
Test Database 1	dbo	BB	1	PK_BB_3DD019E30F9AC8B	CLUSTERED	1	1	NONE	0.02	0.05	0.07	499813275	72057653144977400	false		0

Showing 1 to 10 of 16 entries

Previous 1 2 3 4 5 6 Next

Index Compression Settings of Test Database 1

Time Collected: 1/3/2021 12:30 am

No index compression settings available

SQL CONFIGURATION:

This section provides an overview of current SQL Configuration settings.

SQL Configuration
EXPAND

Click "Expand" to see more details as shown below.

Server Description 1

Test Server 1
Test Database 1

Summary
Core Tables
Managed Rerouting
Data Filtering
Reporting

SQL Configuration

Time Collected: 1/3/2021 12:30 am

Search

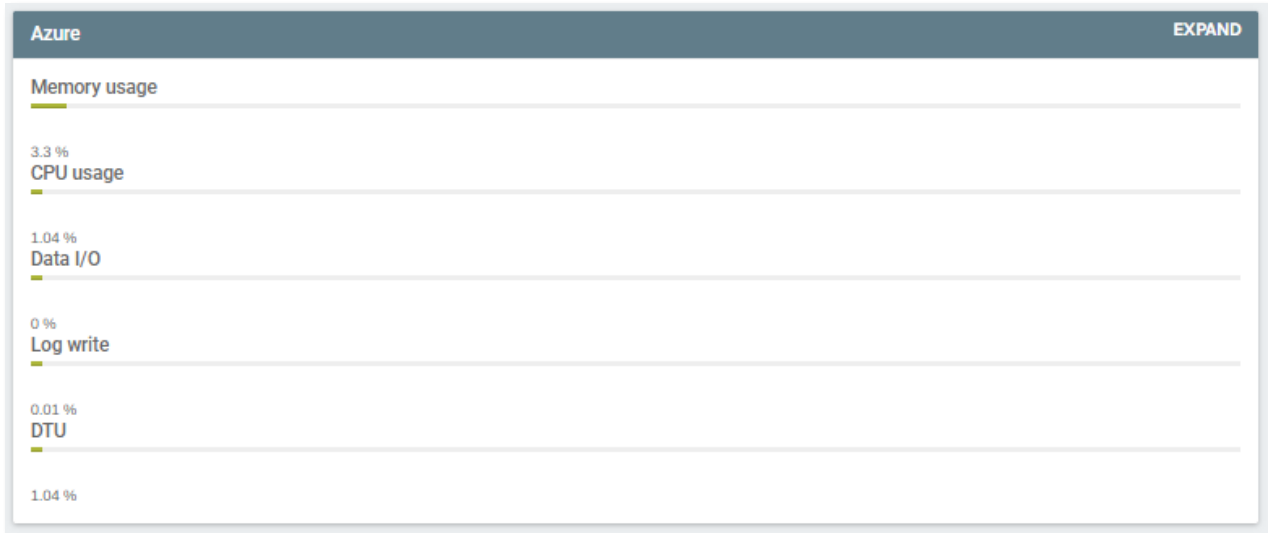
NAME	DESCRIPTION	VALUE	ACTIVE	DEFAULT (MIN - MAX)
ADR cleaner retry timeout (min)	ADR cleaner retry timeout	0	0	(0 - 32767)
ADR Prefillcation Factor	ADR Prefillcation Factor	0	0	(0 - 32767)
allow filesystem enumeration	Allow enumeration of filesystem	1	1	(0 - 1)
blocked process threshold (s)	Blocked process reporting threshold	20	20	(0 (0 - 36400)
clr strict security	CLR strict security enabled in the server	1	1	(0 (0 - 1)
clr enabled	CLR user code execution enabled in the server	1	1	(0 (0 - 1)
polybase enabled	Configure SQL Server to connect to external data sources through Polybase	0	0	(0 - 1)
opensearch auto_create_statistics	Enable or disable auto create statistics for opensearch sources.	1	1	(0 - 1)
default trace enabled	Enable or disable the default trace	0	0	1 (0 - 1)
suppress recovery model errors	Return warning instead of error for unsupported ALTER DATABASE SET RECOVERY command	0	0	(0 - 1)
Data processed daily limit in TB	SQL On-demand data processed daily limit in TB	2147483647	2147483647	(0 - 2147483647)
Data processed monthly limit in TB	SQL On-demand data processed monthly limit in TB	2147483647	2147483647	(0 - 2147483647)
Data processed weekly limit in TB	SQL On-demand data processed weekly limit in TB	2147483647	2147483647	(0 - 2147483647)
tempdb metadata memory-optimized	tempdb metadata memory-optimized is disabled by default.	0	0	(0 - 1)
column encryption enclave type	Type of enclave used for computations on encrypted columns	0	0	(0 - 2)
version high part of SQL Server	version high part of SQL Server that model database copied for	0	0	(2147483648 - 2147483647)
version low part of SQL Server	version low part of SQL Server that model database copied for	0	0	(-2147483648 - -2147483647)

Showing 1 to 17 of 17 entries

Previous 1 Next

AZURE:

This section provides a high level overview of your Server Statistics if your reporting database is hosted on Azure SQL DB (DaaS).



Click "Expand" to see more details and view your data through using an interactive chart as shown below.

