# SQL Server 2016 In-Memory Performance Demo in Retail/eCommerce Scenario

## SQL Server 2016 In-Memory OLTP, Columnstore and Real-Time Operational Analytics

Target audience: Microsoft Field Data Platform Sellers and other field roles (TSP, SSP, DPSA, CSA, PFE etc)

Published: June 2016 (v1)

Owner: Marko Hotti, Sr. Technical Product Manager SQL Server (markohot@microsoft.com)

# About this demo

## Goals

The goal of this demo is to showcase the in-memory features – In-Memory OLTP, Clustered Columnstore Index and Real-Time Operational Analytics in SQL Server 2016. The required components for this demo have been pre-installed and pre-configured for you and are explained in this document.

## Scenario

This demo tells the story of how the retailer Northwind Traders is able to increase store profits, improve reporting performance and gain real-time insights into the data. The scenario can be equally applied to other industries and organizations.

In-Memory Tables can offer extreme OLTP performance powered by its latch-free structure and memory-resident data. A successful implementation of In-Memory technologies can help Northwind Traders increase it revenue by thousands of dollars.

Columnstore indexes can offer blazing fast performance due to its redesigned storage, compression and processing technology.

SQL Server 2016 Operational Analytics can be leveraged for real-time analysis. The columnstore index on an in-memory table allows operational analytics by integrating in-memory OLTP and in-memory columnstore technologies to deliver high performance for both OLTP and analytics workloads.

## Features showcased

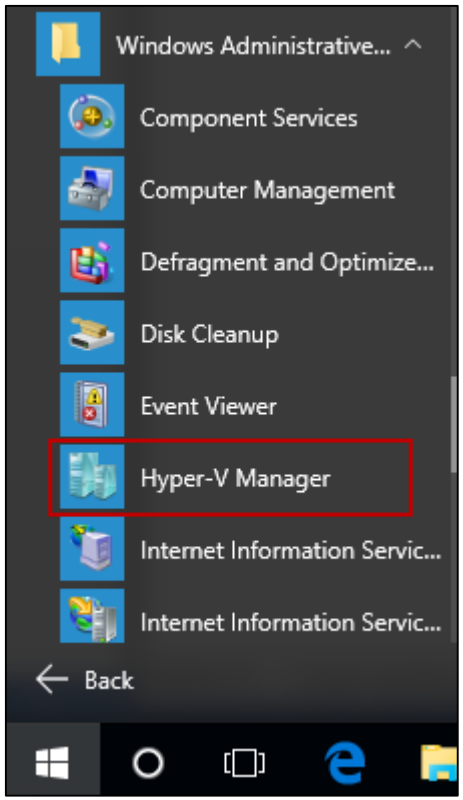
This following features are showcased in this demo

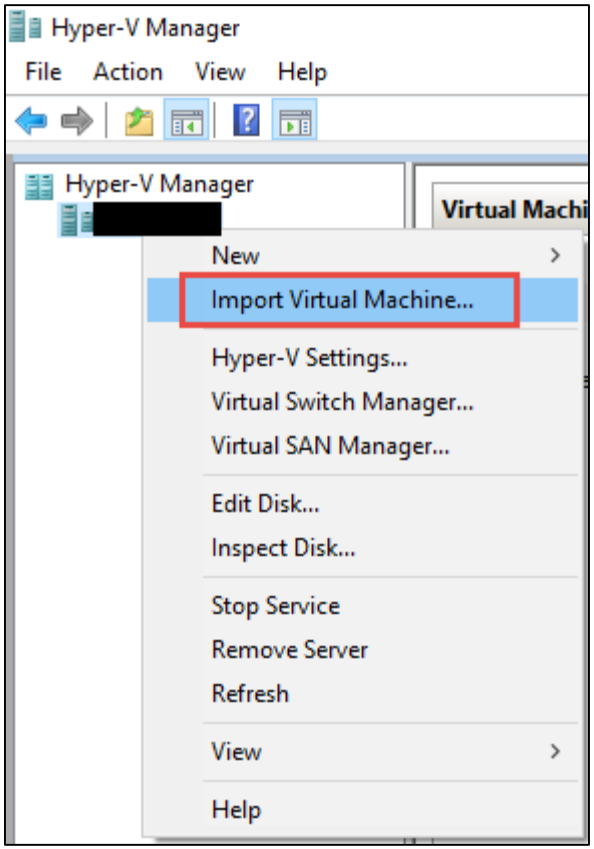
- In-Memory OLTP.
- Clustered Columnstore Index.
- Real-Time Operational Analytics.

# Contents

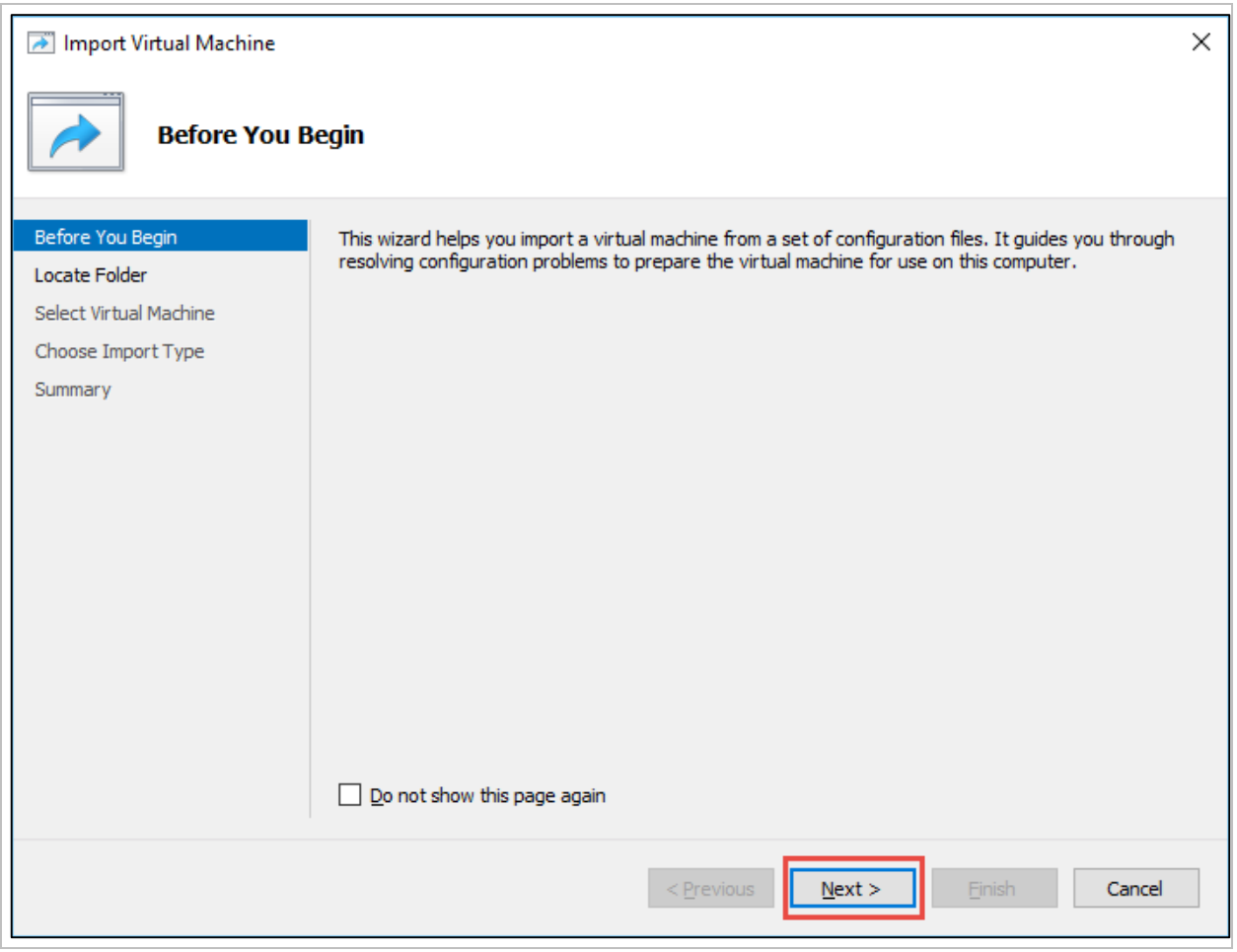
# Setup

## VM Setup

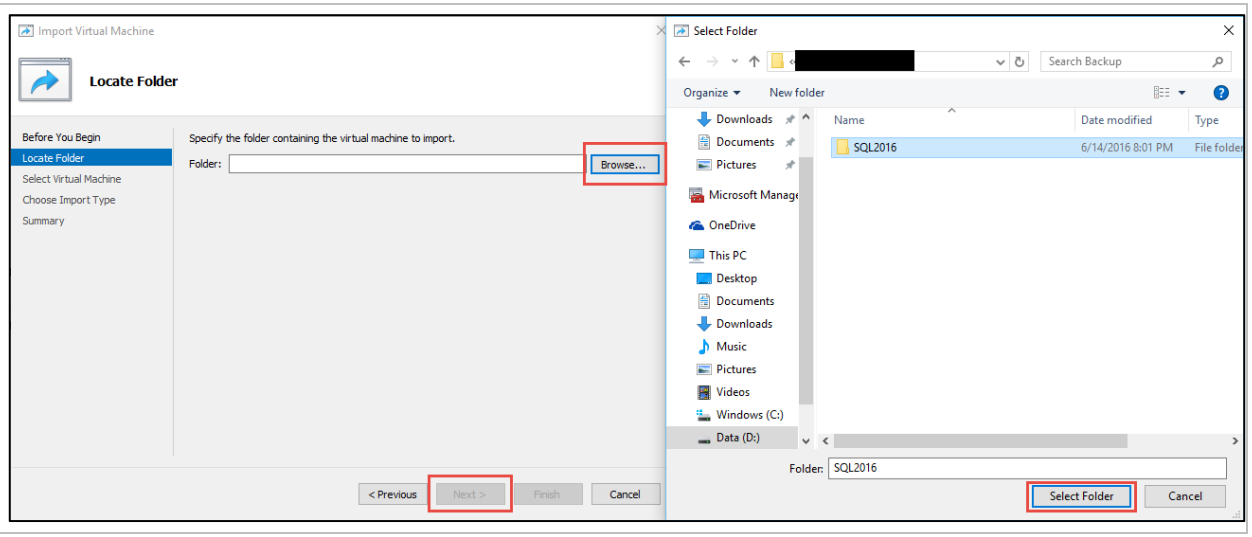
| Steps | Screenshot |
|---|---|
| 1. Download/copy the compressed demo VM (15GB) using one of the URLs: | https://sql2016demoforeastasia.blob.core.windows.net/sql2016inmemoryperfdemo/SQL2016.rar<br>https://sql2016demoforeurope.blob.core.windows.net/sql2016inmemoryperfdemo/SQL2016.rar<br>https://sql2016demoforwestusa.blob.core.windows.net/sql2016inmemoryperfdemo/SQL2016.rar |
| 2. Open **Start** -> **All Apps** -> **Windows Administrative Tools** -> **Hyper-V Manager**. |  |

| | |
|---|---|
| 3. In Hyper-V Manager, right click on the machine name and select **Import Virtual Machine…** |  |

