Microsoft

SQL Server 2016 Airline Scenario Demo showcasing Stretch, Temporal and Always Encrypted - Demo Installation and Demo Script

Contents

Introduction

This demo is to showcase three new features of SQL Server 2016 in an Airline Scenario. The person who is showing the demo will be a Travel Agent booking flights for business travelers. We are showcasing Stretch Database together with Temporal and Always Encrypted.

The demo is to be set up using installation scripts and some manual process. It is up to the presenter to remove any of the resources created in Azure so that they do not incur costs while not in use. It is the presenter's responsibility to start/stop stretching the database if required. Note that this will still incur charges, you need to delete the remote DB in Azure.

It is important to point out that the demo script follows a strict execution path as the frontend site is only partially functional to cater for the story flow. Some of the elements are static and purely cosmetic, any interaction with these will most likely cause undesired outcomes.

To help with this, each step in the demo script section has been clearly labeled with the elements required to tell the story.

It is assumed that the presenter has some prior knowledge with SQL Server 2016 and will be comfortable navigating around SSMS. Some basic PowerShell knowledge may be required.

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

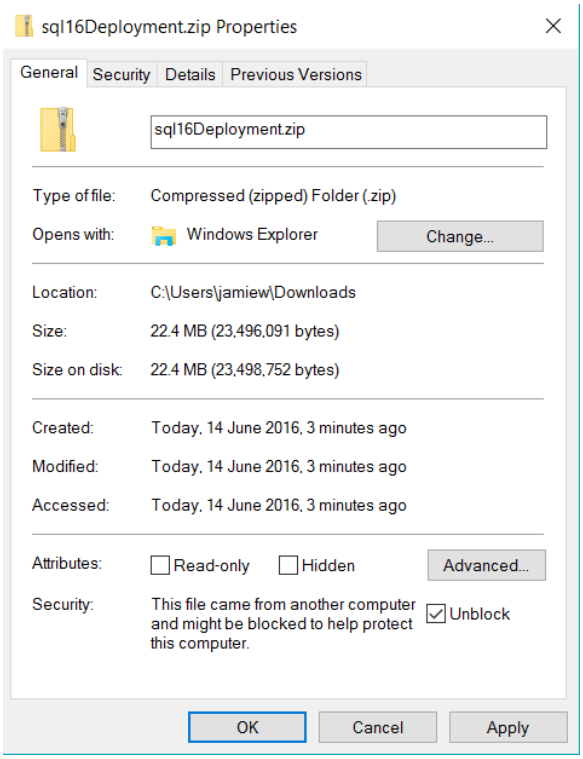
Installation Script

IMPORTANT

This deployment script is built on Azure PowerShell 1.4.0. If you are experiencing any issues, please install this version onto your machine.

You need to run this on Windows 10 with Edge installed. The web front-end is ran locally on your machine from a lightweight portable webserver.

1. Locate sql16Deployment.zip
    a. https://sql2016demoforeastasia.blob.core.windows.net/sql2016airlinedemov1/sql16Deployment.zip
    b. https://sql2016demoforeurope.blob.core.windows.net/sql2016airlinedemov1/sql16Deployment.zip
    c. https://sql2016demoforwestusa.blob.core.windows.net/sql2016airlinedemov1/sql16Deployment.zip
2. You first need to make sure the ZIP is unblocked. Right click properties then select unblock

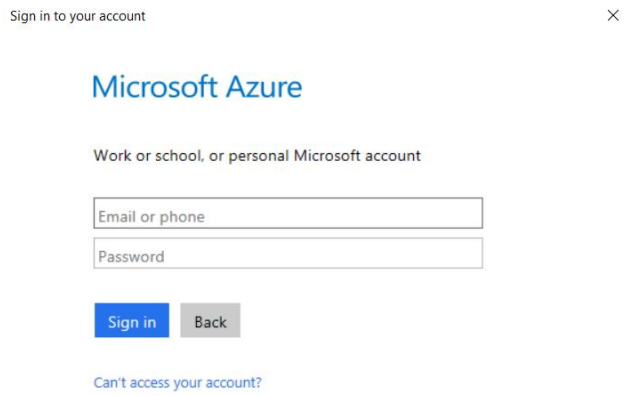


3. Make sure you have installed Azure PowerShell

4. Open an **ADMIN** PowerShell and navigate to the Scripts folder in your sql16Deployment. Run the following command .\deploy.ps1

```
PS C:\Code\MSCorp.SqlStretchDemo\latest deployment\sql16Deployment\sql16Deployment\Scripts> .\deploy.ps1
Starting Deployment for SQL Server 2016 Demo Environment
```

5. You will be prompted to sign into your Azure account.



6. Select the subscription you would like to deply to.

```
Environment           : AzureCloud
Account               : jamie.wilbraham@intergen.co.nz
TenantId              : e94f3bc1-b4f0-44c7-8f45-3ab8a99fa933
SubscriptionId        : 8812bbad-e8fc-4187-8a3d-a1a7b7f8b851
CurrentStorageAccount :

Loading subscriptions

Option SubscriptionId                        SubscriptionName
------ --------------                        ----------------
     1 7af011da-932a-4fff-83fe-ee     4ee7b Int
     2 8812bbad-e8fc-4187-8a3d-a1     8b851 Vis
     3 d76a9f0f-19a3-4b2f-971a-c1     4d135 Pay
     4 15c5cb6e-191a-40ea-9f69-08     7fe97 Azu



Select an option from the above list: 2
```

7. Give your deployment name a unique name that has to be 14 characters or less. This needs to be globally unique. Its good to put a random number at the end of the name to ensure it is unique.

```
Using subscription '8812bbad-e8fc-4187-8a3d-a1a7b7f8b851'
What is the name of deployment? Please make sure it's globally unique: stretcher10
```

