UCLA's New Data Distribution Paradigm

Submitter: Saheli Datta

Business Analyst, Student Affairs Information Technology, UCLA

sdatta@sait.ucla.edu

(310) 206-3657

Project Core Team:

- ❖ Saheli Datta, Business Analyst, Student Affairs IT
- ❖ Rosanna Carrera, Business Analyst, Student Affairs IT
- ❖ Martin Bjel, Manager, Database Administration, Student Affairs IT
- ❖ Praveen Dugar, Manager, Application Development, Student Affairs IT
- ❖ Curtis Fornadley, Manager, ESB Program, Information Technology Services
- ❖ Eddy Mishiyev, Programmer Analyst, Student Affairs IT
- ❖ Donny Morada, Manager, Business Analysis, Student Affairs IT

Introduction

The new data distribution paradigm is a modern, enterprise level solution for UCLA's data needs and greatly reduces the need for customized configurations and point-to-point solutions in application development. It is a landmark accomplishment for the campus and paves the way for an open innovation ecosystem by deploying an extensible architecture. It exposes highly standardized, reusable Application Program Interface (API) services that greatly simplify data integration with web and mobile applications. As part of this collaborative effort across IT departments, a campus registry for these services was also deployed, that makes the discovery of these API services easy and improves operational efficiency by delivering the means to strengthen engagements with campus partners in fulfilling their data access needs.

Application Access and Integration

UCLA has a highly decentralized IT development and support model that has served the campus well over the past several decades. It encourages and supports decentralized development with centrally provided infrastructure fostering agility and innovation across the campus. In this decentralized structure, applications are developed by local IT units to meet business needs that frequently integrate with student data. Consequently, student data retrievals and updates frequently occur in departmentally managed applications such as the student portal (MyUCLA), staff portal (SAWeb), Counselor Desktop, and Course Learning Management Systems (CCLE), to name a few.

UCLA has a well-established Operational Data Store (ODS) with near real-time replicated Student Information System (SIS) data from the mainframe for Student Records and Admissions. Overnight refreshed data is also available for Financial Aid and other Student related domains. The ODS has

successfully served as the hub of data distribution for campus applications by aggregating various student data sources into a central location readily accessible by support staff. As shown below, campus applications connect to the ODS via point-to-point solutions, such as stored procedures, views and other SQL objects to query data.
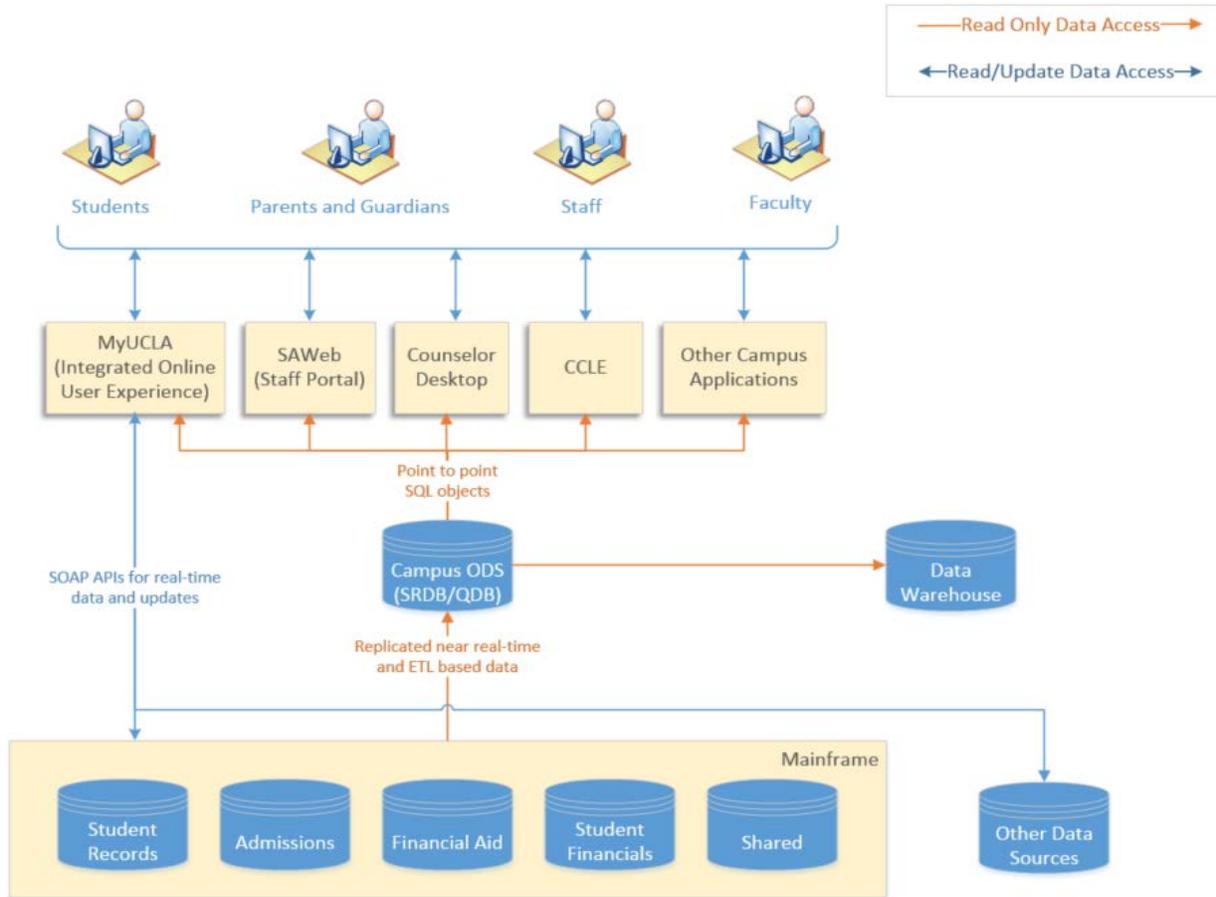
Campus Need for a New Paradigm