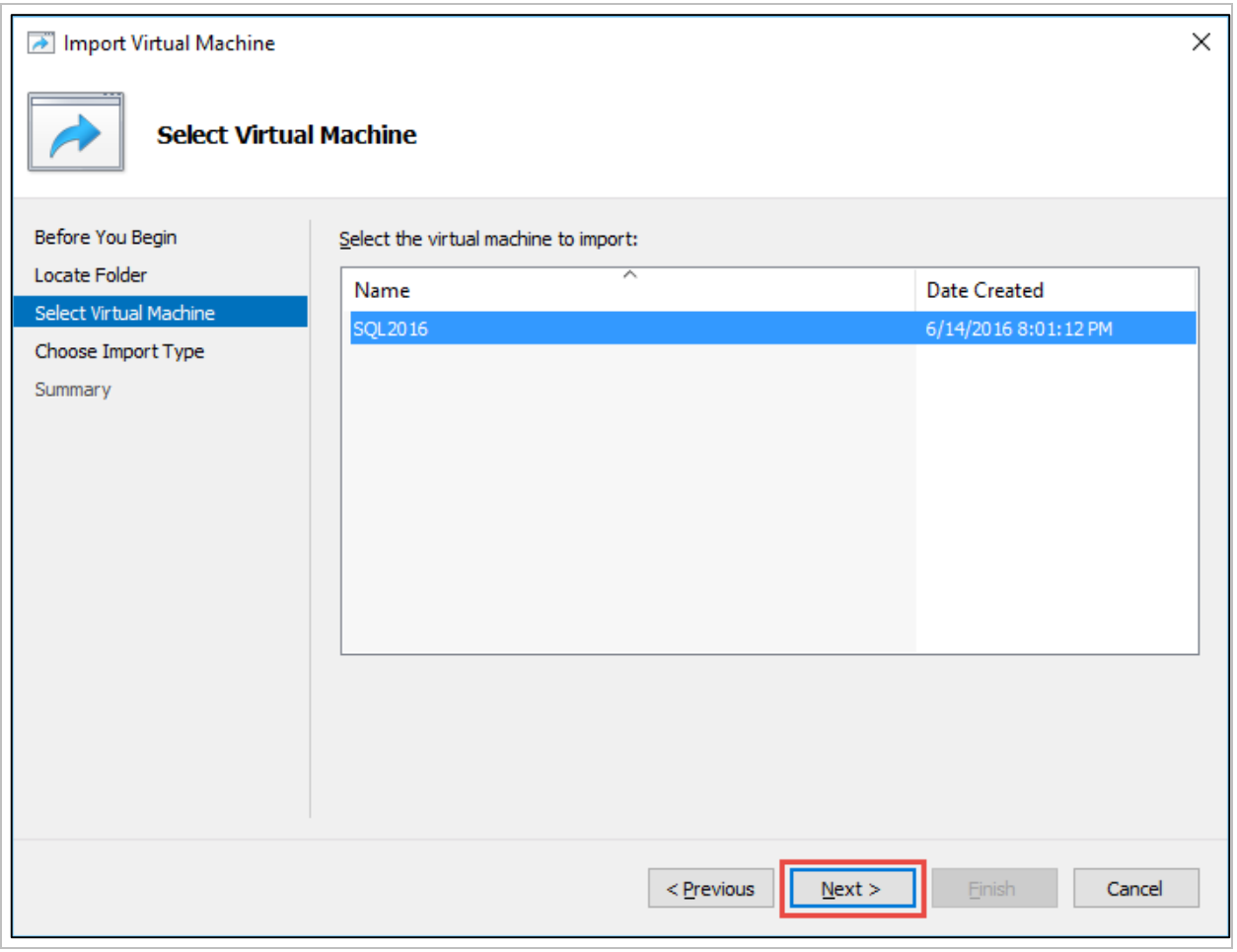
4. In the **Import Virtual Machine** window, click **Next**.



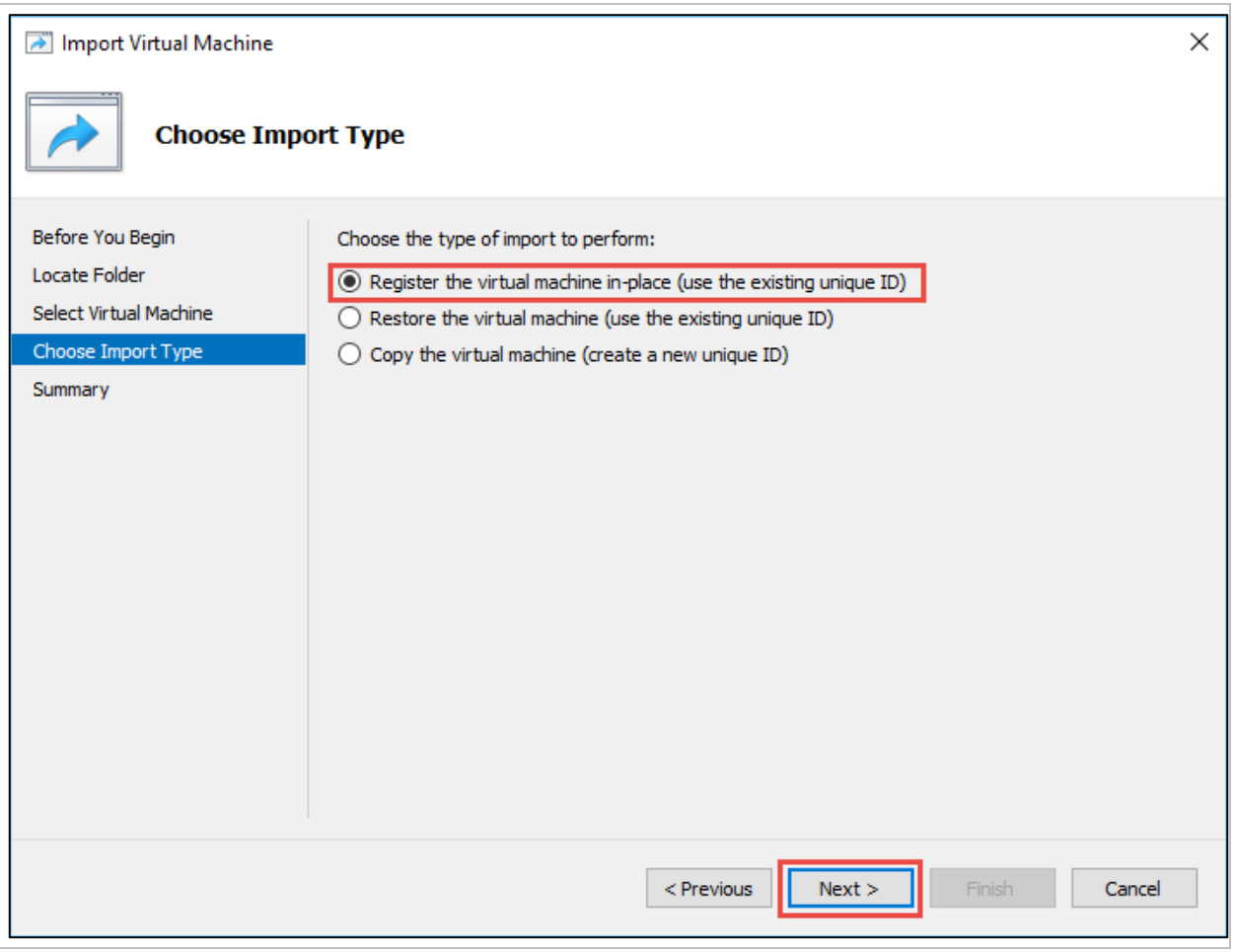
5. In the **Locate Folder** page, click **Browse**.
6. In the **Select Folder** window browse to the VM folder and select **SQL2016** VM folder and click **Select Folder**.
7. In Locate Folder page, click **Next**



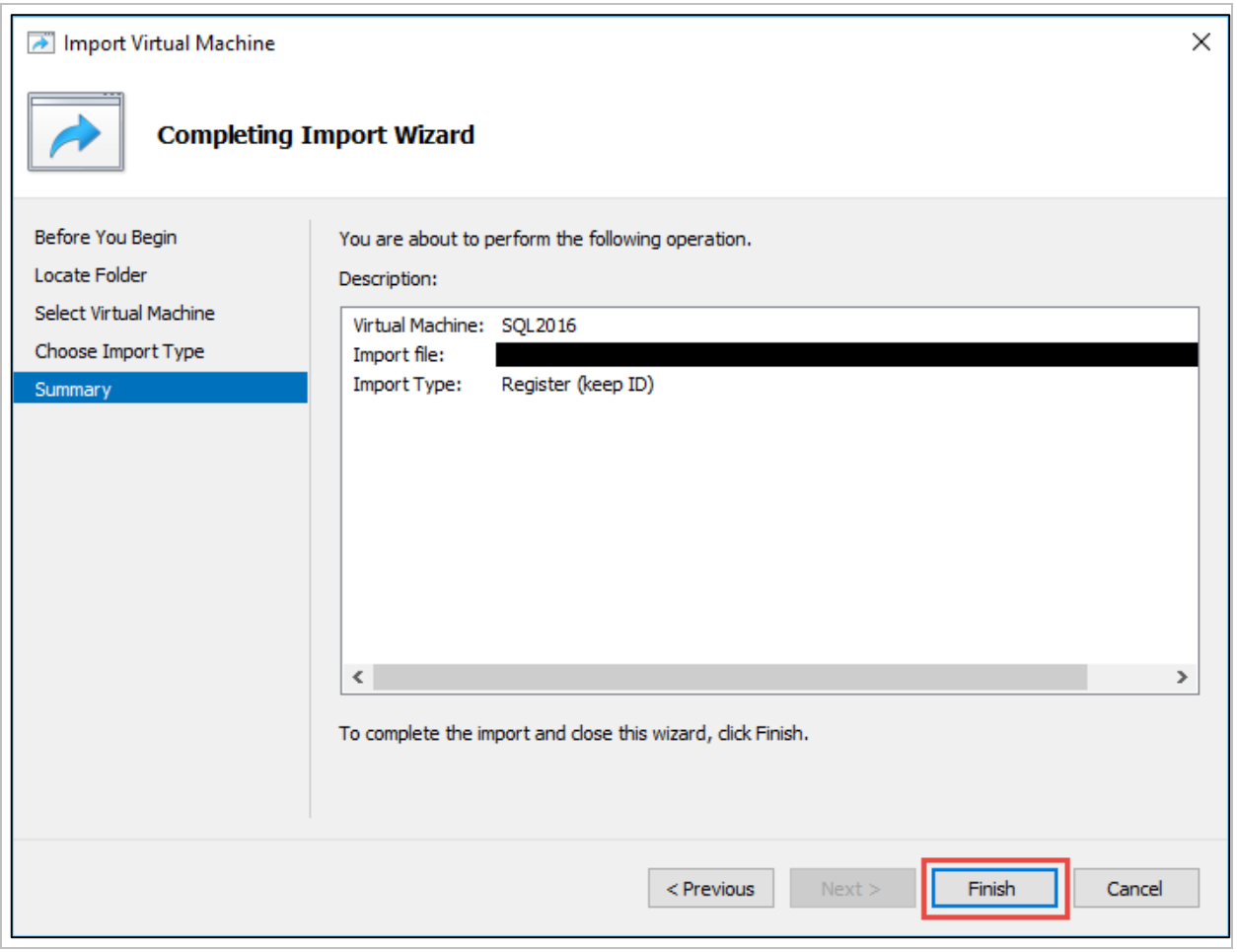
8. In the **Select Virtual Machine** page, leave the default selection and click **Next**.



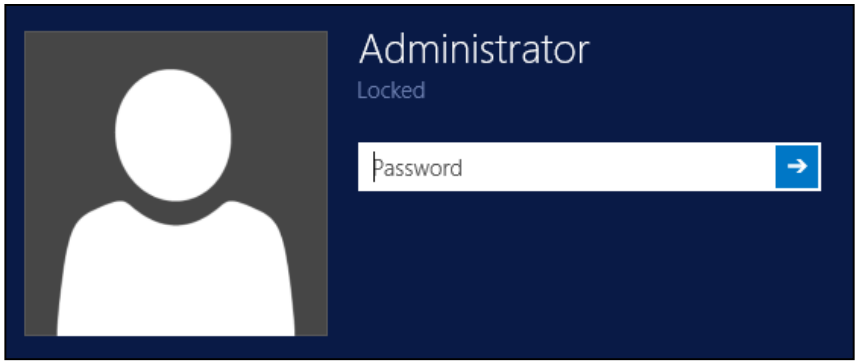
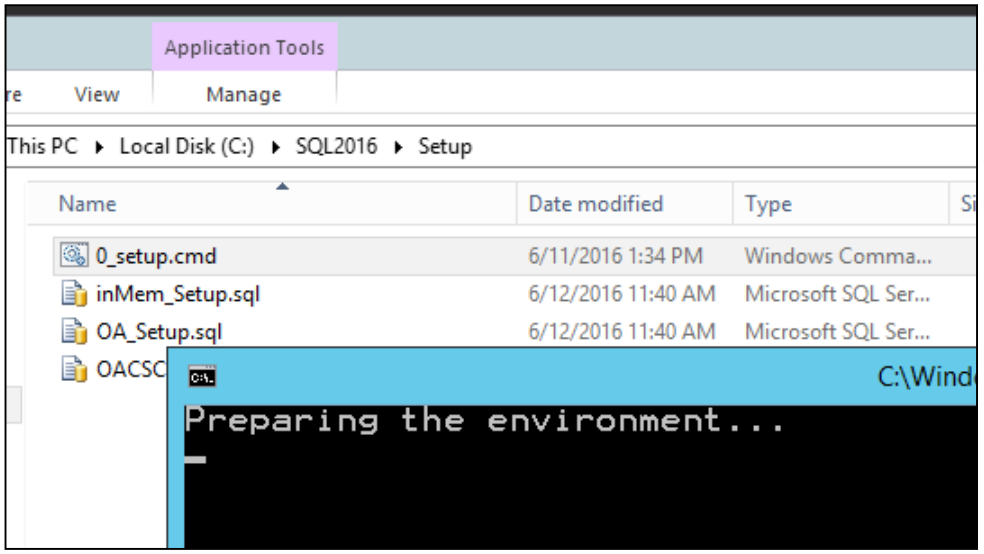
9. In the **Choose Import Type**, select **Register the virtual machine in-place (use the existing unique ID)** and click **Next**.