8. Choose your deployment location and the deployment will start.

```
Choose Region
Where do you want the deployment?
[U] West US  [E] West Europe  [A] Southeast Asia  [?] Help (default is "U"): u
Deploying to West US, name: stretcher10
Creating resource group 'stretcher10' in location 'West US'
```

9. This process can take anywhere from 20-40 minutes to complete depending on your location. The process out the installation can be tracked from the output. An example is below.

```
VERBOSE: 5:24:20 PM - Create template deployment 'Storage-0614-0524'.
VERBOSE: 5:24:32 PM - Resource Microsoft.Storage/storageAccounts 'storagelektwtwljuiz4' provisioning status is running
VERBOSE: 5:25:05 PM - Resource Microsoft.Storage/storageAccounts 'storagelektwtwljuiz4' provisioning status is
succeeded
Microsoft.Azure.Commands.Resources.Models.PSResourceGroupDeployment
Uploading artifacts to blob
Artifacts uploaded
Starting SQL Server 16 deployment...
VERBOSE: 5:25:46 PM - Template is valid.
VERBOSE: 5:25:51 PM - Create template deployment 'template'.
VERBOSE: 5:26:02 PM - Resource Microsoft.Storage/storageAccounts 'storagelektwtwljuiz4' provisioning status is
succeeded
VERBOSE: 5:26:14 PM - Resource Microsoft.Network/networkSecurityGroups 'stretcher10' provisioning status is running
VERBOSE: 5:26:14 PM - Resource Microsoft.Network/virtualNetworks 'stretcher10' provisioning status is running
VERBOSE: 5:26:14 PM - Resource Microsoft.Network/publicIPAddresses 'stretcher10' provisioning status is running
VERBOSE: 5:26:25 PM - Resource Microsoft.Network/networkSecurityGroups 'stretcher10' provisioning status is succeeded
VERBOSE: 5:26:25 PM - Resource Microsoft.Network/virtualNetworks 'stretcher10' provisioning status is succeeded
VERBOSE: 5:26:25 PM - Resource Microsoft.Network/publicIPAddresses 'stretcher10' provisioning status is succeeded
VERBOSE: 5:26:37 PM - Resource Microsoft.Network/networkInterfaces 'stretcher10' provisioning status is succeeded
VERBOSE: 5:26:37 PM - Resource Microsoft.Sql/servers/firewallrules 'stretcher10/AllowEverything' provisioning status is
 succeeded
VERBOSE: 5:26:37 PM - Resource Microsoft.Sql/servers/firewallrules 'stretcher10/AllowAllWindowsAzureIps' provisioning
status is succeeded
VERBOSE: 5:26:37 PM - Resource Microsoft.Sql/servers 'stretcher10' provisioning status is succeeded
VERBOSE: 5:26:48 PM - Resource Microsoft.Compute/virtualMachines 'stretcher10' provisioning status is running
VERBOSE: 5:33:27 PM - Resource Microsoft.Compute/virtualMachines/extensions 'stretcher10/CustomScriptExtension' provisioning
status is running
VERBOSE: 5:33:27 PM - Resource Microsoft.Compute/virtualMachines 'stretcher10' provisioning status is succeeded
VERBOSE: 5:46:32 PM - Resource Microsoft.Compute/virtualMachines/extensions 'stretcher10/CustomScriptExtension' provisioning
status is succeeded
```

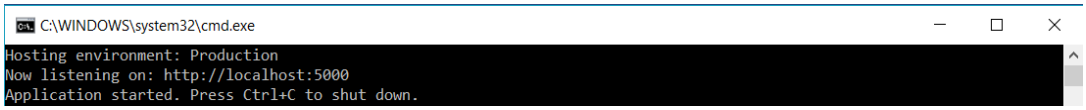
10. Once completed you will see an output of the deployment information. Take note of the Virtual Machine IP address if you want to RDP into it. It is important to note that a private cert is installed on your machine to ensure AE functionality will work.

```
ParametersString        :
                          Name             Type                       Value
                          ===============  =========================  ==========
                          env              String                     stretcher10
                          adminName        String                     DemoAdmin
                          adminPassword    SecureString
                          _artifactsLocation  String
                          https://storagelektwtwljuiz4.blob.core.windows.net/artifacts
                          _artifactsLocationSasToken  SecureString
                          locationFriendly  String                     West US

Outputs                 :
OutputsString           :

Deployment complete
VM deployed to 40.118.254.197
Unzipped WebApp
Installing AE Certificate
```

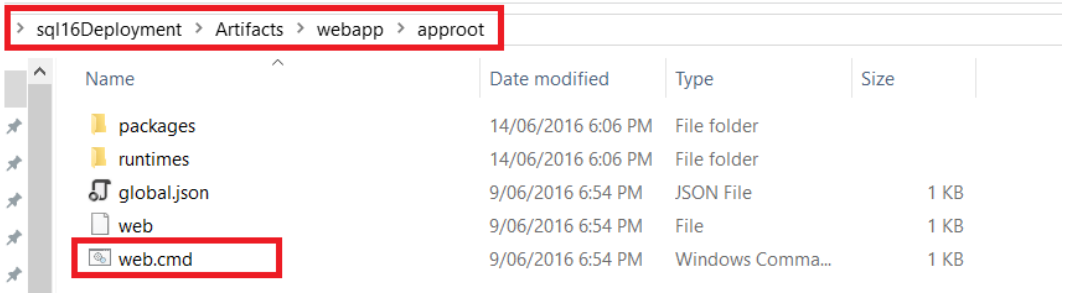
11. You can go to Azure to confirm the components have been installed correctly. The VM database will start stretching straight away.

