

Sharp Community Medical Group Interface Engine Migration

The Diverse Client

Sharp Community Medical Group (SCMG), a large independent physician association (IPA) serving the greater San Diego, California region consists of 207 Primary Care Providers (PCPs) and 520 specialists representing more than 30 specialties. SCMG's physicians are located at 350 locations in the county, with 23 urgent care centers and 11 local hospitals, including all Sharp HealthCare, Palomar Health and Rady Children's hospitals.

Sharp Community Medical Group	
# PCP Providers	207
# Specialty Care Providers	520
# Specialty Services Offered	30
# Locations	350
# Urgent Care Centers	23
Serving # Hospitals	11

The Crucial Crossroads

SCMG leveraged Allscripts TouchWorks™ EHR to provide optimum healthcare for their diverse patient populations. The legacy interface engine, ConnectR consisted of 67 interfaces that served as the backbone for integration to community providers. There were several mitigating factors supporting the need to replace ConnectR which was recognized as an outdated software that had not received updates or support for a number of years. Allscripts strategy to consolidate integration across their suite of solutions, Common Interface Engine (CIE) was introduced as the future platform offering more modern and advanced integration capabilities between outside applications and the Allscripts Touchworks™ EHR.

However, the main motivation to sunset ConnectR and introduce CIE was the end of standard support for MS SQL Server 2008, which was the last supported version for Connect 4.6 and subsequently the final supported version of ConnectR. As a result, ConnectR would be deemed a

non-HIPAA compliant interface server. Furthermore, GE's announcement of the official sunset of ConnectR made the need to migrate to a more modern, flexible interface engine inevitable.

The Uniquely Positioned Partner

Galen Healthcare Solutions had a long standing partnership with SCMG ranging from various interoperability projects, reporting initiatives as well as conversion work. Familiarity with SCMG systems, combined with years of experience with ConnectR and Rhapsody certification positioned Galen as more than capable partner for this ambitious project.

An Ambitious Undertaking

While able to fulfill the basic integration needs of SCMG, ConnectR was always more of a limiting factor. Migration to CIE offered SCMG a tool the organization could utilize to add technical flexibility. SCMG chose to migrate to the Common Interface Engine (CIE) which is a rebranded version of Orion's Rhapsody Integration Engine.

Over the course of a two year period Galen worked closely with the SCMG team to accomplish the following goals:

- Migrate all existing interfaces using a proven phased approach to ensure a seamless transition
- Collaboratively develop consistent interface designs that adhere to a newly created standard development approach
- Provide current & future state documentation as well as proposals to remediate existing issues identified in the legacy interface builds
- Tailor onsite trainings to best suit SCMG's staff current skillsets to transfer knowledge of how to maintain and develop additional interfaces in the future
- Develop 11 net new interfaces during the course of the migration, including several custom interfaces with the dbMotion Health Information Exchange (HIE)

The overall goals not only incorporated migration of the SCMG interfaces from one platform to another, but also planned for a more auspicious integration experience. Thus leaving SCMG in the position to support their integration environment independently at the conclusion of the project while concurrently ensuring business integrity on a day to day basis.

GALEN'S MIGRATION PROCESS

1. Review and document existing integration and workflow.

Capture business requirements, filtering and logic.

2. Assess, analyze and gather existing interfaces and transactions.

Identify objects and principles that directly translate to the new interface engine paradigm.

3. Design and develop interfaces based on requirements.

4. Validate and test regression.

Setup parallel feeds, validating interface output by a "diff."

Target application content and workflow validation.

5. Go-Live

Ensure connectivity (firewall) between source and target applications.

"Cut-over" to the new interface.

from vendor UAT resources. To mitigate some of the impact vendor roles in the UAT process was reevaluated. For instance dedicating one week of intense testing with a vendor received better responses and turnaround times as opposed to several weeks of light testing. Additionally certain portions of the UAT were suitable and more efficient with testing that did not involve the vendor. For example, a result interface can be heavily validated using sample messages from production while an order interface requires more involvement from the vendor to ensure the new order messages are processing correctly in the vendor system. The UAT process was one of the most time consuming aspects of the project. Careful project management, allocation of staff, and communication was utilized to meet project milestones.

Crucial Success

The SCMG team provided skilled project management expertise as well as detailed validation that was essential to ensure a seamless rollout to providers with little to no effect on end user workflow. Galen's agile consultants adapted to work successfully with each facet of the project that included multiple vendors, each with varying objectives, skillsets, and technology standards. Even with competing organizational priorities both SCMG and Galen were able to flex as needed when the project schedule was impacted. Collaborative interactions between SCMG and Galen resources transformed the relationship into one strong team with a unified approach.

Careful Management of Dynamic Challenges

Striking the right balance between Development and User Acceptance Testing (UAT) resources to consistently progress was challenging at times. This was generally a result of staffing proportions of resources and lack of responsiveness