10. In **Summary** page, click **Finish** to import the VM.
11. Check the Virtual Machine settings. **The recommended sizing is 12 GB of RAM and 6 cores.**
12. Note. Optionally you can convert the VHDX file to a fixed sized VHD file, upload the VHD to your Azure Subscription and create an Azure VM which uses the attached VHD disk as system disk.

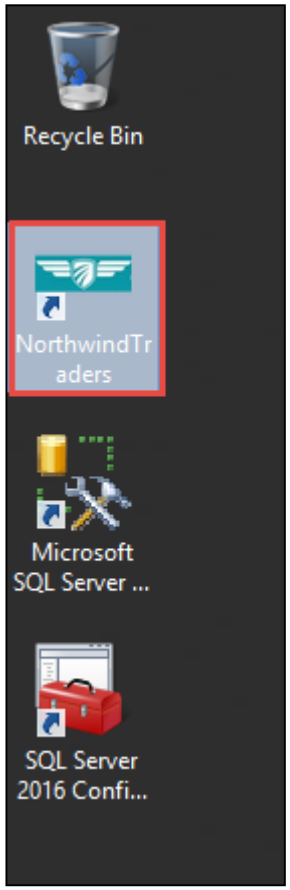


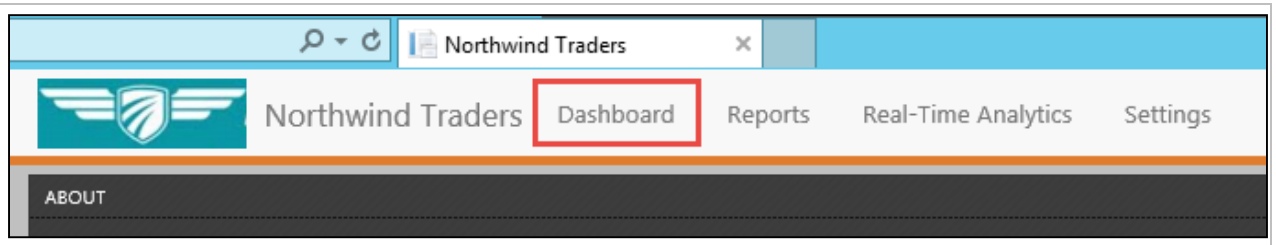
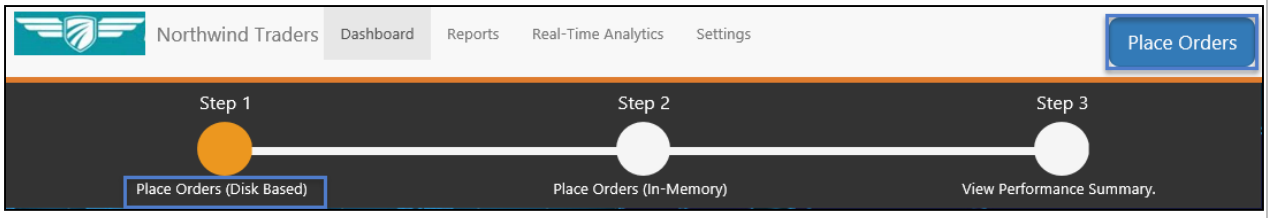
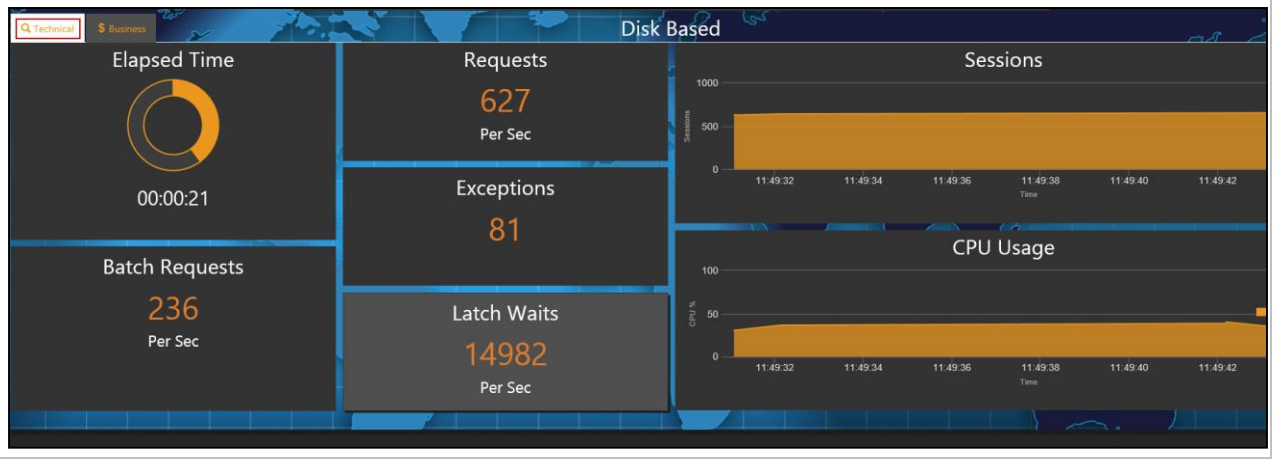
## Demo Setup

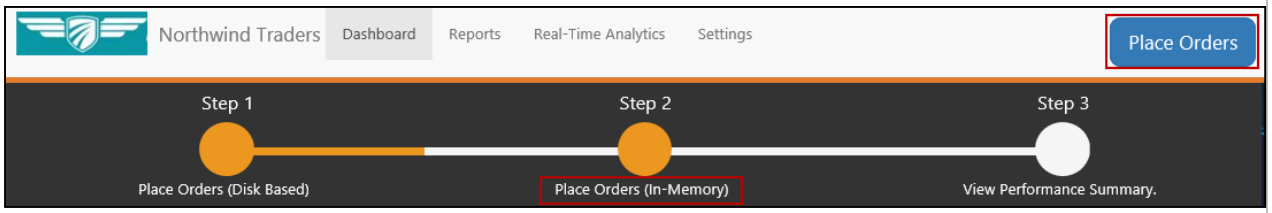
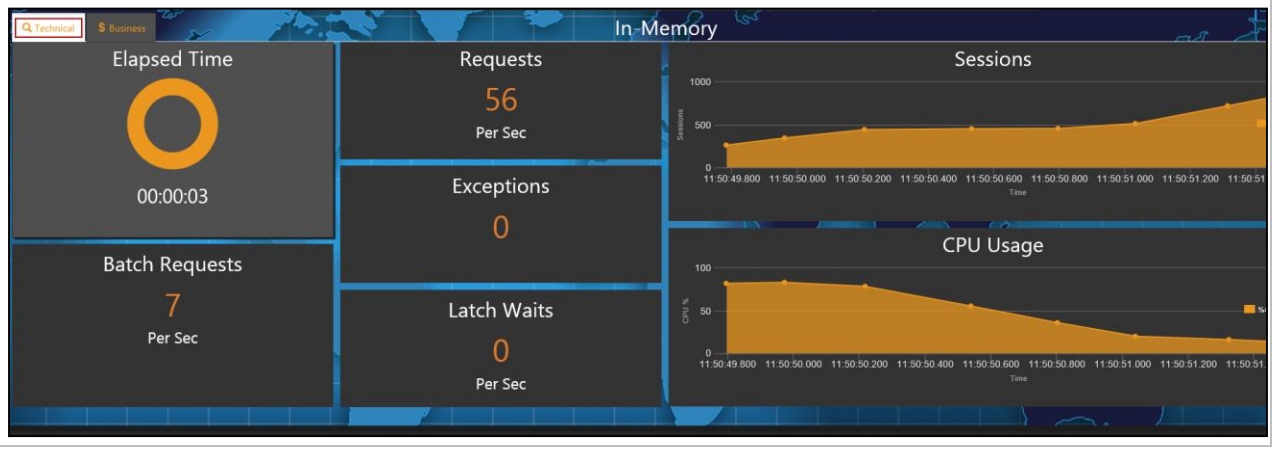
| Steps | Screenshot |
|---|---|
| 1. Sign in to the VM<br><br>Username: Administrator<br>Password: Passw0rd |  |
| 2. Open File Explorer and browse to C:\SQL2016\Setup folder. | |
| 3. Double click on the file 0_setup.cmd to run it. |  |
| 4. After the 0_setup.cmd is executed, close File Explorer. | |

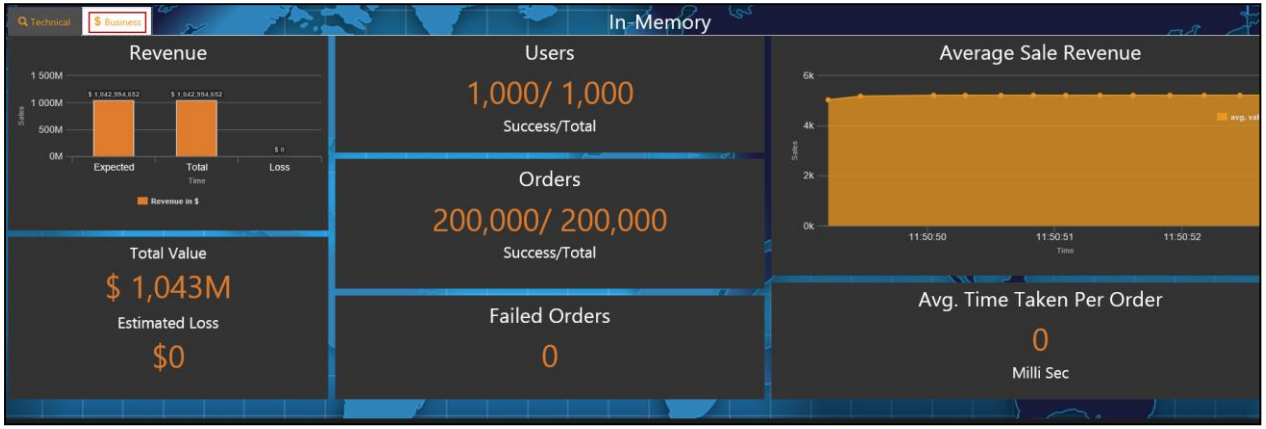
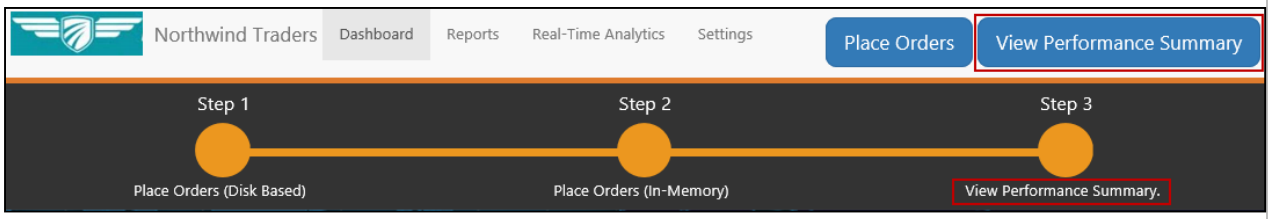
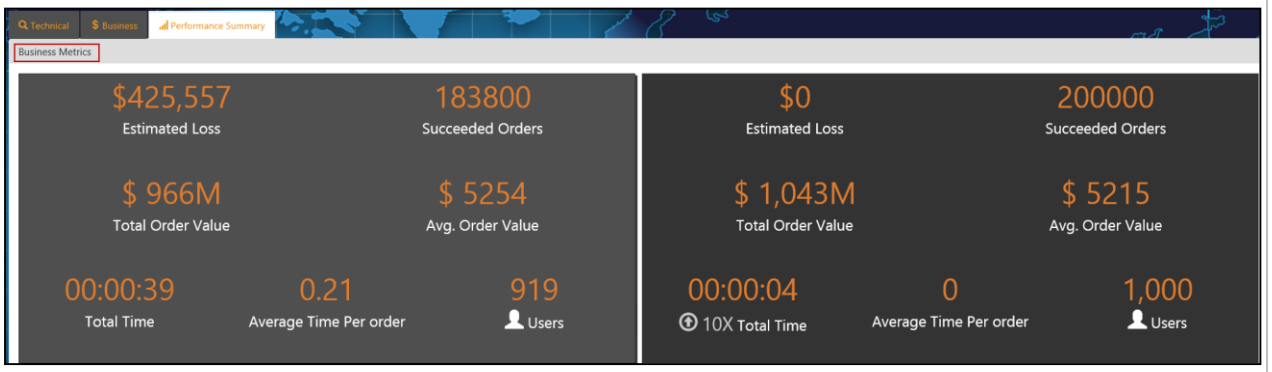
# Demo steps