NAME	TYPE	RESOURCE GROU...	LOCATION	SUBSCRIPT
stretcher10	Virtual ma...	stretcher10	West US	Visual Stu
stretcher10	Network i...	stretcher10	West US	Visual Stu
stretcher10	Network s...	stretcher10	West US	Visual Stu
stretcher10	Public IP a...	stretcher10	West US	Visual Stu
stretcher10	Virtual net...	stretcher10	West US	Visual Stu
stretcher10	SQL server	stretcher10	West US	Visual Stu
RDAstretchdemodb4022F0E2-	SQL datab...	stretcher10	West US	Visual Stu
storagelektwtwljuiz4	Storage ac...	stretcher10	West US	Visual Stu

12. You will notice a cmd window will pop up on completion starting a webserver. You need to browse to http://localhost:5000 to get to the login page of your application.



13. To run this manually in the future, run web.cmd from the approot (see below)



14. This will need to be running to connect to your frontend site. Always browse to localhost:5000 from your local machine and not the VM.

15. To Remote Desktop into the solution enter the following information:

Computer : *The IP address obtained in step 9*

User: DemoAdmin

Password: Pass@word2!

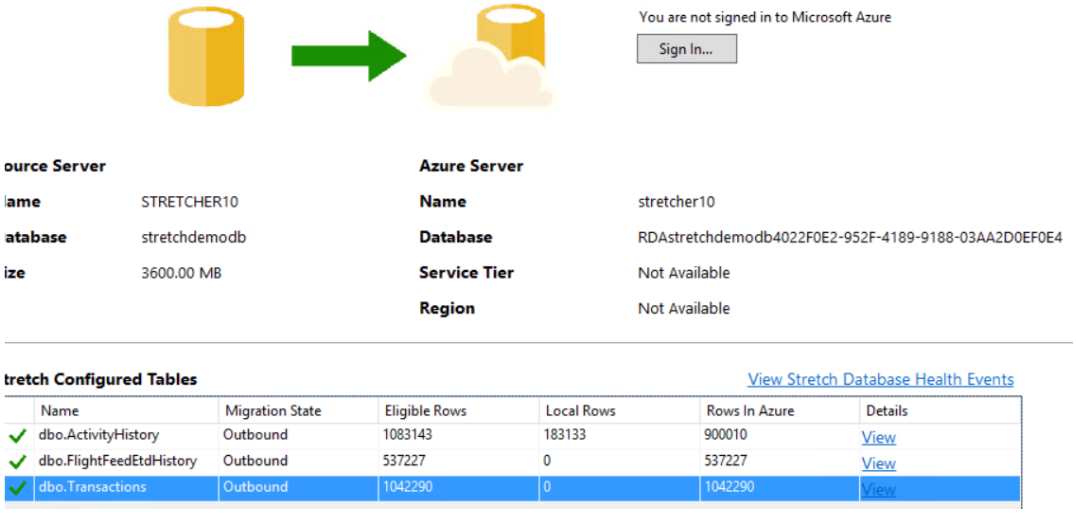
From here you will be able to access SSMS and view the database.

In the object explorer,

right-click stretchdemodb > tasks > stretch > monitor

to ensure the tables are being stretched. Note that this may take a while to complete.

It should look similar to below.



Installation Complete

Demo Script

Scenario

The presenter is a sales agent for Wingtips Travel Co, a travel company that specializes in providing a concierge service to their clients.

In this demo, the presenter will simulate receiving a travel request from a fictional client named Jamie Campbell, for the sake of the demo it is an email request. Jamie needs to book a one-way flight from Arizona to Quebec and this flight requires a connection in Chicago.

Due to a range of factors in Chicago that can cause delays, the sales agent needs to find a suitable flight that is well priced and reliable. The sales agent (presenter) will showcase some of the new features in SQL Server 2016 that will ultimately guide them into booking a flight based on existing data.

1. Login

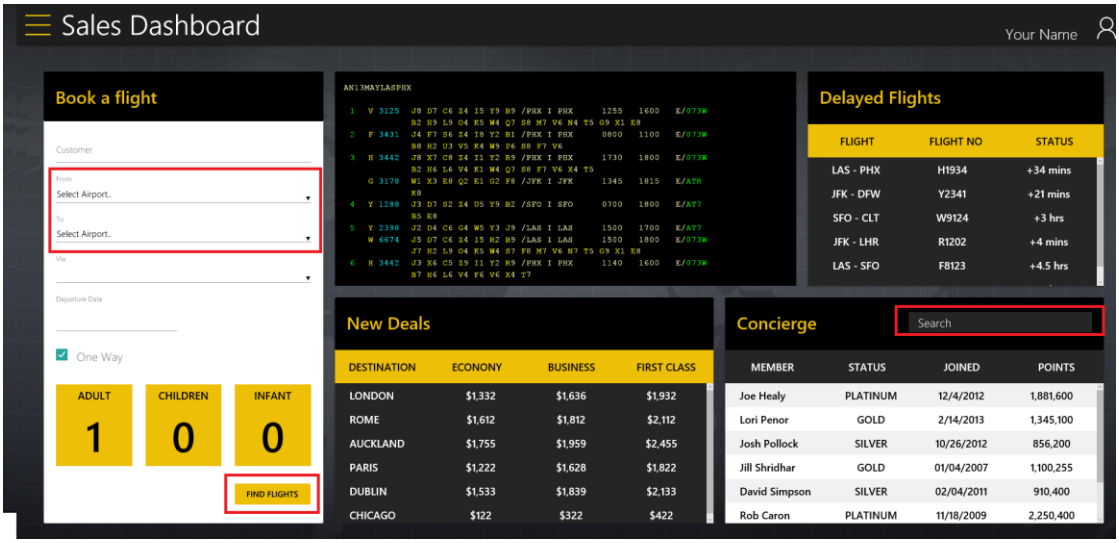
After you have arrived at your homepage in the MS Edge browser (see installation steps), you will be presented at the following login screen. The password is prepopulated, however, you have the option to enter a username. This will flow through the app, if its left blank, it will default to “Sales Agent”.