As campus demand for student data has expanded over the years, it has resulted in an exponential growth in the number of point-to-point solutions and SQL objects that need to be maintained for continued application development and support. This has led to redundancy, expensive integration and maintenance processes, and increased operational IT costs over time.

The lack of business and technical documentation around these point-to-point solutions has resulted in considerable hand-holding by IT teams providing this data. In addition, there has been no concerted and comprehensive effort to review data governance or standardize data definitions across campus. For example, there is still a lack of a shared definition of a 'student' among campus academic and administrative departments.. Departments regularly adopt data feeds for directory and other campus services as the definition of a student, often leading to students getting denied access to critical student services. Since campus departments are now increasingly making their services accessible through web

and mobile applications, the need for a new data distribution paradigm was recognized as even more acute.

The New Data Distribution Paradigm

To address this need, UCLA has developed a comprehensive set of API services for student data for data access and application integration. The API services are self-contained, loosely coupled and highly reusable units of functionality that can be combined to fulfil the business process requirements of large, complex web applications. Student Affairs and IT Services have collaborated to route and secure the APIs via the campus Enterprise Service Bus (ESB) Gateway, and deliver much needed documentation around canonical data models and consistent data definitions via a centralized Service Registry. In this way, the APIs enable business objectives of campus departments as well as help eliminate arbitrary application/data silos, thereby providing a true enterprise data distribution solution.

The RESTful APIs developed by UCLA are a standardized way for campus departments to access SIS data from a centralized location and integrate their web based applications using a single Uniform Resource Identifier (URI), standard HTTP methods, and Internet media type for the dissemination of data and hypertext links to navigate to various components of the data. The APIs currently expose Student Records and Financial Aid data along with a subset of Common Module data to users in a secure manner, thereby greatly reducing the need for point-to-point solutions. A comprehensive and new underlying infrastructure has been put in place to support the enterprise data distribution solution. A subset of the infrastructure components is highlighted below.

Service Oriented Architecture (SOA)

Service Oriented Architecture (SOA) is an architectural design principle focusing on business processes and their packaging into reusable, interoperable services. The adoption of SOA in designing UCLA's RESTful APIs ensures that their design and applicability is extensible and scalable across multiple campus departments. They are designed to be highly reusable and can support new and changing requirements by leveraging the functionality of existing services. UCLA's RESTful APIs deploy a standardized service contract; this means that the API signatures and naming conventions are consistent across domains. This generalized enterprise solution to UCLA's data distribution challenges is designed to augment business and technology alignment, increase interoperability and organizational agility, and reduce IT burden over time.

