The purpose of this document is to descript the steps to copy Auzre SQL server database between different types of Azure DB Instance ( in terms of Dev/UAT or Prod environment) by using “Export/Import Data-Tier Application” (to/from .bacpac backup file) task through SSMS or Azure Data Studio.

Normally we can just use SQL database copy in Azure Portal, but sometimes the Azure DB instance security setting or the other setting compitable issue prevent us to copy directly.

We can first copy the database to a different database name (prevent intefering with live database) **and add to the Hyperscale elastic pool. The instance should be close to your location.** The copy through Azure Portal will work ifit’s on same type of instance (Dev/UAT on Dev/UAT, Prod on Prod).

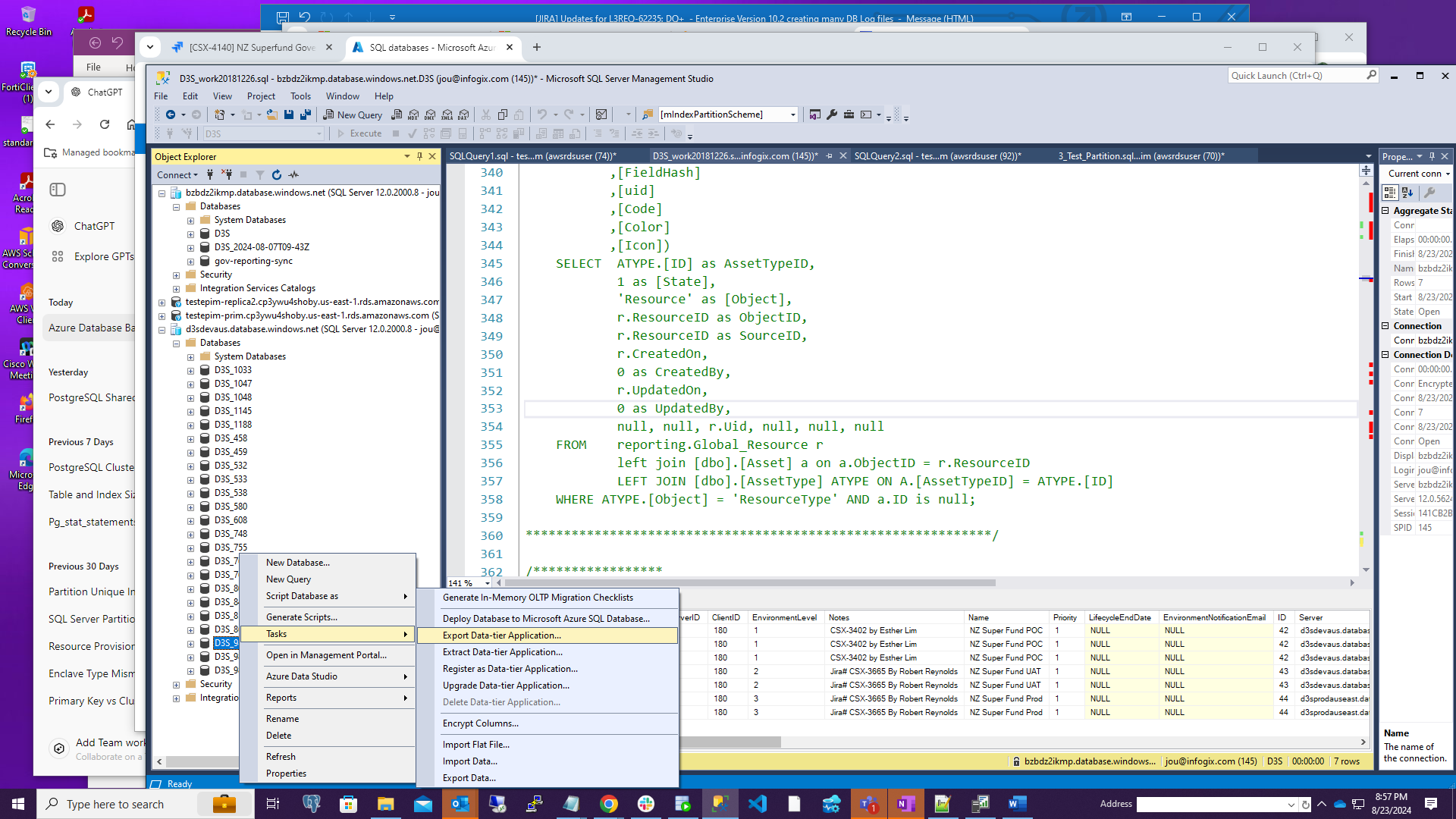
Most likely it will run into time-out error whenever the table is large (> millions records), we must make sure the copy performance is fast enough and the database in in Hyperscale elastic pool. The Time-out erro mainly depends on specific tables, not the total database size. Time-out issue only happens on export process.