Select the login button to continue to the Sales Dashboard.

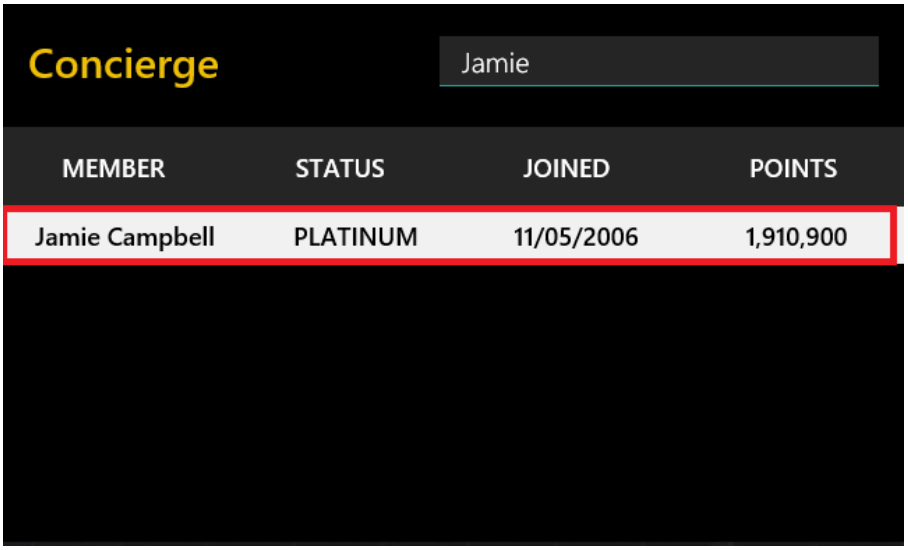
2. **Sales Dashboard**

The sales dashboard’s primary purpose is to aid the story. It doesn’t really show any features of SQL Server 2016 and should the time spent here should be brief. The presenter will be inputting some values from the

clients request so that they can assess potential flights. The image below shows the elements you will need to interact with.



First, start typing the name “Jamie” in the concierge search and the list will filter with Jamie Campbell displaying. Select this user.



You will now need to enter additional information into the “Book a flight” section. You will notice from the last step “Jamie Campbell” has already been prepopulated in the customer input element. Enter the following information to replicate the image below.

FROM – Arizona - AZA

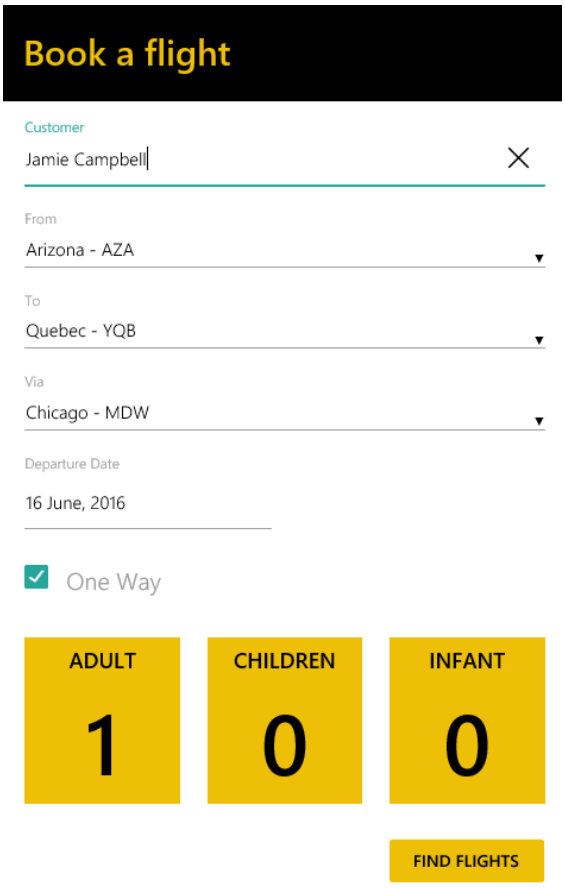
TO – Quebec – YQB

VIA – *this will be populated with Chicago – MDW after completing the "To" input. It's the only option indicating the only flight is through Chicago.*

