Before the interview, I created scripts to understand the 800+ database system. These work on my local desktop, however not sure how it will interact with Esub. It may require minor testing and edits.

1)

Currently Esub has 800+ multitenant databases,

In preparation to create a reporting database,

Please supply a Template Customer database.

Additionally, Please create a DatabaseDestination which has same exact schema/tables as Template database. It can be a empty shell db.

I created ReportingDB. One can use publish profile or schema compare, etc to generate the empty shell Reportingdb.

You can edit the variables before running.

declare @DatabaseTemplate nvarchar(max) = 'Customer2'

declare @DatabaseDestination nvarchar(max) = 'ReportingDb'

if you want to exclude databases, Edit this statement

and db\_Name(database\_id) <> @DatabaseTemplate and db\_Name(database\_id) <> 'reportingdb' and db\_Name(database\_id) <> 'electronics' and db\_Name(database\_id) <> 'mytestdb'

**View:**

It will grab All tables from All databases into one database, and create one view.

However, View performance will be slow. *This is due to selecting many binary trees in different dbs on different pages.*

Result:

Create View customerVw as

SELECT

[customerid],

[customername]

FROM Customer4.dbo.customer

UNION ALL

SELECT

[customerid],

[customername]

FROM Customer3.dbo.customer

UNION ALL

SELECT

[customerid],

[customername]

FROM Customer.dbo.customer

2) Since views are slow, we can utilize StoredProcedure which will materialize a table, I will need to add in code for CreateDate and UpdateDate in a where clause.

Result:

Create Procedure customerImport as

Insert into customer

(

[customerid],

[customername]

)

SELECT

[customerid],

[customername]

FROM Customer4.dbo.customer

UNION ALL

SELECT

[customerid],

[customername]

FROM Customer3.dbo.customer

UNION ALL

SELECT

[customerid],

[customername]

FROM Customer.dbo.customer

3) Finally, to conduct **schema comparisons** to a Template db in Powershell. (This maybe harder to setup.)

Right click on a proper database in SSMS and Task🡪 Generate Scripts

Prepare another text file naming all the databases in separate lines, which ones you want to compare.

This will print out which databases have a schema difference.

#OriginalDacpac – Right click on a database in SSMS and Task🡪 Generate Scripts

$OriginalScript = 'C:\Users\Ritwik\Desktop\DatabaseCompare\OriginalScript.sql'

#Source: Place in test file below, list databases by running select \* from sys.databases where database\_id > 4

$DatabaseSourceList = 'C:\Users\Ritwik\Desktop\DatabaseCompare\DatabaseList.txt'

#Destination: Output Database Files Generate Scripts

$Filepath='C:\Users\Ritwik\Desktop\DatabaseCompare\scripts' # local directory to save build-scripts to

**Output:**

Created Database Customer

CustomerFiles are the same

Created Database Customer4

Customer4 Files are different