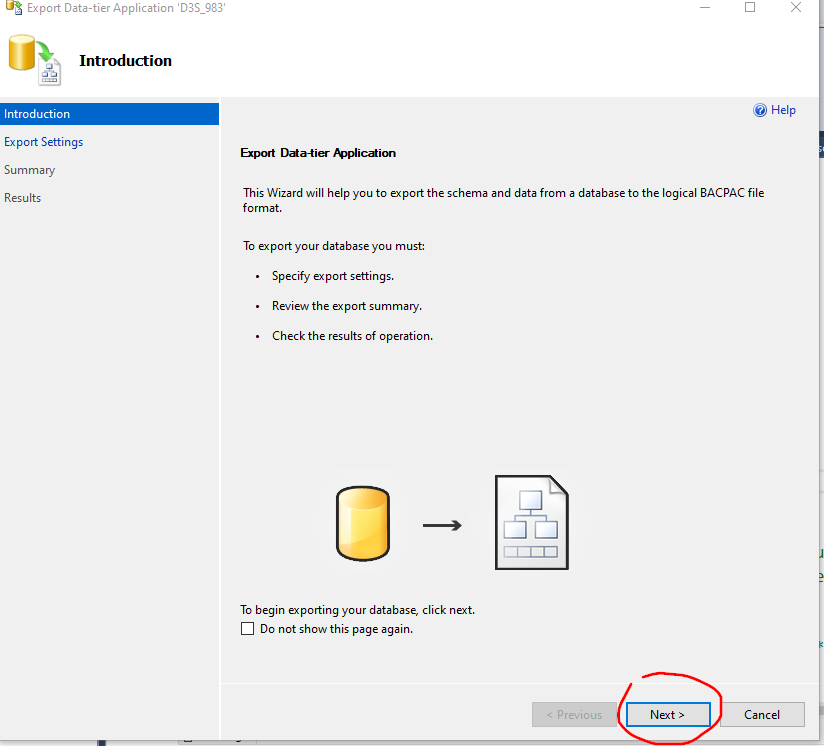
For 5 GB database, the whole process will take 2 hours. For 50 GB database, it will take more than 10 hours. The normal database copy through Azure Portal will take less than an hour.

The process will ONLY work by using SSMS or Azure Data Studio, I’ll discuss the reason at the end of this document.

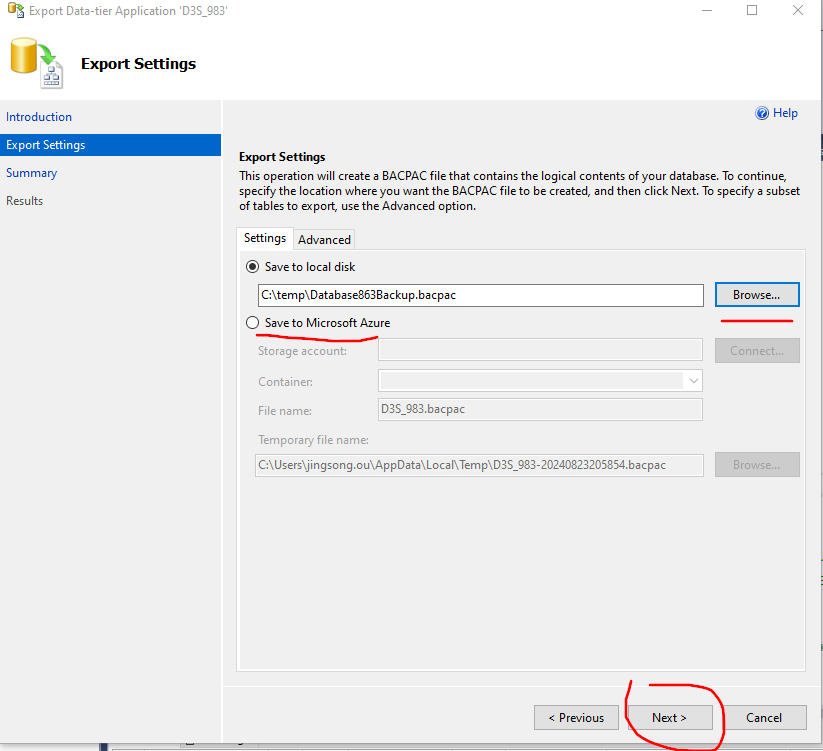
**Export Data:**

First right click the source database name and choose Tasks-> Export Data-tier Application