DEPARTURE DATE – Set a date in the future relative to when the demo is presented

ONE WAY – Leave selected

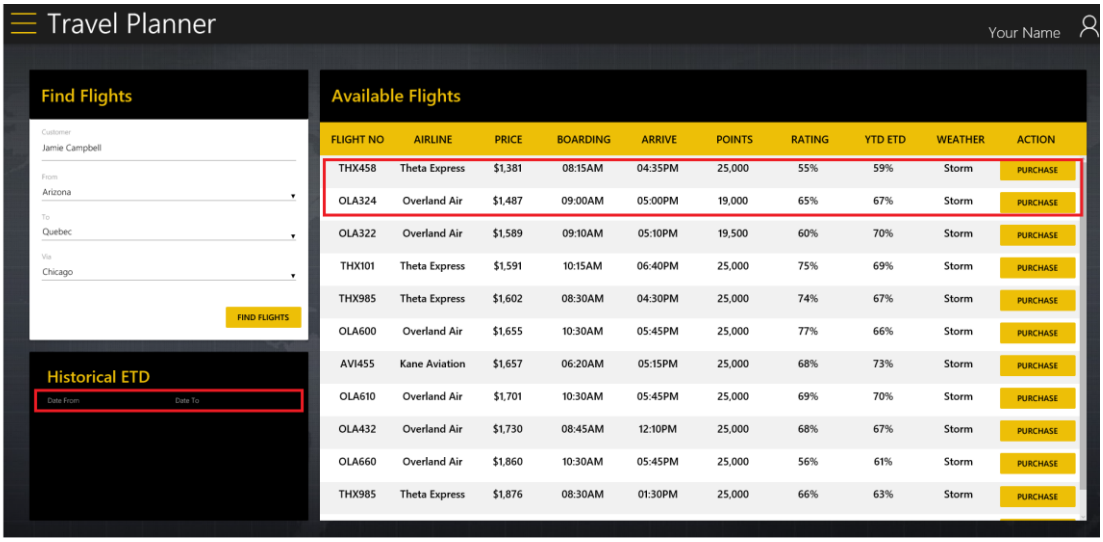
Your input should look like the following:



Select "Find Flights" to continue to the travel planner.

3.Travel Planner

The part of the demo highlights 2 features of SQL Server 2016, temporal and also stretching the history of the system-versioned temporal table. Below indicates the page elements that you will need to interact with in this stage of the demo. It will also require switching to your database in SSMS to highlight how the components are functioning in the DB.

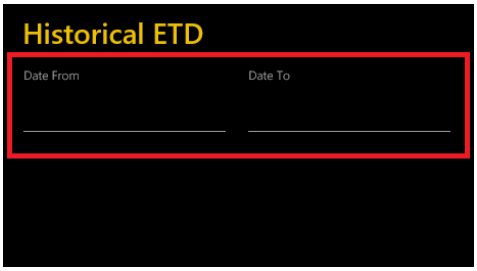


The two top flights returned are the ones of interest. The YTD ETD (Year-To-Date Estimated-Time (of)-Departure) and PRICE are the columns of interest. We can see that flight THX458 have a cheaper price, but a lower chance of leaving on time.

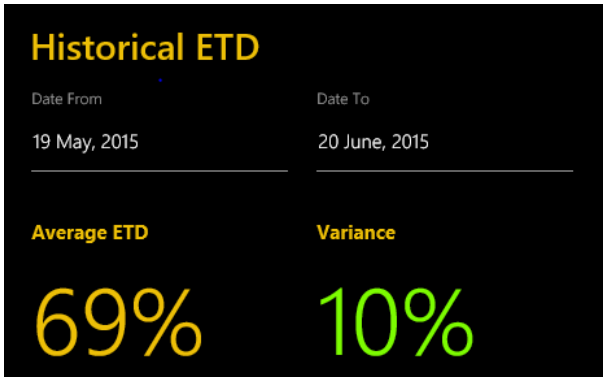
Select flight THX458 and the line will be highlighted grey.

With the power of temporal you are able to query the history of data as it was at a previous time. The values here are produced client-side and not actually hooked up to the DB as it is too complex to know when the presenter will be carrying out the demo. Hypothetically, these values would be read from the dbo.FlightEtdFeed in the DB.

Choose From/To dates below ensuring they are in the past (over a year ago). The dates are irrelevant and the flight selected indicates the value.



You will notice the result always yields a positive variance against the current YTD of 59%. You can repeat with different dates to get the point across. It will never generate a negative variance for the sake of the demo.

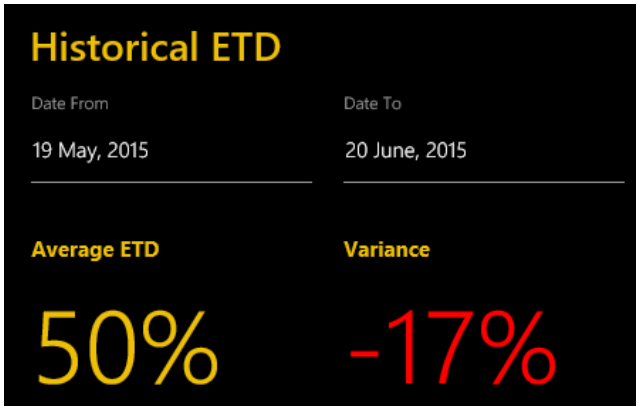


So we now know the flights history is usually shows a higher ETD currently shown, which is good information for the sales agent to help assist their decision

We are still interested in flight “OLA324” as an option, so we repeat the process. Leave the dates the same for now.

Select flight OLA324

You will immediately notice the variance drops, indicating “OLA324” flights ETD history has not been as reliable as it is currently. You can repeat this with dates to express the point.



It now makes sense to book flight “THX458” as it is cheaper and is has had a relatively promising history compared to “OLA324”.