## In-Memory OLTP

| Steps | Screenshot |
| --- | --- |
| 1. On the desktop Double click on **Northwind Traders**. This will launch the Northwind Traders home page. |  |

| | |
|---|---|
| 2. In the web browser, in the top panel click on Dashboard tab to go to In-Memory OLTP dashboard.<br><br>*Note: Do not refresh or click on any other tabs in the panel till this demo is finished. Sequence of steps should be followed.* |  |
| 3. In the top panel click on Place Orders to run the place orders workload on the disk-based table |  |
| 4. While the workload is running explain the technical metrics in the **Technical** tab.<br><br>*Note: Show that the latch waits is very high, batch requests per second is low. There are exceptions and the time taken is high.*<br>*The CPU usage is low for a longer duration.* |  |

| | |
|---|---|
| 5. Click on the **Business** tab and explain the business metrics.<br><br>*Note: Few orders failed and there is loss in revenue.* |  |
| 6. In the top panel, click on Place Orders to run the place orders workload on memory optimized table. |  |
| 7. While the workload is running explain the technical metrics in the Technical tab.<br><br>*Note: Show that the latch waits is very low, batch requests per second is high. There will be 0 exceptions and the time taken is very less.*<br>*The CPU usage is high for a very less duration.* |  |

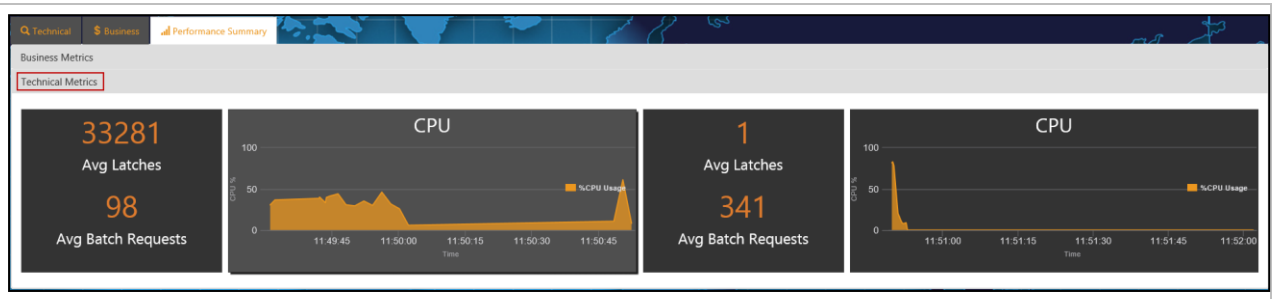
| | |
|---|---|
| 8. Click on the **Business** tab to see the business metrics.<br><br>*Note: All orders have succeeded with no loss.* |  |
| 9. After the workload is completed, click on **View Performance Summary** to open the performance summary tab. |  |
| 10. In the performance summary tab, click the **Business Metrics** to expand the business metrics comparison.<br><br>*Note 1: There is no loss of revenue with memory optimized tables.*<br><br>*Note 2: Show the improvement in time taken with memory optimized tables.* |  |

11. Click on **Technical Metrics** to expand technical metrics comparison.

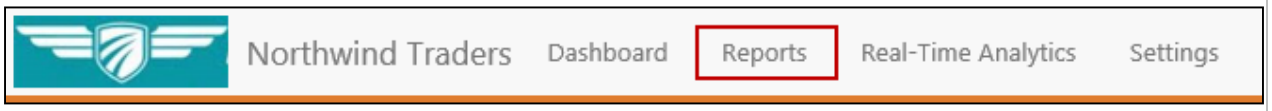
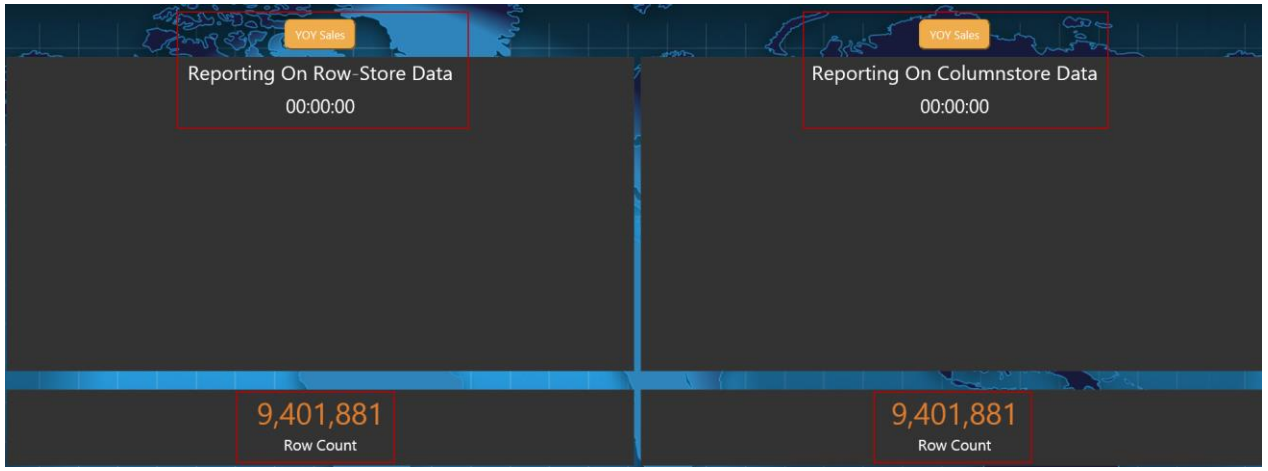
    *Note 1: The latch contention is higher with disk-based tables.*

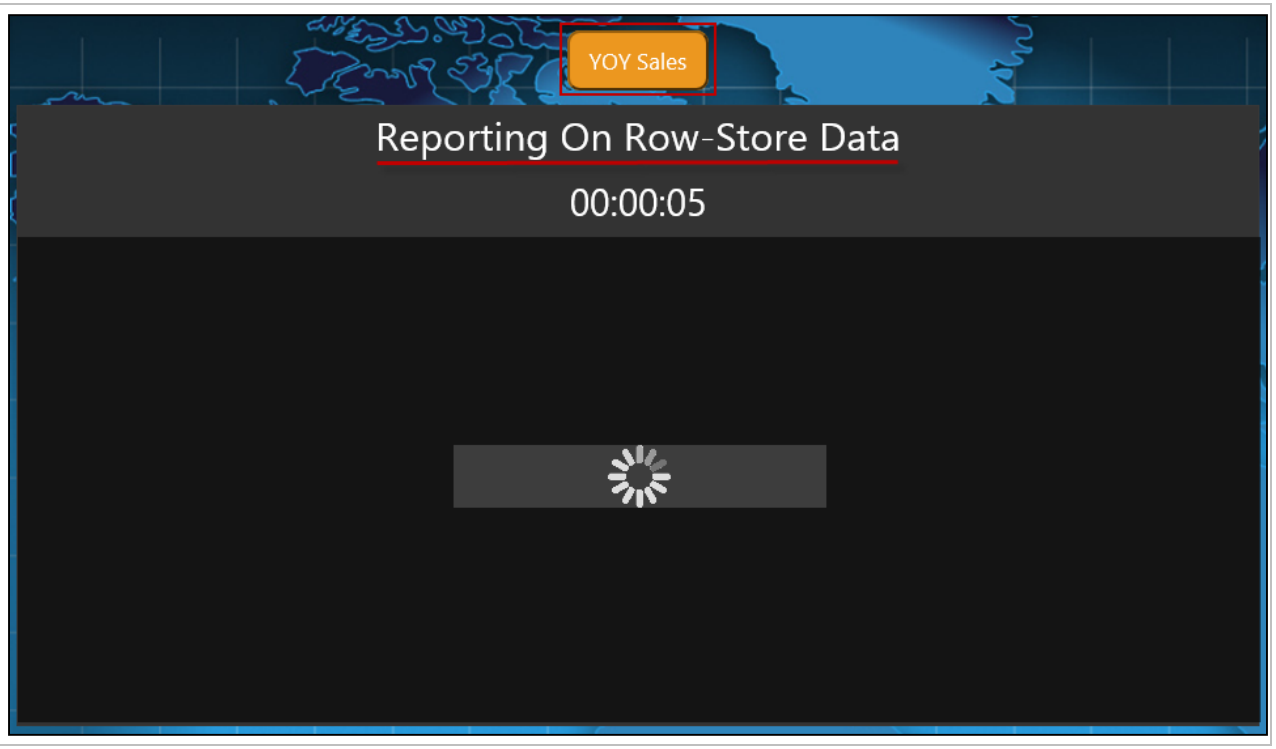
    *Note 2: The average batch requests per second is higher for In-Memory tables.*

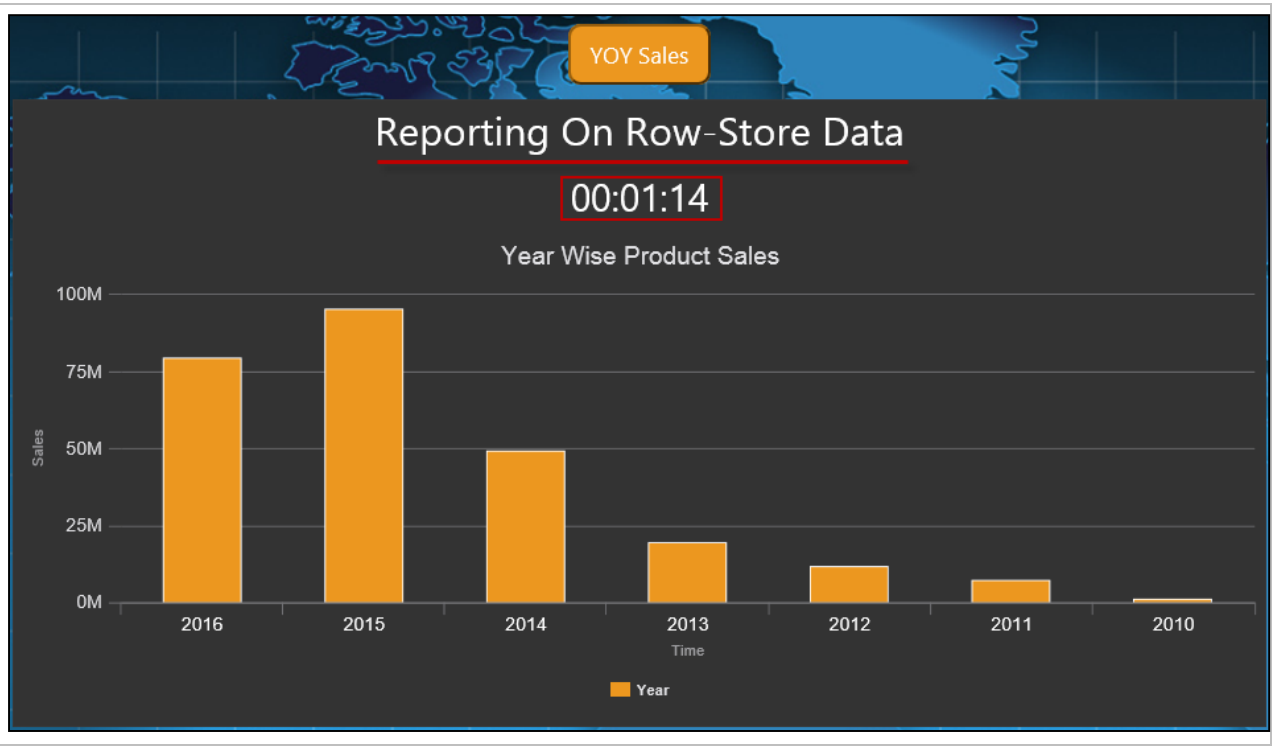
    *Note 3: The CPU is utilized efficiently during the workload execution for in-memory tables. The CPU utilization is low for disk-based table as more time is spent on IO.*



