SIS data integration

A Student Information System (SIS) is an essential, enterprise system in the educational technology landscape. It is the system of record for a broad range of data that are pertinent to data science, research, and learning analytics. Consequently, SIS data is a required part of the overall UDP implementation.

This document describes high-level details of an SIS context data integration with the UDP. For an in-depth description of an SIS data integration's nuts and bolts, please review our technical documentation on writing a UDP context data integration.

SIS loading schema

A UDP loading schema is the conceptual anchor of a context data integration with the UDP.

The UDP's SIS loading schema is developed and maintained by Unizin and its institutional community. It defines the full extent of SIS data that can be integrated into a UDP instance.

All institutions are expected to support the latest version of the SIS UDP loading schema. It is an essential part of a UDP instance and UDP implementation.

Entity requirements

The SIS loading schema is unique from other UDP loading schemas insofar as it categories its entities in three priority classifications: required, strongly recommended, and recommended:

- **Required.** These entities are required from the SIS for any UDP implementation. The UDP cannot import and join LMS and other LTI tool data into the UDP without these entities from the SIS.

- **Strongly recommended**. These entities are almost always needed for research, data science, and learning analytics use cases. Consequently, Unizin strongly recommends that institutions include these entities in their SIS data integration.

- **Recommendation**. The need for these entities varies by institution. Institutions may add data for these entities over time as needed.

As a starting point, Unizin's recommendation is that Academic institutions implementing the UDP should provide the "strongly recommended" entities in addition to the required entities.

Scope and frequency

We strongly recommend that you generate an SIS context dataset every night. Please note that the UDP currently only accepts full historical datasets. Consequently, Academic institutions will need to reproduce the history of SIS data they wish to be available in the UDP on a nightly basis.

Frequently asked questions

What values are allowed for the Course section enrollment entity's "role" and "status?"

The "role" of the Person in the Section and the "status" of that Person in their role are option set values. No other value can be used other than those of the option set. At present, the two values are required for a Course section enrollment record.

Is instructor data tracked anywhere within Unizin?

The Person entity in the UCDM can be used to model data for any kind of individual. Persons may have different relationships (to course sections, say) that are reflected by their "role" and "status."

The Unizin data feed specification describes the data elements, but not the data population. Should we feed data for the current term? Any future term data? Any past term data?

The breadth and historical depth of SIS data sent by institutions varies. Our recommendation is that institutions send SIS data for at least 5 years of history or until the beginning of their use of their current LMS, whichever is *longer*. Generally speaking, longer historical time horizons are attractive for longitudinal analytics and research.

How have other multiple-university systems populated the Unizin "Campus" entity? Do we want this to represent our four Universities or use it for something else?

All institutions, regardless of their number of campuses, should populate the Campus entity. We *strongly* recommended that all multi-campus institutions populate this entity to include all of their campuses. Rolling up data and analytics by campus is often essential to reporting, for example. We also recommend that you properly model all Campus-based relationships (e.g., Academic organization, Course offering, Facility). Those relationships express the dimensions along which data can get rolled up by Campus.

Should "Person" data be unified or differentiated? If unified, how should data conflicts be resolved? Does the Home campus wins on a conflict? If unified, how should FERPA be handled for students who have differing values? What would be different about FERPA? The restrictions applied to the student? Do we would carry through the most aggressive protection?

A*t present*, we do nothing with the FERPA directory block value except capturing it from institutions. It does not change the way we present information in the context store. Nor is it excluded from the data that can be seen by users in the context store. Unizin does not publish the data anywhere. It's only stored in the context store. We capture the FERPA directory block variable mainly in case we do, in future uses of the data than just capturing it, need to use it to selectively present data.

There are several fields where the four universities all use the same values, sometimes to mean different things. Should these be unified or differentiated? If unified, how should data conflicts be resolved? How have other systems within Unizin addressed this conceptually?

Everywhere data can be unified across campuses, it should be unified across campuses. For example, a single set of values for Academic careers should be pursued. The goal is to let the relationships explain where the data comes from.

Composite identifiers can be useful to differentiate records from different campuses. This is acceptable and even routine for entities like Course offerings and Course sections, where the identifiers need to be unique across the whole system.

If data is to be differentiated, how should that be done? Prefix the data with a campus prefix? (C for MU, K for UMKC, R for S&T, S for UMSL) Some other method? This might be dependent upon the answer to the first questions.

Composite identifiers will likely be required for certain entities, like Course offering and Course section, where there are common codes across campuses. Every record in a dataset must be unique along some grain of data. If you're aggregating data across multiple campuses, you may need to use campus-specific markers in the identifiers to make the datasets unique.