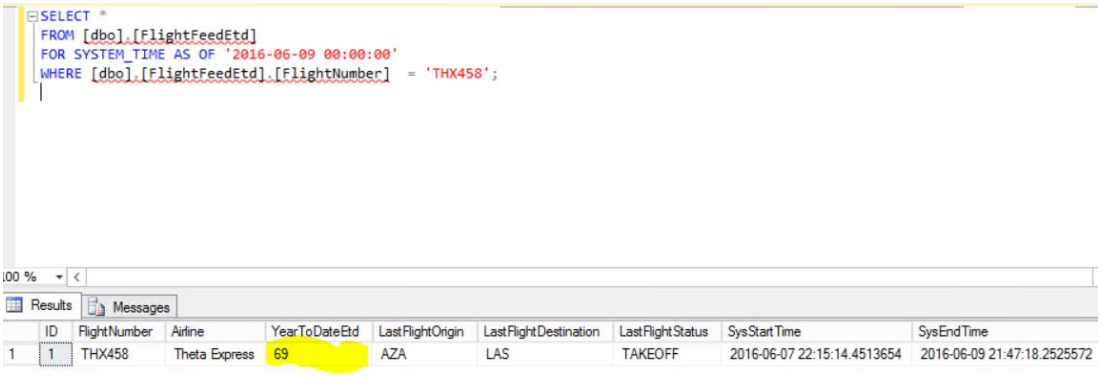
Before purchasing the flight, you need to switch to the DB and run some queries.

Explain you are going to run a query to demonstrate what the value of flight THX458 was at a given time. For the sake of the demo, we can look at what it was on the 9th of June 2016 (it will be guaranteed to be different).

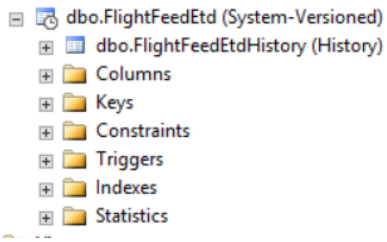
Run the below query on the DB

```
USE stretchdemodb

SELECT *
FROM [dbo].[FlightFeedEtd]
FOR SYSTEM_TIME AS OF '2016-06-09 00:00:00'
WHERE [dbo].[FlightFeedEtd].[FlightNumber]  = 'THX458';
```

You will notice the YTD ETD is higher that is what is being shown in the frontend which abides to the logic of the story.



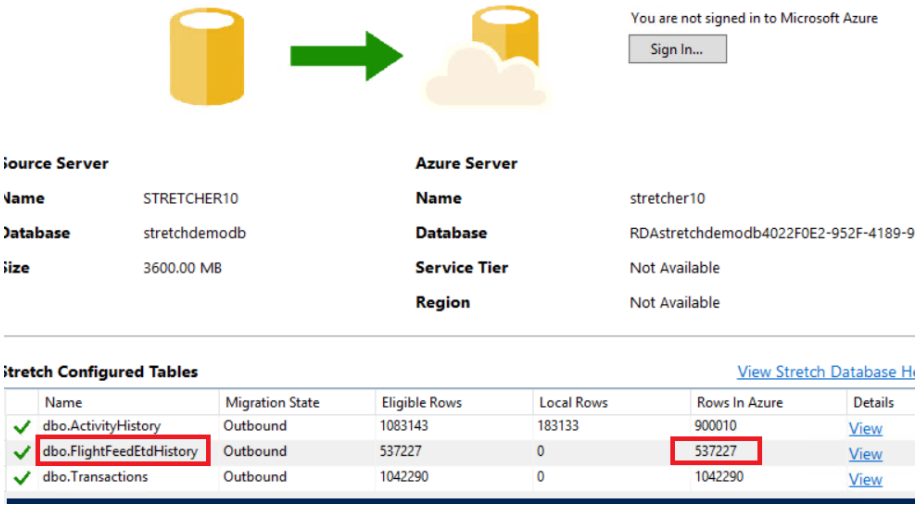
Expand the dbo.FlightFeedEtd table to show there is a history table. Explain that this is being stretched full to Azure



Check the monitor page to show the values in the history are being fully stretched in Azure. Point out the transaction count – you will need to bring this up after the transaction has been completed later in the demo.

In the object explorer,

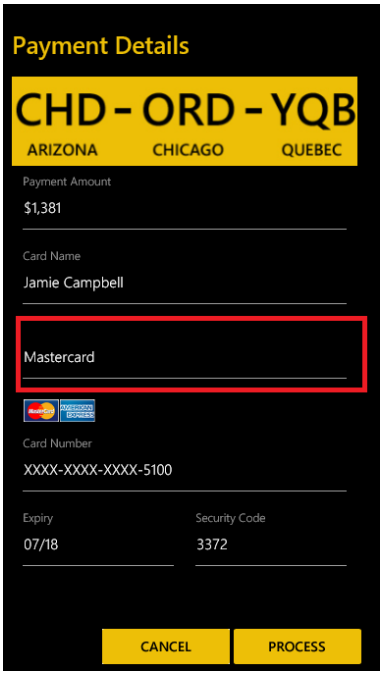
right-click stretchdemodb > tasks > stretch > monitor



Switch back to the front end website

After deciding on the flight in the previous steps, click on the yellow PURCHASE button to proceed to the payment details modal.

You only need to select the card from the dropdown, the rest of the information will be populated. See image below for expected input.