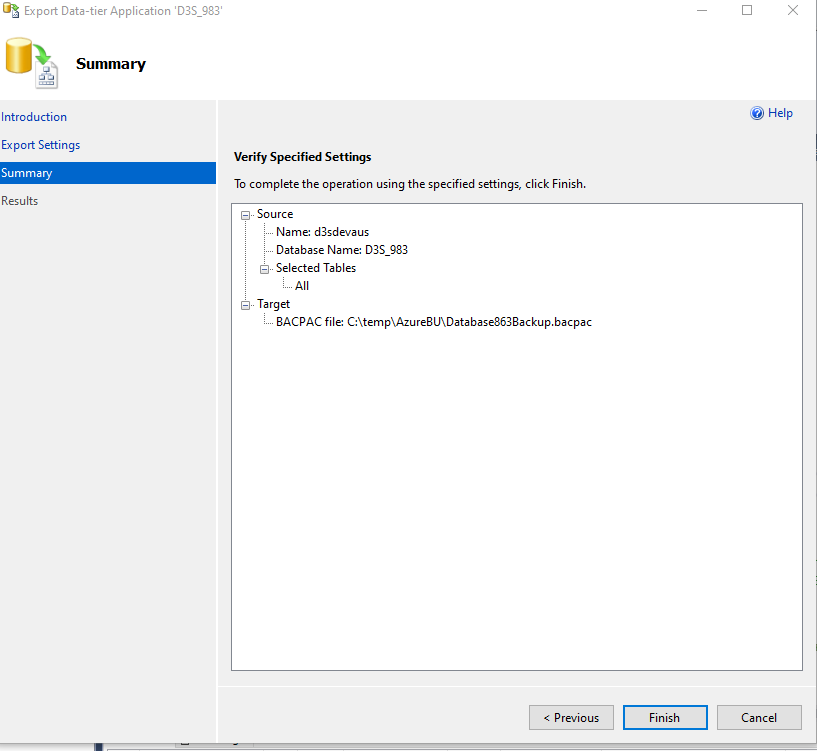




Choose the folder to store the backup file that will end with \*.bacpac



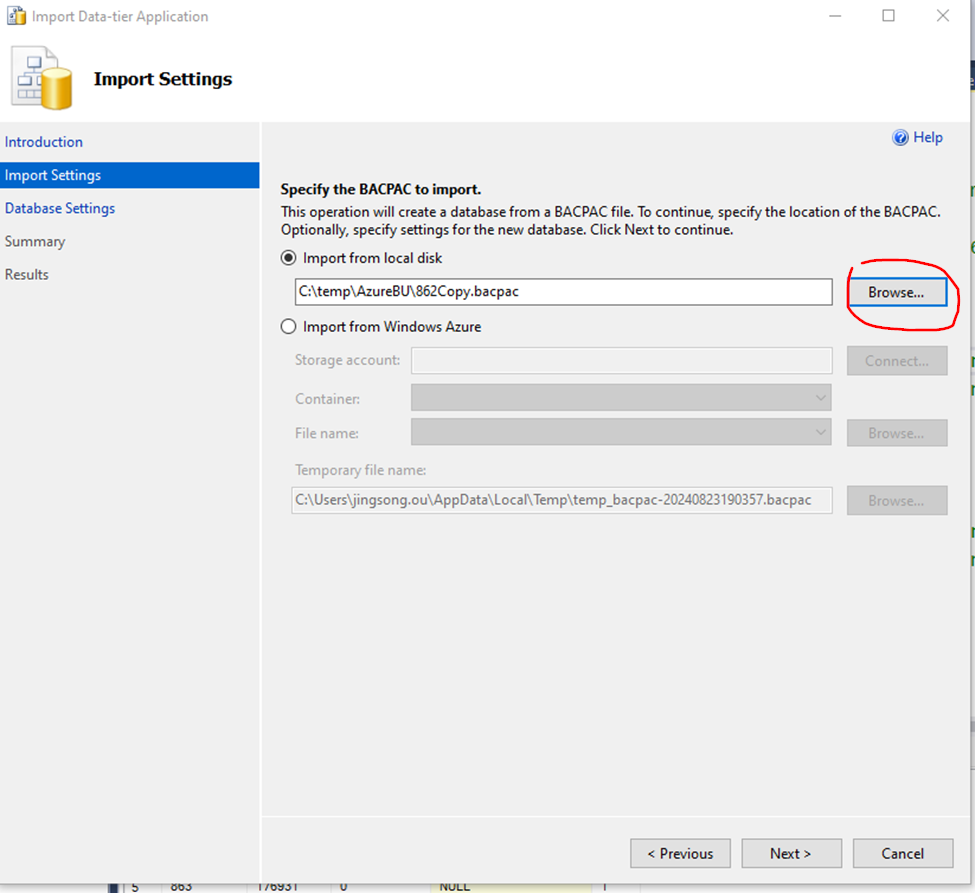
Option “Save to Microsoft Azure” will be too slow and result in timeout. So pickup “Save to local disk.”