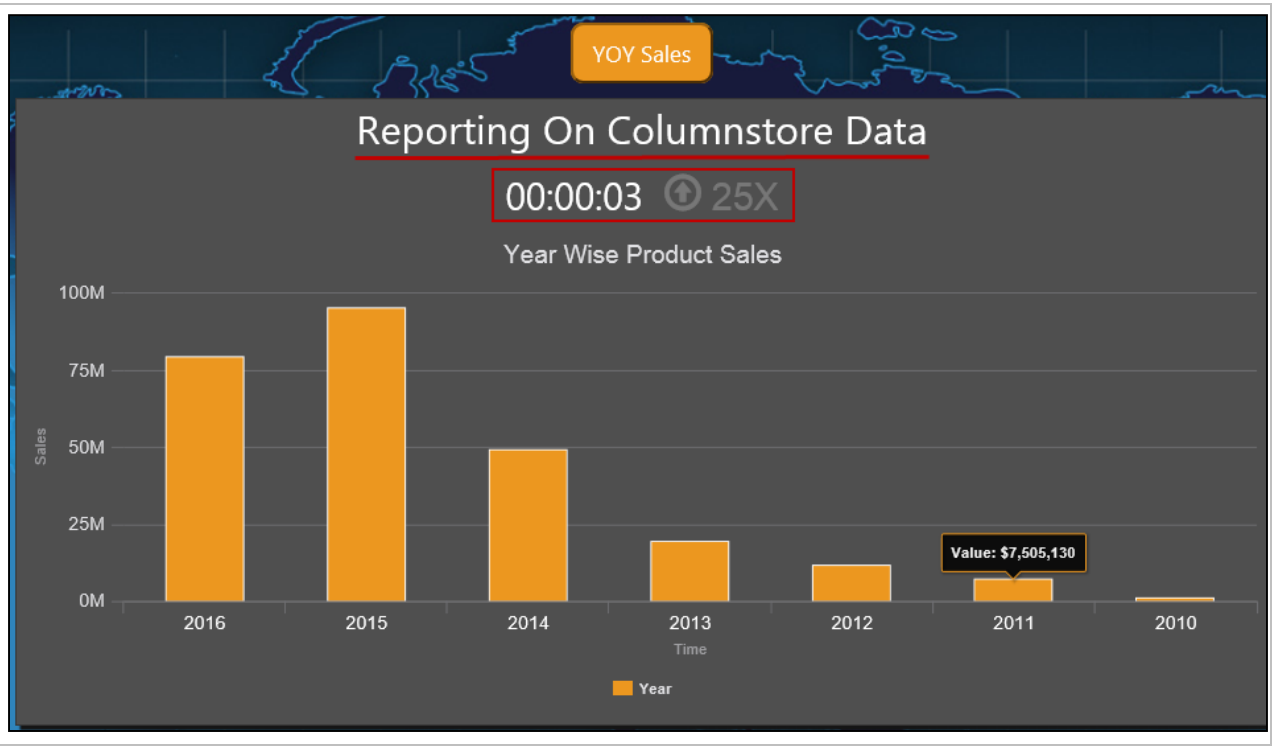
# Columnstore Indexes

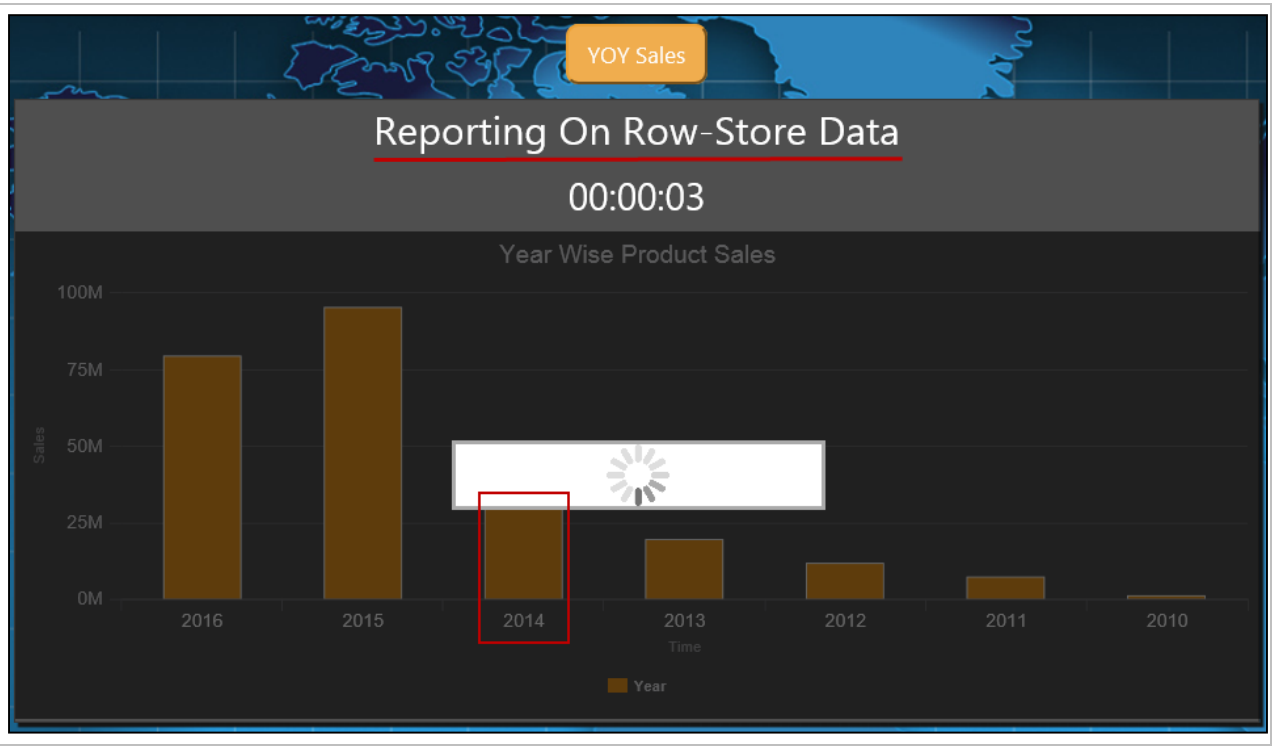
| Steps | Screenshot |
|---|---|
| 1. In the web browser, in the top panel click on **Reports** tab to go to Columnstore index demo page.<br><br>*Note: Do not refresh or click on any other tabs in the panel till this demo is finished. Sequence of steps should be followed.* |  |
| 2. The page is split into two parts. The left side pane is Reporting on Row-Store Data and the right side pane is Reporting on Columnstore Data.<br><br>*Note: The row counts of the tables used for this reports is displayed in the second set of panels.* |  |

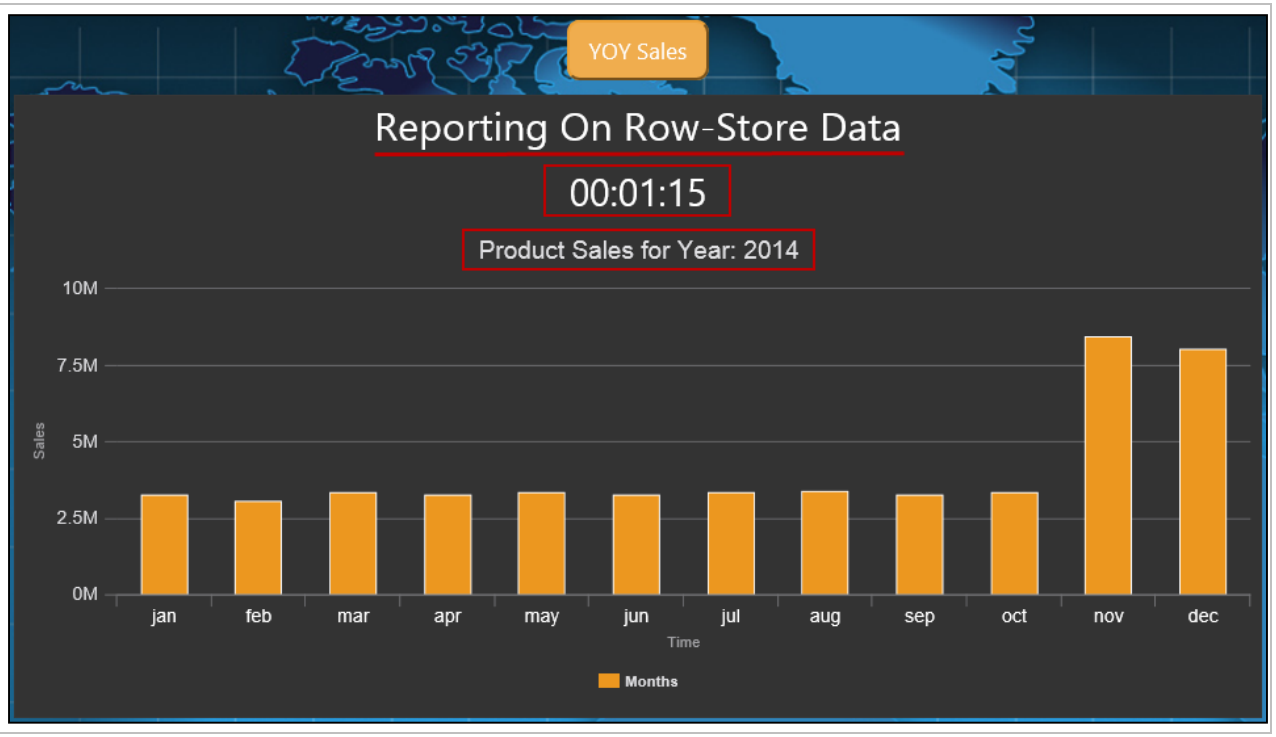
| | |
|---|---|
| 3. In the left side click on **YOYSales** button to run the year-on-year sales report on row-store data.<br><br>*Tip: While the report is executing the columnstore performance can be explained. Ref: https://msdn.microsoft.com/en-us/library/dn935005.aspx#Anchor_1* |  |

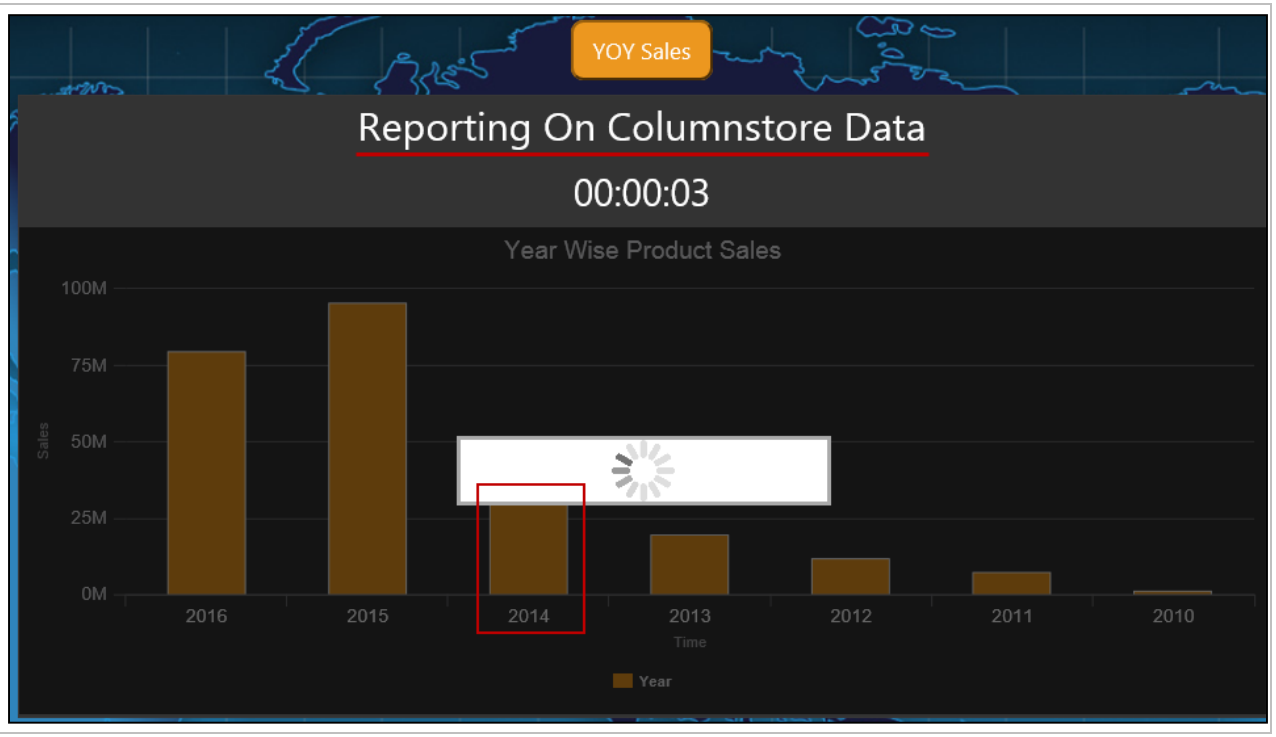
| | |
|---|---|
| 4. Once the report execution is completed the total time taken is displayed.<br><br>*Note: Wait for the report to complete execution before proceeding to next step.* |  |

| | |
|---|---|
| 5. In the right side click on **YOYSales** button to run the year-on-year sales report on columnstore data.<br><br>*Note: Click only after the YOY Sales report completes execution on row-store data.* |  |