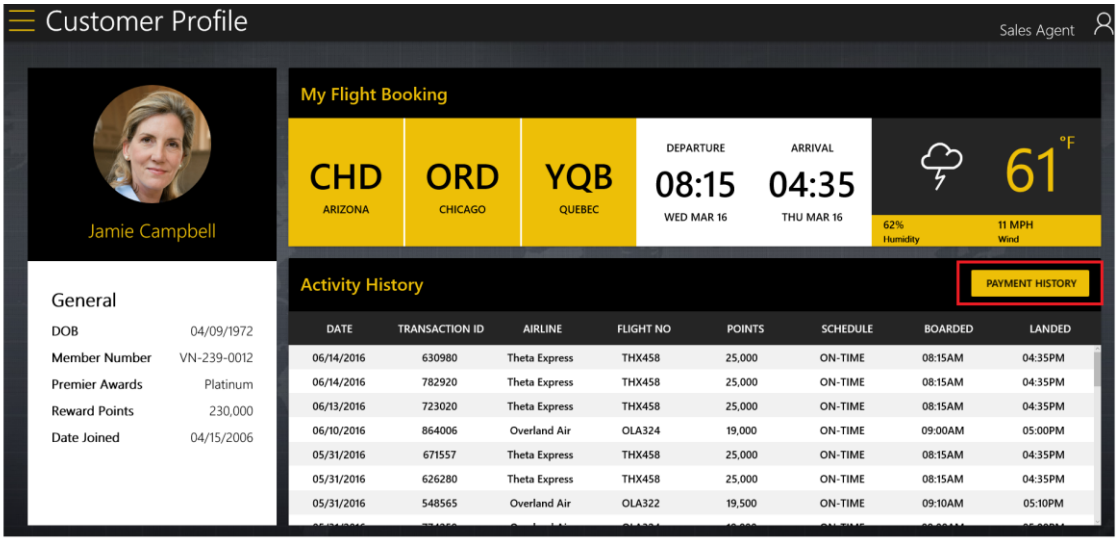


Click process to complete the transaction and be redirected to the customer profile.

4. Customer Profile

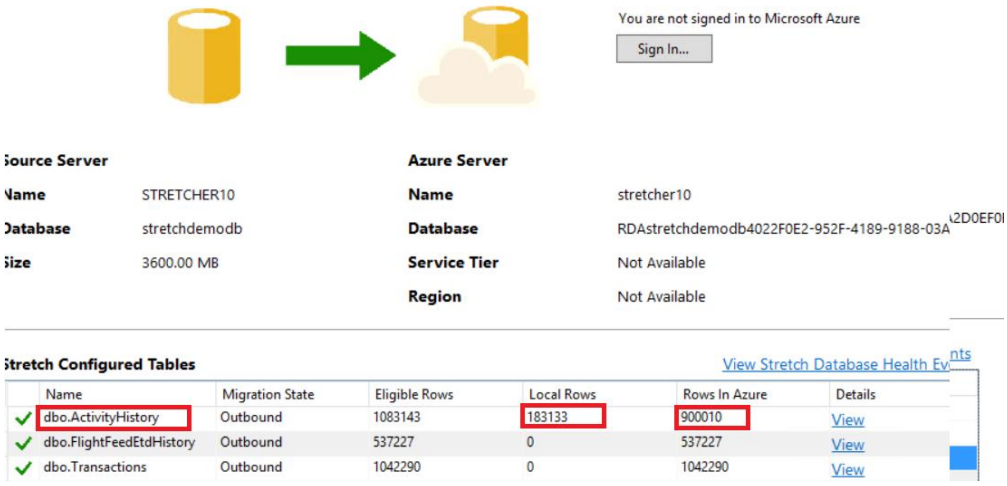
The purpose of this page is to highlight partially stretching a DB based on a query and also stretching Always Encrypted columns.

You will be required to switch to the DB for this part. Below are the elements you will be interacting with.



The activity history shows Jamie Campbell's flight bookings. It is important to note that any transaction from before 1-1-2014 is being stretched to Azure based on a filter. The top flight is the one you just purchased.

You will be able to see the rows that are stretched and which are local in the stretch monitor.



Select PAYMENT HISTORY

The history shows credit card information and purchase details that is actually encrypted in the DB. As well as being completely stretched seamlessly.

It is important to touch on that this is seamlessly handled in the API that calls the endpoint by just installing a client certificate on the API server.

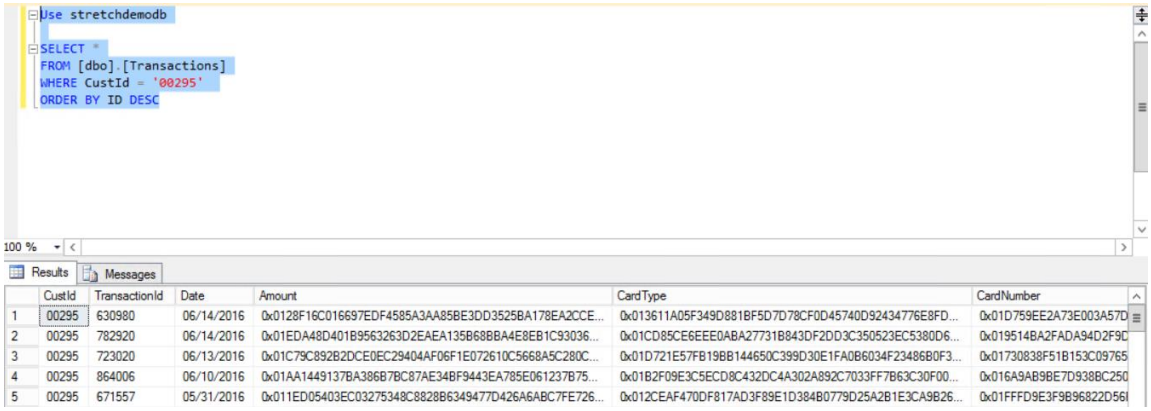
Payment Details

DATE	TRANS ID	AMOUNT	CARD TYPE	CARD NUMBER	POINTS
06/14/2016	630980	$1,381	Mastercard	xxxxxxxxxxxx5100	25,000
06/14/2016	782920	$1,381	Mastercard	xxxxxxxxxxxx5100	25,000
06/13/2016	723020	$1,381	Mastercard	xxxxxxxxxxxx5100	25,000
06/10/2016	864006	$1,487	Mastercard	xxxxxxxxxxxx5100	19,000
05/31/2016	671557	$1,381	AMEX	xxxxxxxxxxx21001	25,000
05/31/2016	626280	$1,381	AMEX	xxxxxxxxxxx21001	25,000
05/31/2016	548565	$1,589	Mastercard	xxxxxxxxxxxx5100	19,500

Switch to the database and run the following query

```
Use stretchdemodb

SELECT *
FROM [dbo].[Transactions]
WHERE CustId = '00295'
ORDER BY ID DESC
```

This will return all the transaction in with the latest first, you can clearly see below the returned data is encrypted.