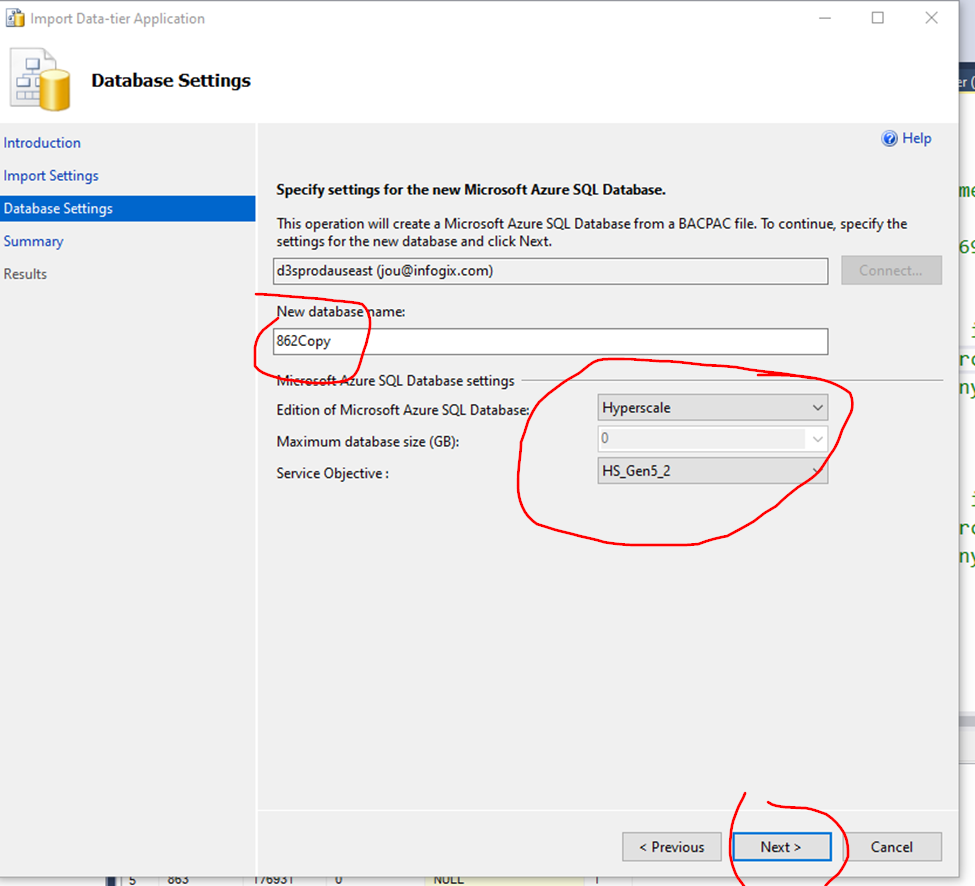


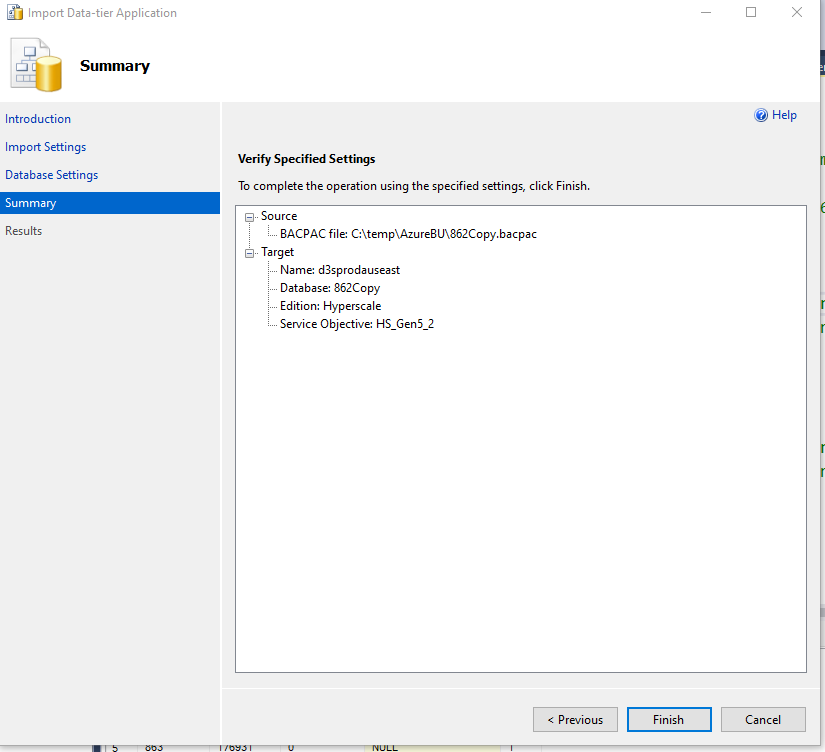
Click Finish to save the file.

**Import data:**

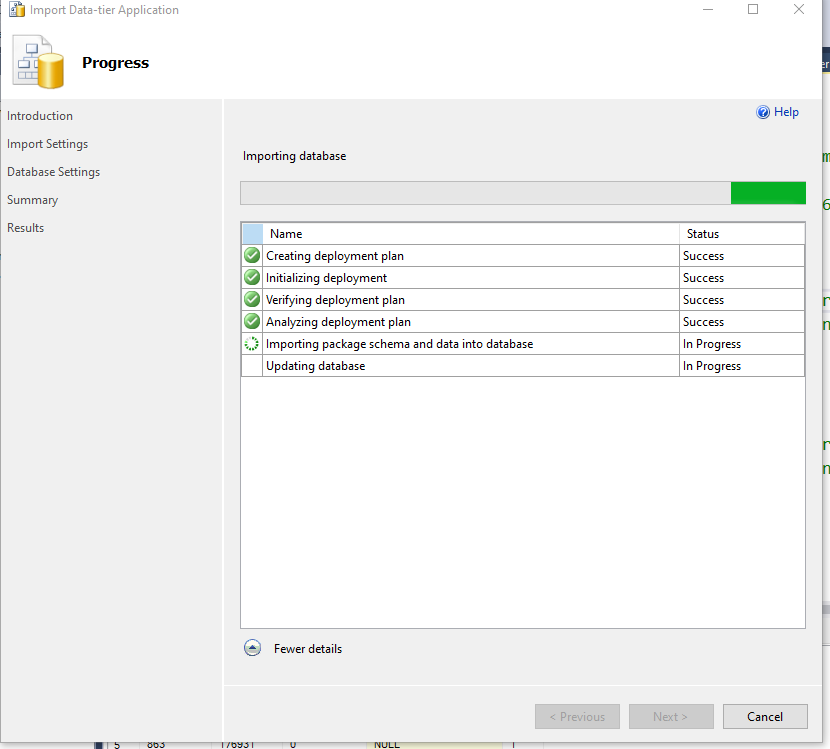
Right click “Databases” on SSMS, Choose “Import Data-Tier Application” and the following windows will popup.







Click Finish.



**It takes about one hour to import 4 GB bacpac file**.

The import process will be long, but it will not run into time-out error.

Go to Azure Portal, add newly created database to proper elastic pool. Rename the database and active the application by changing Company.Status in Community database.

Remember to rename or delete the intermediate database (through Azure Portal) that copied from source database.

**The reason that why we cannot use database export function in Azure Portal** and only save backup file in Portal container instead of local disk:

The current security setting in Azure Portal preventing us doing it.

The Azure Portal database Export step only provide us two options to connect to database, SQL authentication (with one login) or Active Directory.

We cannot use Active Directory. Currently the only way to connect to database is using Active Directory MFA, and Azure Portal does not provide the MFA option for us, and it will not work by just using Active Directory Password.

SQL authentication will not work, because the export requires to access both user and master database by using one login/user, and we can’t make the same user login to access these 2 databases, master database prevent us to grant permission on any user.