Month-End Reporting
Visual Studio Processing

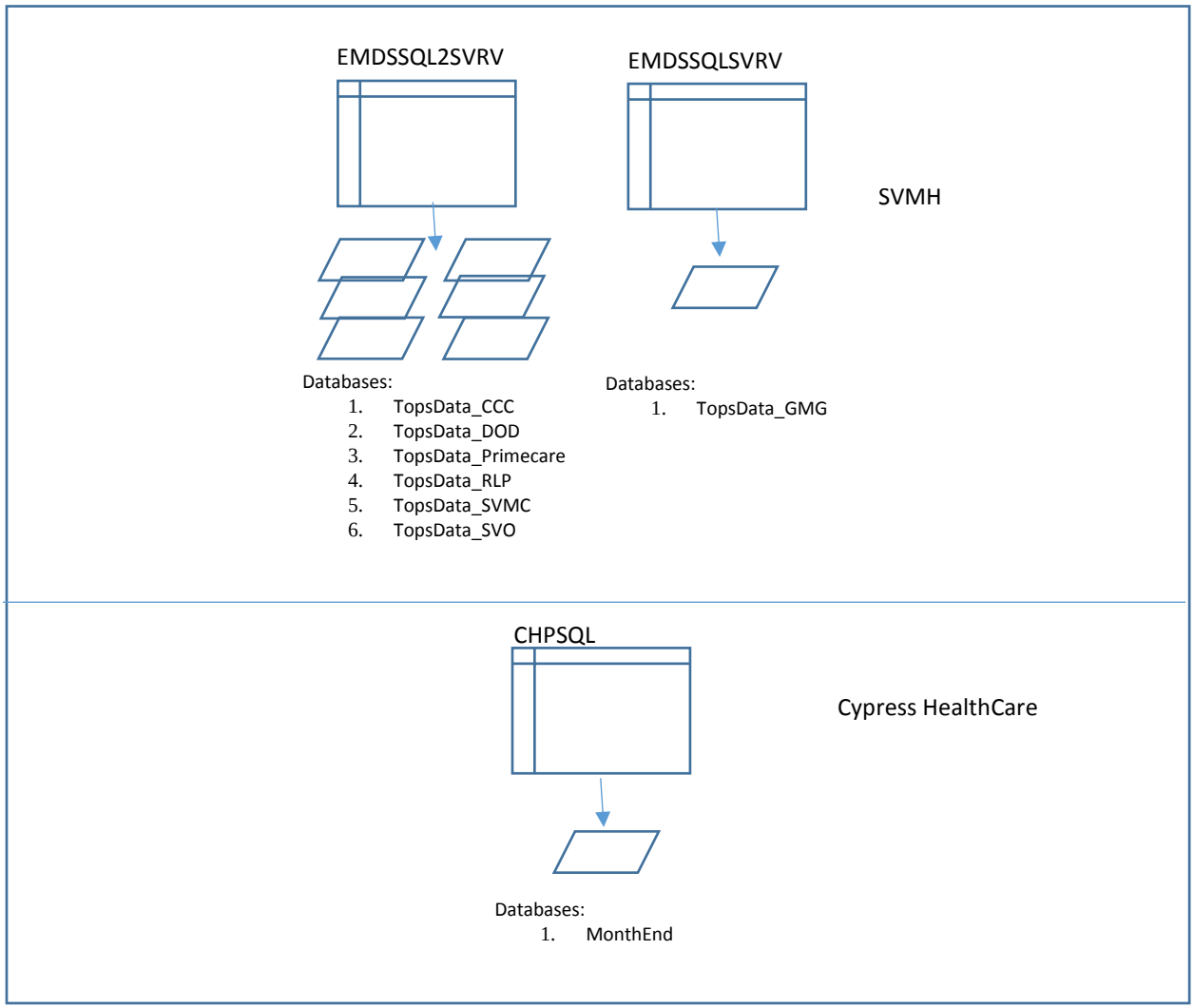
Overview

As of this writing (11/28/2017), the Salinas Valley Medical Hospital (SVMH) SQL (S)ervers, EMDSSQL2SVRV and EMDSSQLSVRV, house the information that Cypress Healthcare Partners (CHP) uses for its management functions and that is required for its (CHP) month-end reporting.

The servers are architected such that each medical practice or group, has its own database. The data model for each database is similar to all others and the patient/billing/provider information that CHP uses is stored in each database consistent with the data model.

But because CHP does not own the servers that contain the management/reporting information, and because the EMDSSQL servers exist on the SVMH network rather than the CHP network, a Reporting SQL-Server has been set up on the CHP network—server CHPSQL—so that CHP can craft the processes it needs so as to make reporting and use of its information, more effective.

Diagrammatically, the Month-End Reporting network looks like this:



Initially, the SVMH Network was architected by SVMH I/T working in conjunction with CHP, and data was extracted through a series of Crystal Reports that were crafted from CHP specification by SVMH I/T. Each database has its own series of reports and all reports across the databases operated similarly. They are in-effect, mirrors of one another

At the beginning, reporting was effected by executing the Crystal Reports (cr). The reports would be executed on each database and the result would be parsed into various spreadsheets that were the mechanism for CHP reporting.