	CustId	TransactionId	Date	Amount	CardType	CardNumber
1	00295	630980	06/14/2016	0x0128F16C016697EDF4585A3AA85BE3DD3525BA178EA2CCE...	0x013611A05F349D881BF5D7D78CF0D45740D92434776E8FD...	0x01D759EE2A73E003A57D
2	00295	782920	06/14/2016	0x01EDA48D401B9563263D2EAEA135B68BBA4E8EB1C93036...	0x01CD85CE6EEE0ABA27731B843DF2DD3C350523EC5380D6...	0x019514BA2FADA94D2F9C
3	00295	723020	06/13/2016	0x01C79C892B2DCE0EC29404AF06F1E072610C5668A5C280C...	0x01D721E57FB19BB144650C399D30E1FA0B6034F23486B0F3...	0x01730838F51B153C09765
4	00295	864006	06/10/2016	0x01AA1449137BA386B7BC87AE34BF9443EA785E061237B75...	0x01B2F09E3C5ECD8C432DC4A302A892C7033FF7B63C30F00...	0x016A9AB9BE7D938BC250
5	00295	671557	05/31/2016	0x011ED05403EC03275348C8828B6349477D426A6ABC7FE726...	0x012CEAF470DF817AD3F89E1D384B0779D25A2B1E3CA9B26...	0x01FFFD9E3F9B96822D56

You can switch to the monitor to also show that the dbo.transactions table is fully stretched to Azure and contains a million+ rows.

You can also point out the remote count has increased by 1 transaction *(1042290 -> 1042291).*

You are not signed in to Microsoft Azure

Sign In...

Source Server		Azure Server	
Name	STRETCHER10	Name	stretcher10
Database	stretchdemodb	Database	RDAstretchdemodb4022F0E2-952F-4189-9188-03AA2D0EF0E4
Size	3600.00 MB	Service Tier	Not Available
		Region	Not Available

Stretch Configured Tables View Stretch Database Health Events

Name	Migration State	Eligible Rows	Local Rows	Rows In Azure	Details
dbo.ActivityHistory	Outbound	1083144	183134	900010	View
dbo.FlightFeedEtdHistory	Outbound	537227	0	537227	View
dbo.Transactions	Outbound	1042291	0	1042291	View

End of demo.

Terms of Use

By using this Hands-on Lab, you agree to the following terms:

The technology/functionality described in this Hands-on Lab is provided by Microsoft Corporation in a "sandbox" testing environment for purposes of obtaining your feedback and to provide you with a learning experience. You may only use the Hands-on Lab to evaluate such technology features and functionality and provide feedback to Microsoft. You may not use it for any other purpose. You may not modify, copy, distribute, transmit, display, perform, reproduce, publish, license, create derivative works from, transfer, or sell this Hands-on Lab or any portion thereof.

COPYING OR REPRODUCTION OF THE HANDS-ON LAB (OR ANY PORTION OF IT) TO ANY OTHER SERVER OR LOCATION FOR FURTHER REPRODUCTION OR REDISTRIBUTION IS EXPRESSLY PROHIBITED.

THIS HANDS-ONLAB PROVIDES CERTAIN SOFTWARE TECHNOLOGY/PRODUCT FEATURES AND FUNCTIONALITY, INCLUDING POTENTIAL NEW FEATURES AND CONCEPTS, IN A SIMULATED ENVIRONMENT WITHOUT COMPLEX SET-UP OR INSTALLATION FOR THE PURPOSE DESCRIBED ABOVE. THE TECHNOLOGY/CONCEPTS REPRESENTED IN THIS HANDS-ON LAB MAY NOT REPRESENT FULL FEATURE FUNCTIONALITY AND MAY NOT WORK THE WAY A FINAL VERSION MAY WORK. WE ALSO MAY NOT RELEASE A FINAL VERSION OF SUCH FEATURES OR CONCEPTS. YOUR EXPERIENCE WITH USING SUCH FEATURES AND FUNCITONALITY IN A PHYSICAL ENVIRONMENT MAY ALSO BE DIFFERENT.

FEEDBACK. If you give feedback about the technology features, functionality and/or concepts described in this Hands-on Lab to Microsoft, you give to Microsoft, without charge, the right to use, share and commercialize your feedback in any way and for any purpose. You also give to third parties, without charge, any patent rights needed for their products, technologies and services to use or interface with any specific parts of a Microsoft software or service that includes the feedback. You will not give feedback that is subject to a license that requires Microsoft to license its software or documentation to third parties because we include your feedback in them. These rights survive this agreement.

MICROSOFT CORPORATION HEREBY DISCLAIMS ALL WARRANTIES AND CONDITIONS WITH REGARD TO THE HANDS-ON LAB , INCLUDING ALL WARRANTIES AND CONDITIONS OF MERCHANTABILITY, WHETHER EXPRESS, IMPLIED OR STATUTORY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. MICROSOFT DOES NOT MAKE ANY ASSURANCES OR REPRESENTATIONS WITH REGARD TO THE ACCURACY OF THE RESULTS, OUTPUT THAT DERIVES FROM USE OF THE VIRTUAL LAB, OR SUITABILITY OF THE INFORMATION CONTAINED IN THE VIRTUAL LAB FOR ANY PURPOSE.