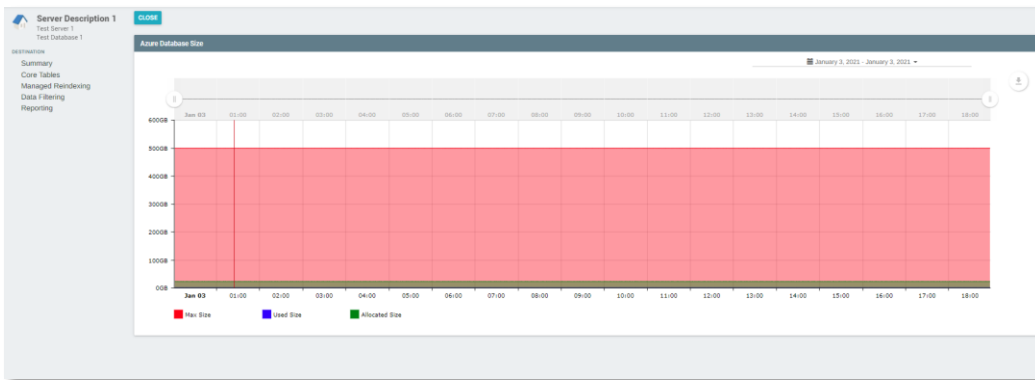


AZURE DATABASE SIZE:

This section provides an overview of the details related to the size of your database if your reporting database is hosted on Azure SQL DB (DaaS).

Azure Database Size		EXPAND
Database Size	1.6 GB	
Allocated Size	23.5 GB	
Max Size	500.0 GB	

Click "Expand" to see more details and view your data through using an interactive chart as shown below.

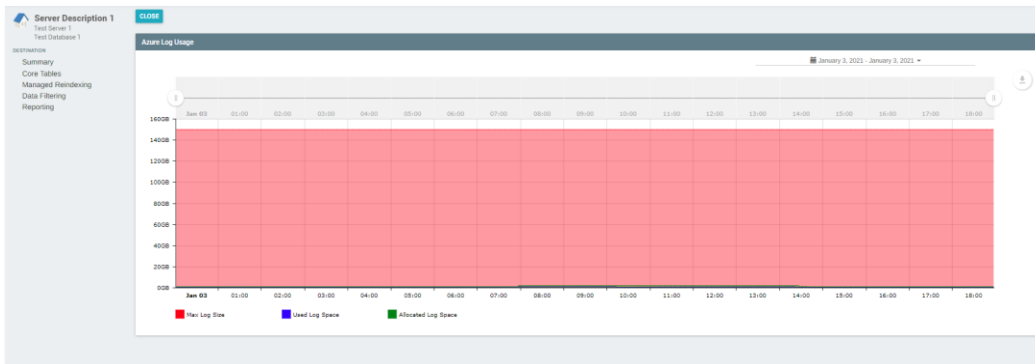


AZURE LOG USAGE:

This section provides an overview of information pertaining to your Azure log usage if your reporting database is hosted on Azure SQL DB (DaaS).

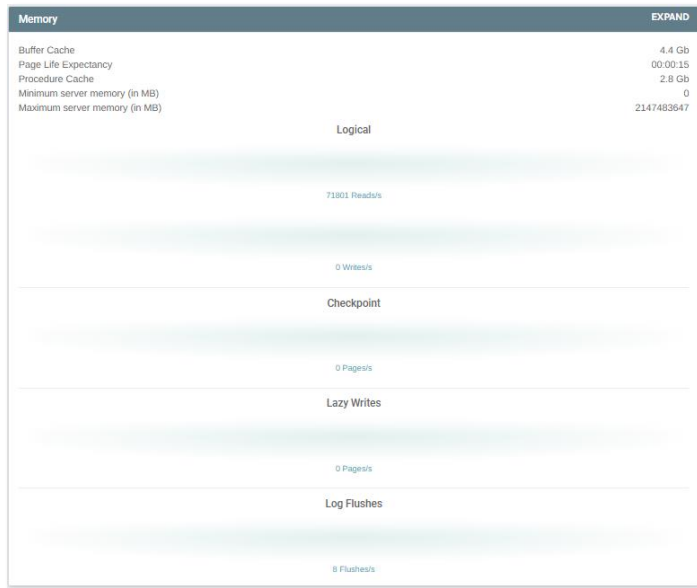
Azure Log Usage		EXPAND
Log Size	753.7 MB	
Allocated Log Size	1.1 GB	
Max Log Size	150.0 GB	

Click "Expand" to see more details and view your data through using an interactive chart as shown below.



MEMORY:

This section provides an overview of statistics related to the allocated memory for your server



Logical:

- **Reads:** The flow shows the rate at which pages in the buffer cache are being referenced by SQL connections (logical page reads).
- **Writes:** This flow represents data/index pages written to disk.

Checkpoint:

- **Pages:** This flow represents being written to disk by the Checkpoint Process.

Lazy Writes:

- **Pages:** This flow represents the number of pages per second that the lazy writer flushing to disk.

Log Flushes:

- **Flushes:** This flow represents the log pages being written to disk by the Log Writer process.

Click "Expand" to see more details and view your data through using an interactive chart as shown below.