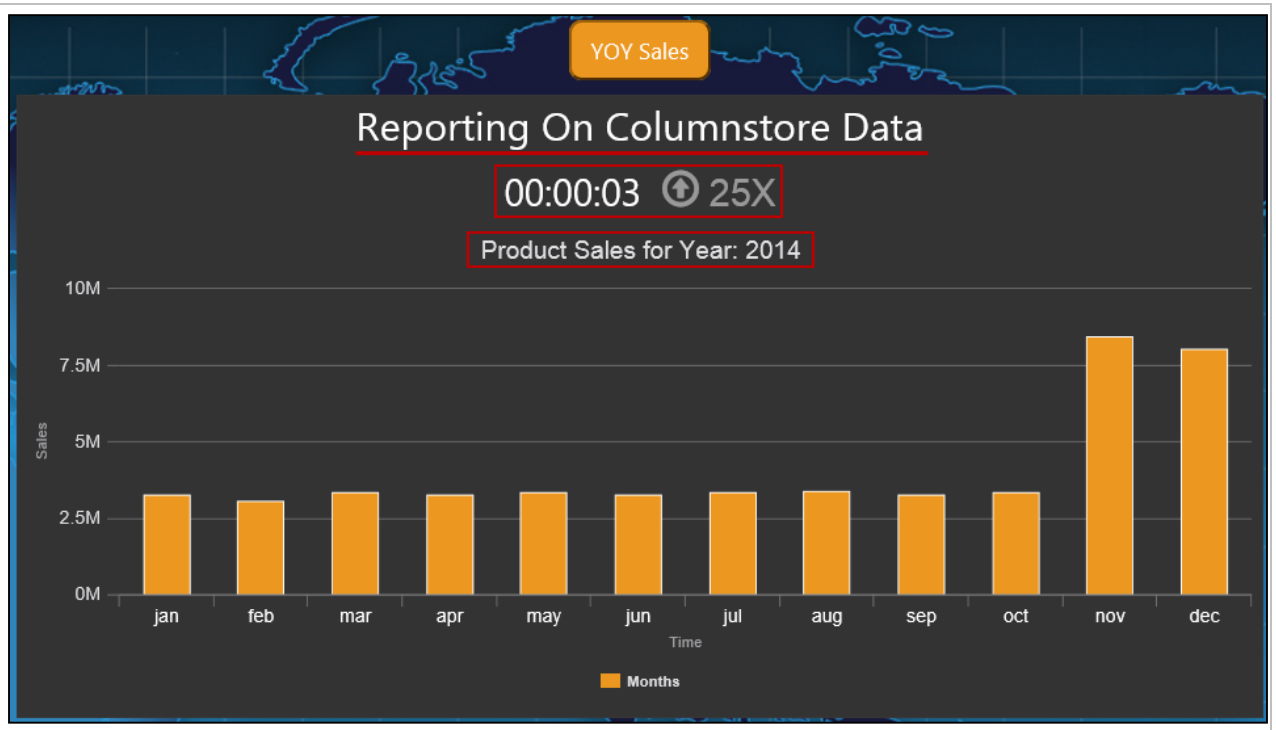
| | |
|---|---|
| 6. Once the report execution is completed the total time taken is displayed. The improvement is flashed beside the time taken (**25X**). |  |

| | |
|---|---|
| 7. In the row-store graph click on any year to drill down to monthly sales in the year.<br><br>*Note: Preferred year is 2013 or 2014 as this gives the best results.*<br><br>*Tip: While the report is executing the columnstore performance can be explained. Ref: https://msdn.microsoft.com/en-us/library/dn935005.aspx#Anchor_1* |  |

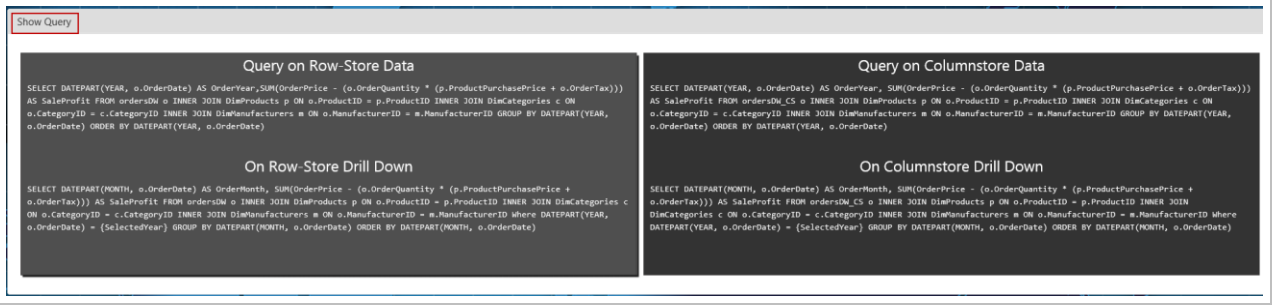
| | |
|---|---|
| 8. The monthly data for the year selected from row-store data will be displayed along with the total time taken.<br><br>*Note: Wait for the report to complete execution before proceeding to next step.* |  |

| | |
|---|---|
| 9. In the columnstore report graph, click on the same year which is selected in row-store graph.<br><br>*Note: Click only after the drill down report on row-store data is completed.* |  |

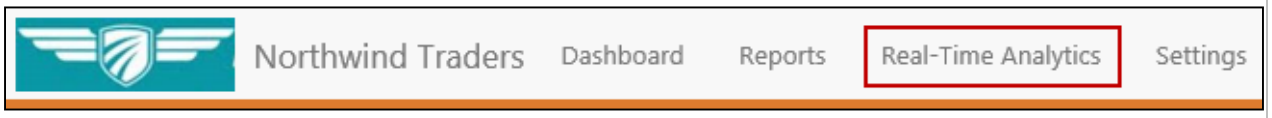
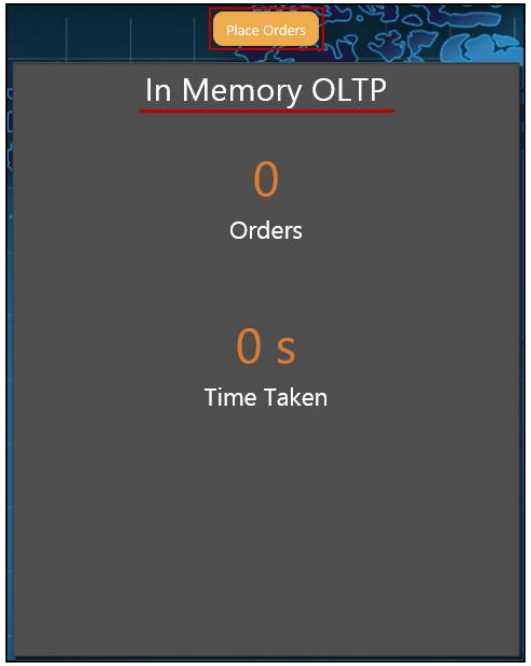
10. The monthly data for the year selected from columnstore data will be displayed along with the total time taken. The improvement is flashed beside the time taken (**25X**).

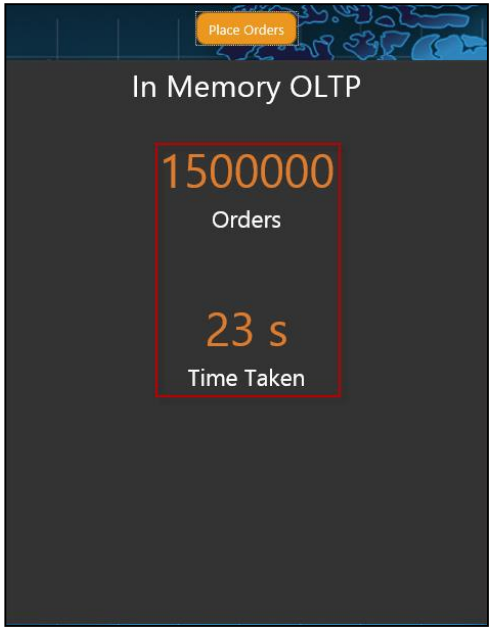


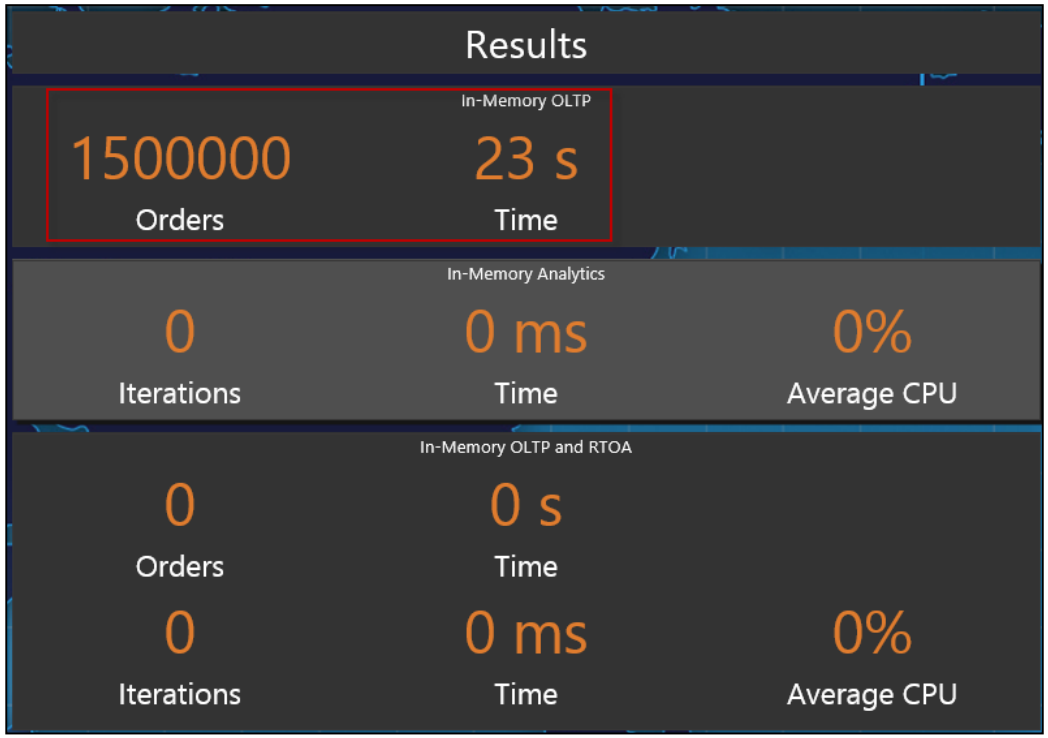

11. *Tip: If needed, click on Show Query to show the queries which are used for the year-on-year sales and drill down reports.*