There are 4 basic report types that serve most of CHP's reporting requirement:

1. Executive Summary – This report summarizes charges, payments, adjustments and the a/r balances for each practice group and providers that bill within that practice group
2. Coding Profile – These reports focus on the CPT billing code associated with invoice line items
3. Work RVU – These reports focus of the Work RVU associated with invoice line items. It is common practice for service providers to be paid based on the Work RVU associated with a billing item and therefore, an important source of information for the providers that CHP manages
4. A/R Aging – These reports focus on receivable collection and aging

The Current Mechanics of Reporting

The architecture of the SVMH databases does NOT include Normalization of the data model as is the usual practice in a relational database environment. Rather, each database is structured after a business fashion—loosely normalized, and tables are then loosely integrated pieces of the business rather than a 3rd Normal Form object.

Because of the unique architecture of the databases, the current reporting model uses the SQL from the above described Crystal Reports in order to maintain consistency from the originally established (and presumably, vetted) reporting process to the newer model. Quite simply, the SQL was extracted from the above report types and configured to minimize "parsing" of data to be reported. In each (database) case, the SQL is crafted to contain selection criteria that is appropriate to the report and to the environment. As an ongoing maintenance concern, it is important to understand the peculiarities of the database models and the "fit" of the Crystal Reports SQL As long as the EMDs server/reporting environment is used, the vetted report SQL will be the best way to accomplish CHP's reporting goals

Along with the data model concerns, because CHP reporting emanates from two networks, there is a "bridging" issue that is also part of the reporting consideration. CHP is unable to install custom data objects or software in the SVMH network. Tables that reflect the CHP reporting model/needs and/or tools to create reports, have to exist in the CHP environment. Which means, the data used for reporting needs to be transported from the environment where it is housed, to the environment from which, it is reported. Simply speaking, the way transport works is this

1. The CR report SQL is executed across the 6 database environments
2. The extract results are copied to and excel spreadsheet that is saved to an SVMH folder that mirrors a reporting folder in the CHP environment
3. Data is saved to a particular folder that reflects the month-end reporting directory and the reporting month. For example, Coding Profile information extracted from TopsData_DOD is saved to: documents/monthend/codingprofiles/dod/October.
4. Each spreadsheet is named consistently for automation purposes. Coding Profile information is save to CodingProfile.xlsx

5. It is important to maintain the Month folder and spreadsheet name formats so that the automated process of the new reporting model can access and use the extracted reporting data
6. CHPSQL is set up so that the SVMH/documents environment is a virtual attached drive. CHPSQL drive Y is the //192.168.2.7/documents/monthend/ drive that the extracted data was save to
7. Because drive Y is a virtual drive, automated processes for accessing the data do NOT recognize drive Y and as a result, the drive Y data must be transferred to a CHPSQL drive that is NOT virtual in order to use the data for reporting
8. To aid in the transfer process, the Visual Studio MonthEnd Desktop application has two functions to help streamline the mechanics of creating storage and reporting folders
    a. Open MonthEnd from the desktop of CHPSQL and select MonthEnd Processing
        i. Create Working Folders – detects the current month and creates working folders in the Y. drive for the previous month (the month to be reported)
        ii. Copy Data Folder Files – Once the extract data is saved to the Y:/monthend/process/month folders, Copy Data Folder Files will copy the newly created Month folders and the file they contain, to the E:\Monthend\Monthend\MonthendSouce\ folders that will allow further processing by the MonthEnd desktop functions
9. Once the data has been transferred to the Monthend\MonthendSource\ directory, the data can be loaded into the reporting tables for further processing through the Load Data Tables function on the MonthEnd Processing menu

Data/Report Types

1. TTM Data – Trailing Twelve Months. wRVU data group into 5 categories by CPT code: 1) New, 2) Established, 3) Inpatient, 4) Delivery and 5) Other. An example is: wRVU\TTM\Primecare\October\AbundisRTTM.pdf
2. wRVU Patient Detail – wRVU detail by Patient (for the reporting month) : wRVU\PatientDetail\Primecare\October\AbundisRPatientDetail.pdf
3. Coding Profile – monthly patient “encounters” by CPT codes 99201 – 99215: CodingProfiles\DOD\October\DanielsMCodingProfile.pdf
4. Missing Ticket – SVEMG ER visits for the month MissingTicket\ERVisits\October\SVEMG ER Visits excluding LWBS.xls
5. Missing Charges – SVEMG listing for ER visits that have been charged. The result of the above report, compared to this report, gives a listing of patients who have not been charged for their ER Visit
6. SVMC Encounters – Patient encounters for the month by department and/or specialty: Z:\SVMC\Month End Dashboard Reports\SVMC PT Encounters by Specialty\SVMC PT Count by Specialty FY 2014 2015 2016 2017 2018 103117.xlsx

Processing

1. Data is loaded into its respective table by a stored procedure called spReadInsertExcelData (see SQL Server database Monthend\programability). Both the data type and the company code are passed to the proc which in turns reads the appropriate excel spreadsheet and loads the spreadsheet data into the appropriate table
2. Once the needed data is loaded click on the appropriate MonthEnd Processing menu button to create 1) TTM, 2)wRVU Patient Detail and/or 3) Coding Profile reports
3. MissingTicket/Missing Charges – Once the Visits and Charges data is loaded as above, execute store procedule