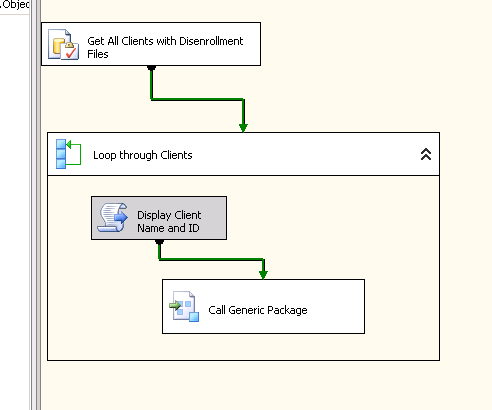
# Death and Disenrollment

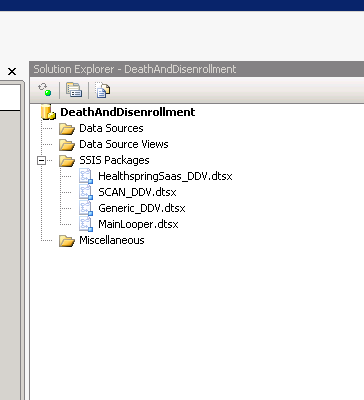
The current structure is as follows:

There is a solution called DND in the artifacts folder. The main package here is called MainLooper.dtsx. It performs the actions listed below.



**Figure 1.** Looping Package – Retrieve all clients marked as “Disenrollment” from the ETLAutomation.dbo.ClientParameters table

The looper package does just what you would expect –it looks through each client and calls the generic package within the solution (Generic\_DDV.dtsx).



**Figure 2.** Solution Packages

There are currently 2 different breeds of disenrollment files that we receive (even though the spec says only one). I’ll detail those below.

**Breed 1: SCAN (3 columns – no header). For SCAN, D is for Disenrolled and E is for Expired/Deceased**

*Sample Data*

*(MemberID)|(DisenrollmentType)|(Date of Disenrollment) – not client provided*

4002615XXXX|D|2017-09-30 23:59:00.000

**Breed 2: Healthspring (5 Columns – with header). For Healthspring, 02 is for Disenrolled and 01 is for Expired/Deceased**

*Sample Data*

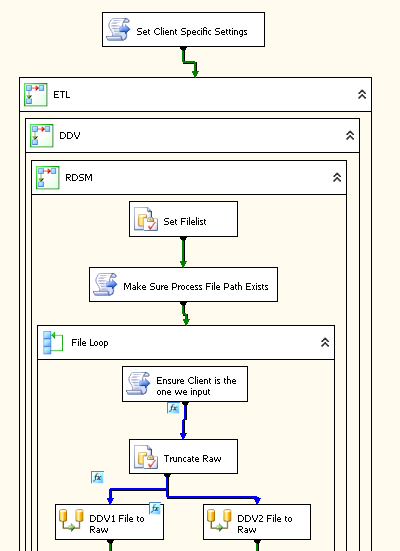
MemberID|ContractID|OptOutCode2017|DisenrollmentDate2017|EnrollmentDate2017

774587753\*01|S5617|01|2017/08/19|2017/08/08

780202273\*01|S5617|01|2017/07/25|2017/07/06

There are a few quirks within the Generic package. There is a match to the Patient table to retrieve PMD\_PatientID. We currently due this based on the distinct PMD\_ClientID that is hardcoded due to possible space concerns. We may remove this dependency in the following so this can be truly “generic”.

NOTE: The Generic package takes a ClientName variable to be used to simplify processing. The final resting place of this data currently is in a “rdmDB” “Spec” table that is ready to be proessed into MedPro.



**Figure 3.** Sample flow of the Generic Package

**QUICK NOTES: Added on 2/14/2019 (Brian Williams)**

* Package Location: [\\sqlrdp\SSIS\ETL\DeathAndDisenrollment](file:///\\sqlrdp\SSIS\ETL\DeathAndDisenrollment)
* The SQL Agent job runs from the “PRODUCTION” folder path under the above so you have to deploy/copy paste to make it active for the job itself. You can run thru Visual Studio without fear of
* @PMDClientiD:Disenrollment:1 (clientid:name:value) pair needs to be put into ETLAutomation.dbo.ClientParameters
* Add mapping entries (client-specific) to DisenrollmentReasonsCD\_Client on ETL
* Population Source Map needs to be created in New Populations to the population for the @PMDClientID
* rdsmDDV tables need to be made for the @PMDClientID rdsm database in ETL (copy from either rdsmScan or rdsmHealthspring)
* There is a Population schema on ETL in the rdsm database that pulls this data (Joel will describe later – still learning this piece)

Things to note:

There are 2 main disenrollment reasons we are concerned about: Death and Disenrollment. These are often conveyed as “D” for Death and “E” for Expired or Disenrollment but we often need confirmation from the client.

We often get this value as a 1 or a 2. There are CMS specific values for these but even with those, we have to ask. You can use the tables DisenrollmentReasonsCD and DisenrollmentReasonsCD\_Client to look up what values clients use. There is also a VIEW called vwDisenrollmentReasons on ETL that combines this all into a compact view.