سلام استاد وقت به خير. پروژه NorthwindDW من تموم شد كه 8 تا package دارد. براي استفاده از اين پروژه در هر حالت

1. راه اندازي اوليه پروژه
2. اينيشيال لود فكت از ابتدا
3. اينكرمنتال لود

فقط نياز هست يك package آن يعني Run This Only با يك پارامتر ProjectRunState اجرا شود يعني نيازي به مراجعه به پكيج هاي ديگر اصلا نيست. اجرا كردن اين Package با پارامتر 0 پروژه را از ابتدا ميسازد و شامل يك incremental load هم هست، زیرا به خاطر اینکه اجرای اولیه معمولا طول میکشد، اگر در حين اجراي پروژه ركوردهاي Dimension ها و Fact تغيير كنند، آخرين تغييرات را به Data Warehouse منتقل كند. با پارامتر 1 اينيشيال لود Fact دوباره انجام میشود و با پارامتر 2 فقط incremental load انجام میشود. توضيح مختصر اينكه Package هاي پروژه چه كاري را انجام مي دهند:

1. **Creating Data Warehouse Structure:**  
    ساختار Data Warehouse و SP ها و V\_FactOrders View را ميسازد اما index ها و relation هاي هر جدول بعد از پر كردن کامل داده هاي آن در ProjectRunState=0 ساخته ميشوند.
2. **CDC Settings:**  
    سرویس CDC را روی Northwind و جداول Orders و Order Details تنها در صورتی که در حال حاضر فعال نباشند فعال میکند. اگر این چک را انجام ندهد دستور فعال سازی CDC روی جداول ارور خواهد داد.
3. **ETL Settings Database and Other Settings:**  
   دیتابیس و ساختارها و آبجکت های ETL Settings را می سازد.
4. **Prerequisites:**  
   سه Package بالا را به ترتیب اجرا میکند
5. **FillDimensions:**  
   دایمنشن ها را پر کرده و سپس برای آنها index و relation میسازد (در صورتی که already ایندکس و ریلیشن نداشته باشند، به این معنا که بار اول اجرای این package ساخته شده و بارهای بعدی از روی آن عبور میکند)
6. **Fact Initial Load:**جدول را با استفاده از snapshot پر کرده و ایندکس و ریلیشن برای آن می سازد. بلافاصله بعد از بارگزاری دیتا در انتهای package ء snapshot دراپ میشود.
7. **Fact Incremental Load:**به صورت توصیف شده در کلاس Incremental Load را انجام می دهد.

پروژه را طوري طراحي كردم كه در business هاي 24/7 یعنی در زمان های عملیاتی نیز قابل استفاده باشد. پر کردن دیتای تمامی جدول ها در تمام Package ها، Transactional است یعنی در حین وقوع Disaster، مشکلی پیش نمی آید. Inferred record ها توسط بارگزاری Fact به طور کامل مدیریت میشوند. یعنی همان طور که فرمودید همه اطلاعات رکورد دایمنشن از سیستم عملیاتی lookup شده و در Dimension هنگام بارگزاری Fact درج می شوند و جداول Dimension ستون isInferred ندارند. نام دیتابیس Data Warehouse ای که ساخته میشود NorthwindDW2 هست تا اگر NorthwindDW داشتید برای آن مشکلی ایجاد نکند زیرا در ابتدای ProjectRunState=0 در صورتی که دیتابیس NorthwindDW2 وجود داشته باشد آن را دراپ میکند. تمام آبجکت ها در پروژه ساخته میشوند بنابراین در کنار این پروژه فقط به دیتابیس Northwind در حالت اولیه نیاز است و اینکه Connection Manager ها دوباره تعریف شوند. برای Package ها Annotation زیادی وجود ندارد چون اسم آبجکتها بسیار توضیح دهنده است.

DimDate در حين ساخت Data Warehouse ساخته ميشود كه شامل تاريخ شمسي هم هست و اگر Named Column رو تاريخ شمسي انتخاب كنيم ميتوانيم در گزارشات تاريخ شمسي را نمايش دهيم.

مشکل بسیار نادری وجود دارد که به خاطر صرفه جویی در وقت، آن را هندل نکردم. در صورتی که سطور فاکتور یک سفارش (Order Details) تغییر کند (که معمولا تغییر نمیکند) و مشتری آن نیز Historical Arrtibute خود را (کشورش) تغییر دهد و با تغییر فاکتور، سیستم عملیاتی نیز OrderDate فاکتور را تغییر ندهد، مشکلی در بارگزاری incremental فکت بوجود می آید که بسیار بسیار نا محتمل است. البته اگر سطور فاکتور تغییر کند به عقیده من OrderDate باید تغییر کند که در غیر این صورت مشکل طراحی اپلیکیشن سیستم عملیاتی است.

ضعف این پروژه نیاز آن به دسترسی sysadmin روی سیستم عملیاتی هست اما اگر ساختن snapshot را حذف کنیم (كه به راحتي قابل انجام است) ProjectRunState 1 و 2 که نیاز اصلی ما هستند، بدون نیاز به دسترسیِ sysadmin، در سیستم عملیاتی قابل اجرا هستند. از سیاست مایکروسافت مبنی بر اینکه dbowner یک دیتابیس که sysadmin نیست نمی تواند snapshot بسازد تعجب میکنم و به نظر من منطقی است که کاربری که به همه اطلاعات یک دیتابیس دسترسی کامل دارد بتواند بدون sysadmin بودن در سطح سرور از آن snapshot بسازد. البته كه در ديتابيس عملياتي معمولا به ما دسترسي dbowner نمي دهند. اين مطلب را همين جوري خدمتتون گفتم.