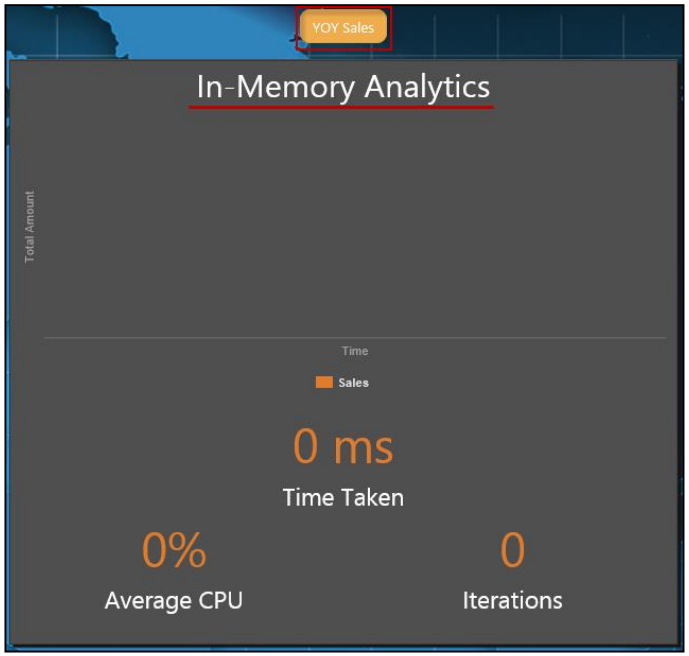


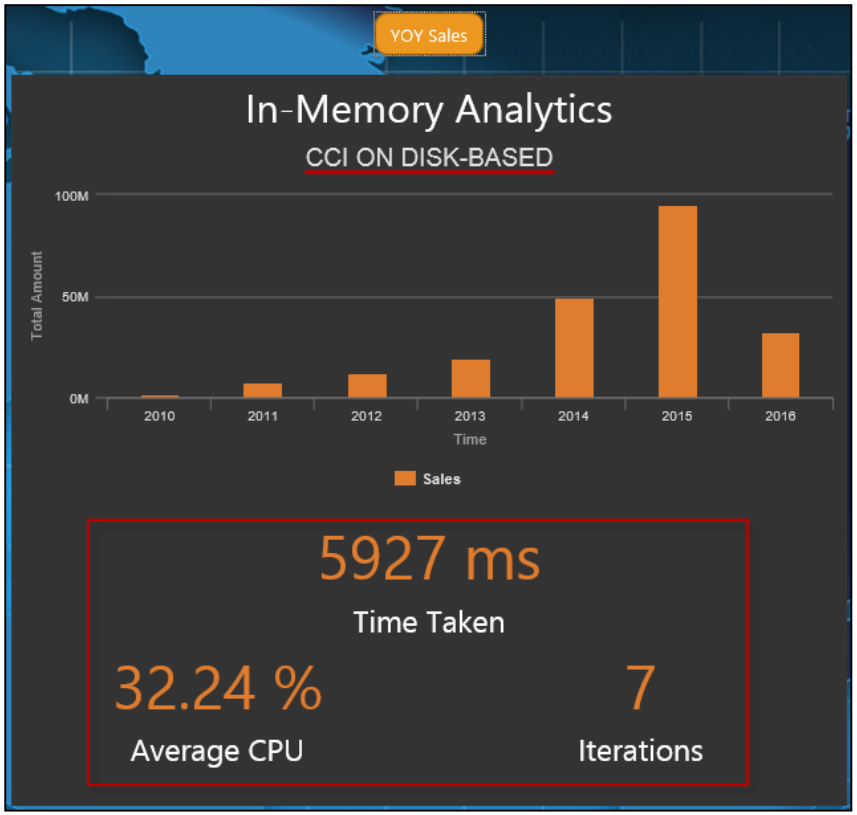
# Real-Time Operational Analytics

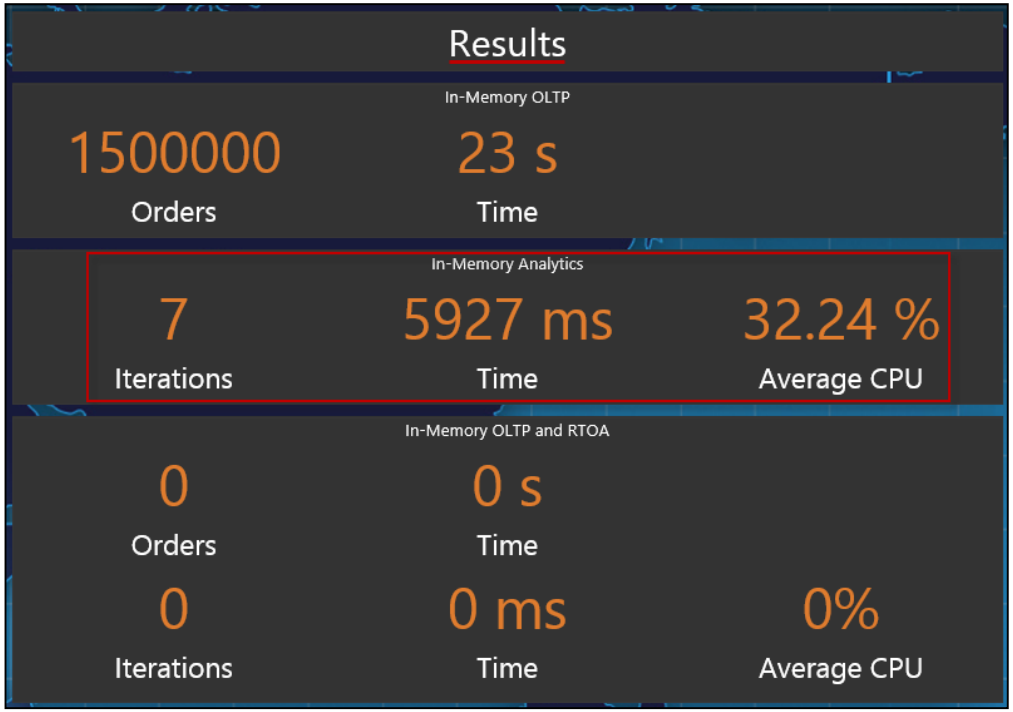
| Steps | Screenshot |
| --- | --- |
| 1. In the web browser, in the top panel click on **Real-Time Analytics** tab to go to Real-Time Operational Analytics demo page. |  |
| 2. In the first pane, click on **Place Orders** to insert orders into memory optimized table.<br><br>*Tip: The underlying table is a memory optimized table without any additional indexes.* |  |

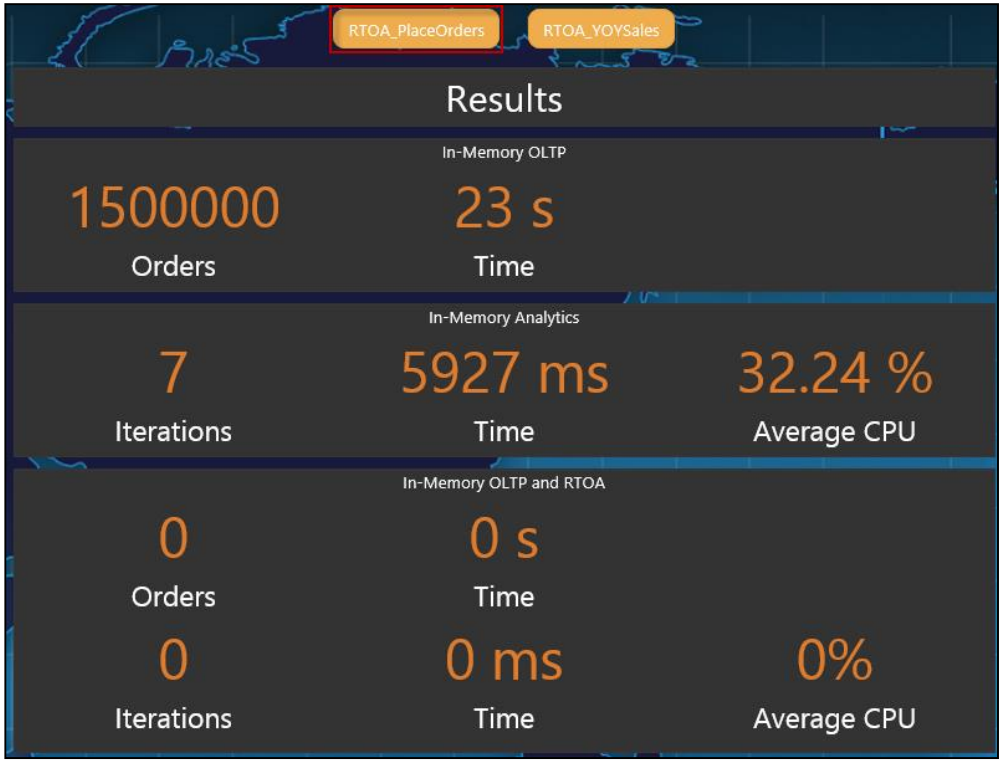
| | |
|---|---|
| 3. Once the orders are inserted the total time taken is displayed. |  |

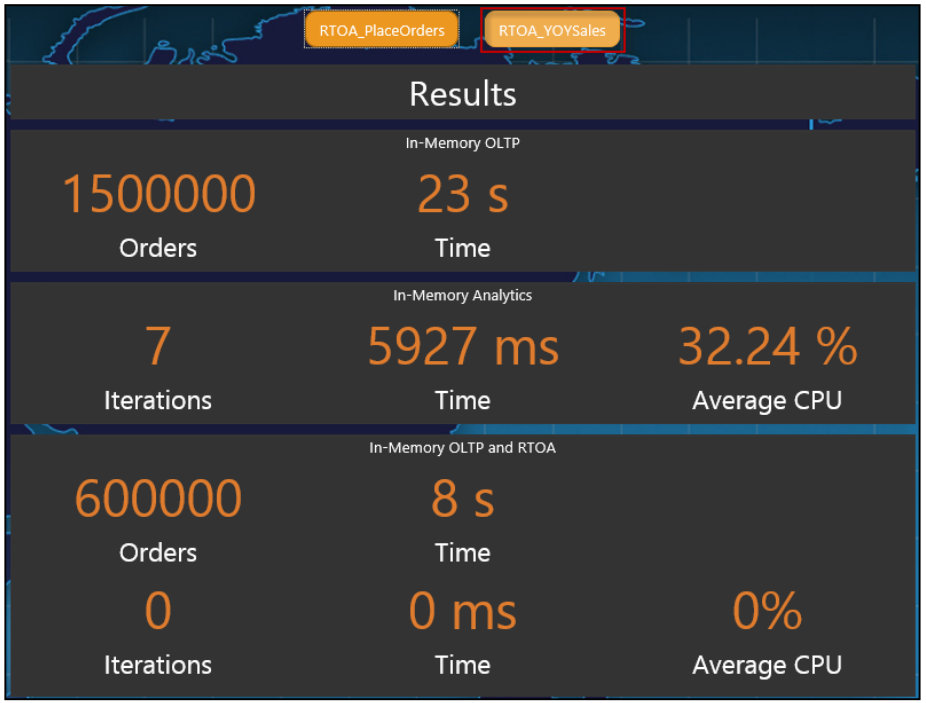
| | |
|---|---|
| 4. The results are recorded in the **In-Memory OLTP** panel in the third pane under the **Results**. |  |

| | |
|---|---|
| 5. In the second pane, click on **YOYSales** to run the year-on-year sales report from columnstore index on disk-based table.<br><br>*Tip: The underlying table is a disk-based table where the data is loaded with ETL job. There is a clustered columnstore index on the table.* |  |

| | |
|---|---|
| 6. The report is executed in multiple **iterations**. The **Time Taken** and the **Average CPU** usage is displayed.<br><br>*Note: Hover on year 2016 to show the sale amount of the year. Note this value and compare it with real-time value shown in step 11.* |  |