Enterprise Service Bus (ESB)

An Enterprise Service Bus (ESB) is a middleware component required to implement SOA. Generally speaking, an ESB is a mechanism to manage access to services via a simple and consistent interface to all end-users. UCLA's RESTful APIs are routed via the campus ESB gateway, providing modularity, service orchestration as well as supporting synchronous and asynchronous interactions. UCLA has a well-established integration hub comprised of a locally hosted ESB based on Red Hat JBoss Fuse 6.2.1. The various technologies comprising Fuse are all based on open source products. Fuse provides a basic tool kit for building secure services, routing, data transformations, and asynchronous message queues. The APIs are managed via the ESB's Enterprise Service Manager (ESM) Gateway, an in-house developed gateway that routes the services and manages service security and access with OAuth2 Client Credentials, as shown below.
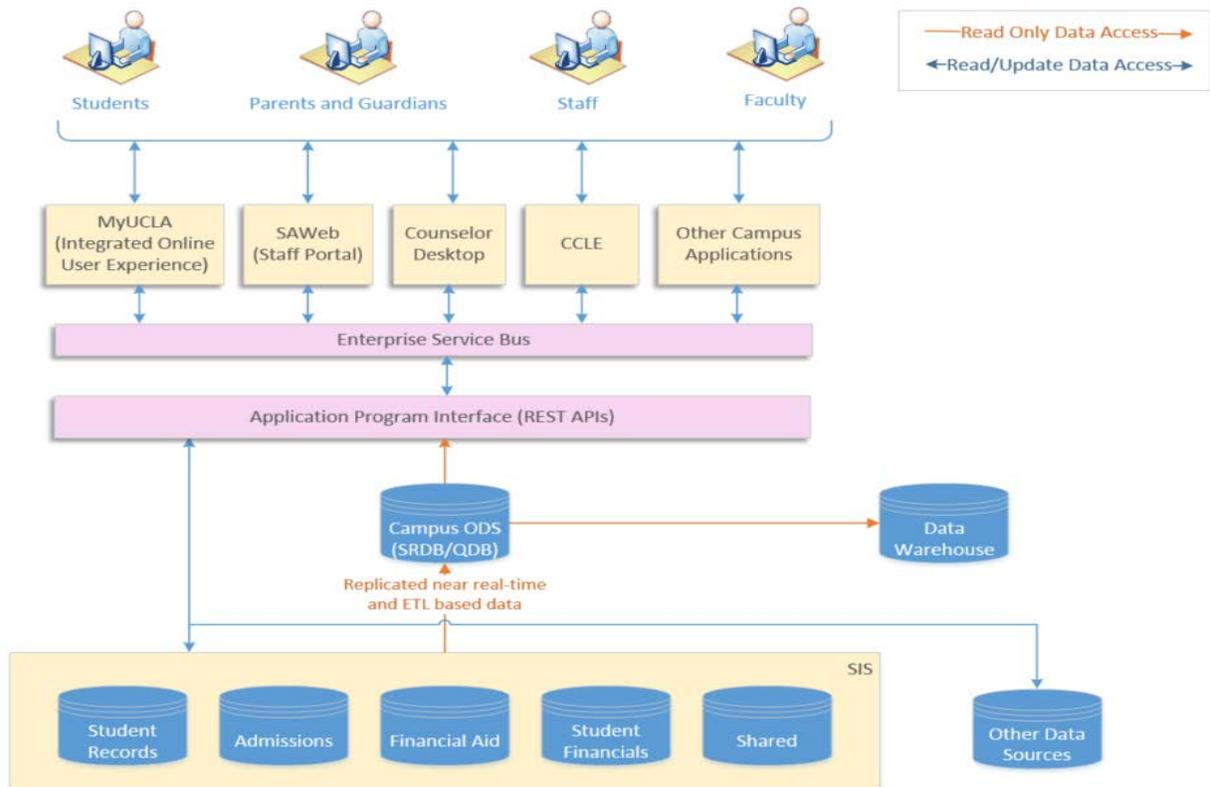
Service Registry

UCLA's Service Registry (api.ucla.edu) serves as a central repository of all RESTful APIs in the SIS as well as MyUCLA application domains, where campus departments can discover and request access to the services. The Registry is supplemented with communicative metadata about the APIs, by which they can be effectively discovered and consumed. It also allows the API services and their capabilities to be easily identified for effective reuse by existing and potential service consumers, thereby positioning them as true enterprise assets with an increasing return on investment.