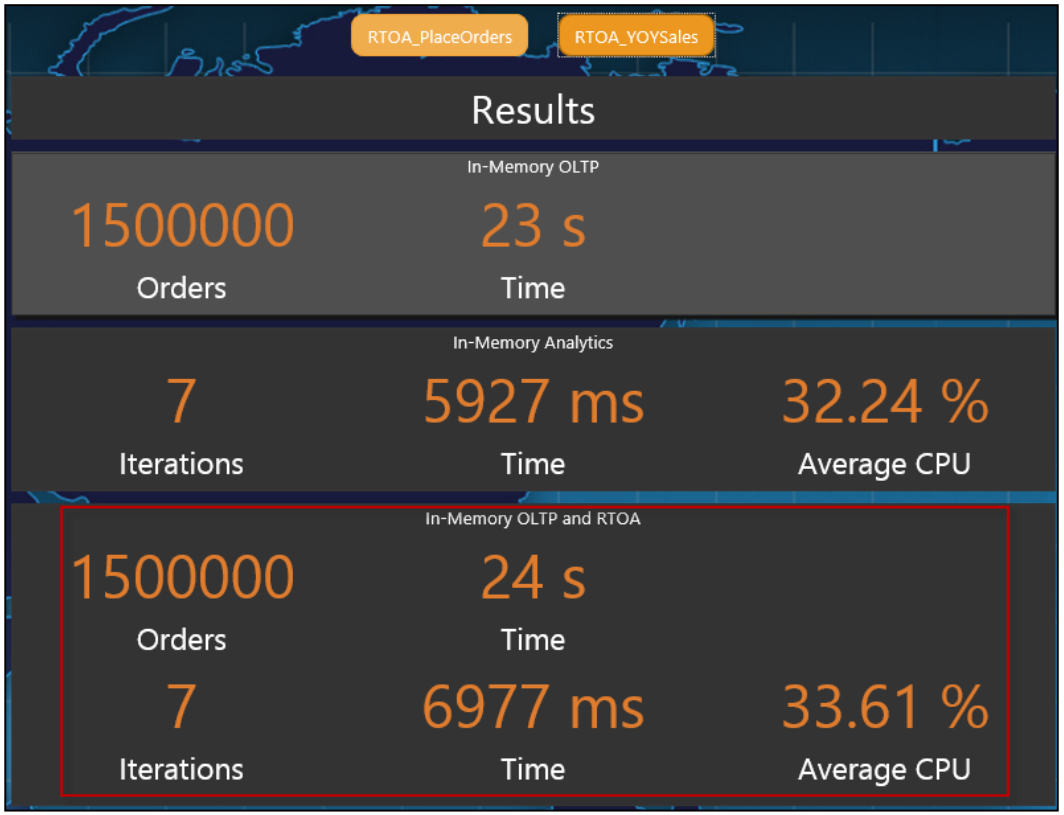
| | |
|---|---|
| 7. The In-Memory Analytics results are recorded in **In-Memory Analytics** panel under **Results** pane. |  |

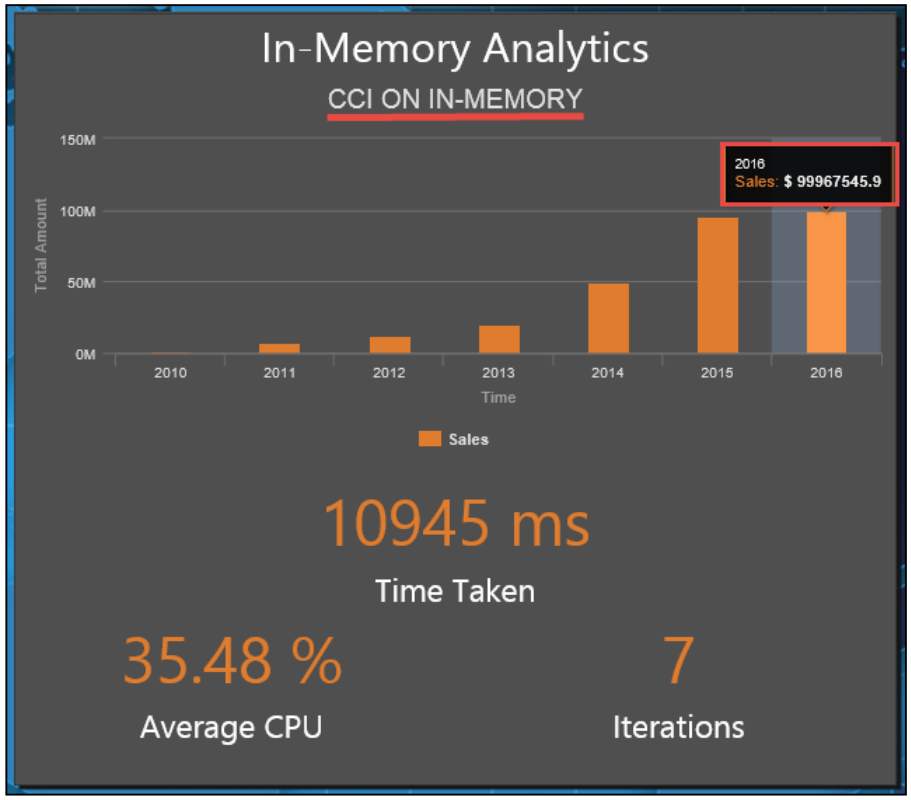
| | |
|---|---|
| 8. In the third panel, click on **RTOA_PlaceOrders** to insert orders into memory optimized table with clustered columnstore index.<br><br>*Tip: The orders insert progress will also be displayed in the first pane.* |  |

| | |
|---|---|
| 9. While the OLTP workload is running, click on **RTOA_YOYSales** to run the year-on-year report from in-memory table with clustered columnstore index.<br><br>*Note: The RTOA_YOYSales should be clicked when the OLTP workload is running to simulate real-time operational analytics scenario.* |  |

10. Once the execution is completed the results are displayed for Real-Time Operational Analytics in the **In-Memory OLTP and RTOA** panel under **Results** pane.

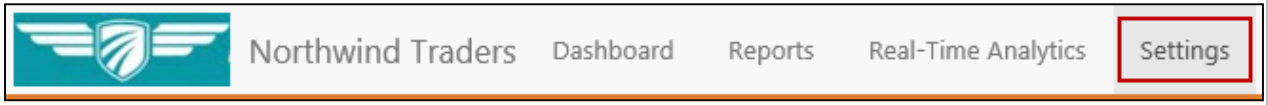
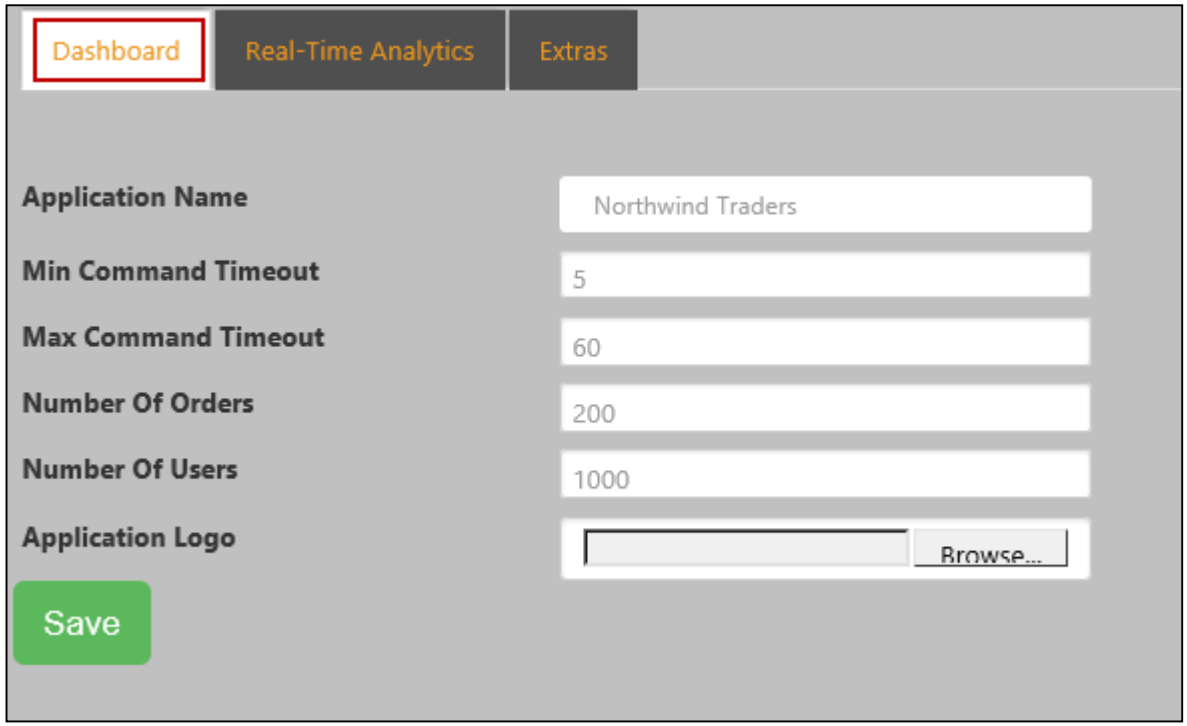
    *Note: The time taken for running real-time analytics is slightly higher than CCI on disk-based. There is minimal/no impact on the OLTP workload.*

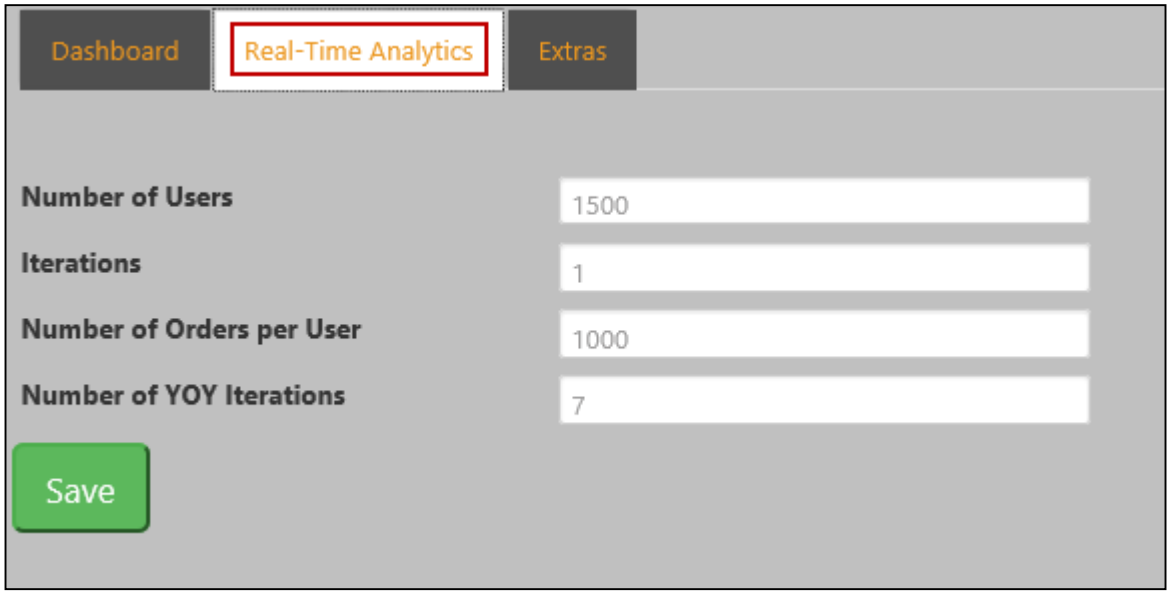
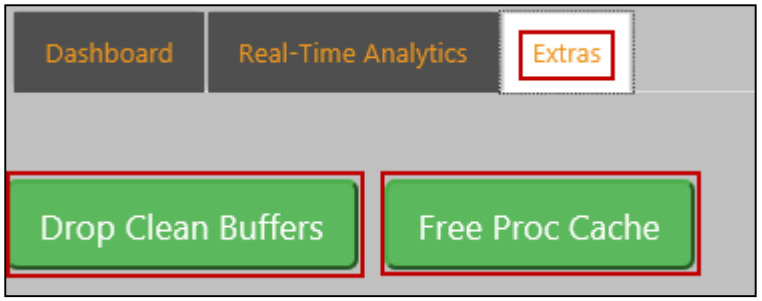


| | |
|---|---|
| 11. Once the report execution is completed, in the second pane under **CCI on In-Memory**, hover on the year 2016 to show the real-time sale amount for the year. |  |

# Appendix

## Settings

| Steps | Screenshot |
|---|---|
| 1. In the web browser, in the top panel click on **Setting** tab to go to settings page. |  |
| 2. In the Settings page, click on **Dashboard** tab to set the In-Memory OLTP demo settings. Click **Save** to save the settings.<br>1. Application Name is the company name.<br>2. Min Command Timeout – recommended 5 sec.<br>3. Max Command Timeout – recommended 60 sec.<br>4. Number of Orders – recommended 200.<br>5. Number of Users – recommended 1000.<br>6. Application Logo – upload the company logo.<br>*Note: Any changes other than the recommended values should be thoroughly tested.* |  |

| | |
|---|---|
| 3. In the Setting page, click on **Real-Time Analytics** tab to change the settings for real-time operational analytics demo. Click **Save** to save the settings.<br>1. Number of Users – recommended 1500.<br>2. Iterations – iteration of OLTP workload – recommended 1.<br>3. Number of Orders per User – recommended 1000<br>4. Number of YOY Iterations – recommended 7.<br>*Note: Any changes other than the recommended values should be thoroughly tested.* |  |
| 4. In the Settings page, click on Extra tab to execute **DBCC DROPCLEANBUFFERS** or **DBCC FREEPROCCACHE** commands on database server. |  |