Student Records APIs have been published at api.ucla.edu, while Financial Aid and Scholarships APIs are being finalized. Analysis and development of APIs for other domains is currently underway. The bulk of the effort at UCLA has so far focused on query APIs (GET), while the design of APIs for transactional operations (PUT, POST, DELETE etc.) is also underway.

GET	/v1/Students/{uclaID}/Names	GET	/v1/Students/{uclaID}/Veteran
GET	/v1/Students/{uclaID}/Addresses	GET	/v1/Students/{uclaID}/TranscriptCareers
GET	/v1/Students/{uclaID}/Emails	GET	/v1/Students/{uclaID}/MajorProgramOfStudy
GET	/v1/Students/{uclaID}/Phones	GET	/v1/Students/{uclaID}/MinorProgramOfStudy
GET	/v1/Students/{uclaID}/Privacy	GET	/v1/Students/{uclaID}/SpecializationProgramOfStudy

Campus Adoption and Impact

UCLA's RESTful APIs for the Student Records and common student data domains went live in August 2016, followed soon after by the Financial Aid & Scholarships APIs.. UCLA is in the midst of creating similar RESTful APIs for other modules of the SIS including Admissions and Student Financials (Billing and Receivables), with services planned in collaboration with IT Services for transactional APIs to update student data in real-time.

The implementation of UCLA's new data distribution paradigm has been highly successful with an increasing number of new and existing campus departments adopting APIs as their preferred method of consuming SIS data on campus. Its success is reflected by the fact that as existing consumers of student data are moving away from point-to-point solutions for their applications, the adoption rate of the RESTful APIs is skyrocketing. In the month of December 2017, for example, there were approximately **3 million API calls** from just a few campus departments that use student data extensively. More notably, the demand for continued enhancements to the APIs has been steadily growing.

Another significant impact of the new API paradigm has been to implement uniform data definitions and nomenclature across campus as a result of an unprecedented effort to standardize and document API service contracts. One example is the use of "uclaID" in all SIS APIs as a standardized service contract name for the 9-character data field currently represented in the underlying data source by several names like UID, Stu_ID, UCLA_ID etc. This ensures consistency and reliability for data consumers, and has the potential to help the campus make a smooth transition to a new SIS implementation in the future even if the underlying data layer changes.

The new API paradigm has also been instrumental in creating a standardized, shared framework to help define what it means to be a "student" at UCLA. Instead of having to rely on disparate data sources, campus academic and administrative departments can now access and interpret relevant information within this centralized, scalable *Student Definition* framework to construct a definition of student that matches their business requirements. Departments then use this data as an identity and access control mechanism to grant or revoke access in a timely and accurate manner to any number of their web applications that provide crucial digital services which are critical to student success at UCLA.

The new API paradigm exposes the campus' core data distribution services in a modern interface for easy discovery and consumption. At the same times, it creates an extensible and scalable architecture for providing student data that can be leveraged by various campus departments to integrate their own functionality. The following feedback from Christopher Spreitzer, Director of Undergraduate Education IT, reflects the reception of the new RESTful APIs on campus:

"The API is a huge step forward for application data integration at UCLA and provides a secure, robust, and easy to implement environment. We see this model as a way to "future proof" our applications since our contract is now with the API. Any changes to the underlying data sources (including, for example an impending migration to a new SIS), should insulate our applications from any major retrofitting. This is a significant improvement over our legacy point-to-point database object (views, stored procedures, etc.) based solutions. The built in documentation for the API which follows strict universal standards, allows us to better understand the data so that we can request the needed access from the data owner (such the Registrar's Office) with much less consultation and we can more quickly implement opportunistic, composite applications that enable new and exciting capabilities and save time for staff, faculty and students."