DRAFT
University of Wisconsin-Madison
Proposed Student Data Integration Policy

Policy: *Systems that interact with the Student Information System (SIS), either as consumers or providers of Student data will conform to the following principles.*

Premise

Develop and enforce a set of principles that will guide data integrations for SIS and beyond. These principles necessarily couple technical integration patterns and data modeling practices for system to system integrations. Populating and managing data warehouse data is not addressed by this policy.

These principles consist of extensible ideas that allow UW-Madison to expose SIS data to SaaS and on premises solutions in a consistent way and guide the future development of systems like CAOS and other middleware.

The practical application of these principles should insulate the roughly 100 systems connected to SIS from integration problems arising from the SIS 9.2 upgrade effort and beyond.

Integration Principles

---

1. Shift away from building monolithic systems that synchronize, warehouse, and expose data (e.g. CAOS)[1] to collections of independent, reusable, purpose-built services
2. Simplify integrations for connected systems that consume data by establishing a common platform for collections of independent services (e.g. an API manager)
3. Enable shared contributions to the platform that allow developers with domain expertise to contribute independent, purpose-built services
4. Develop common Enterprise Business Objects (EBOs) that can be leveraged across independent services so that standard definitions of business data can be delivered regardless of technology or integration pattern. EBOs should be developed using the process outlined by Office of Data Management & Analytics Services and documented in the "Data Cookbook"

   Commented [SG1]: May want to link to this

5. Develop instrumentation to measure things like performance, usage, and security events that occur across the collection of independent services
6. Use instrumentation to inform microservice and EBO lifecycle management

---

[1] From A2D Guidelines, Application and Data Integration Strategies

University of Wisconsin-Madison
Proposed Student Data Integration Policy

Policy in Action

In this example, we assume a system like AEFIS needs data that is sourced from a variety of systems. Instead of putting the burden on developers to extract data from each source system, transform it, and load it into AEFIS in a way that is consistent with other applications across campus, the following model is proposed:

By using a collection of independent services, fronted by a common platform, developers can leverage data from multiple discrete sources without addressing the complexity of synthesizing the data themselves.

A service like AEFIS can query the common platform that passes data back from various independent services that connect to source systems like CourseLeaf, SIS, and others. Each of the independent services exposes data housed only in the respective source system, and the platform synthesizes it to appear the way AEFIS is expecting (e.g. aggregation of learning outcomes, student information, term data, etc.).

Complexity is abstracted by the platform, and independent service development draws on domain knowledge and possibly staff resources from the parties that manage each system. In this way, no single team needs to be experts in learning outcome, student, term, and other data needed by AEFIS.

This model enables re-use of components that expose data from source systems in a way that is consistent with EBOs across the complete portfolio of campus applications.

Finally, using instrumentation, owners of any given independent service can use metrics to manage independent service lifecycle, defined by usage, technology, operational costs, benefits of using the service, etc.

University of Wisconsin-Madison
Proposed Student Data Integration Policy

Access to Student Data: Glossary

AEFIS

Assessment, Evaluation, Feedback & Intervention System. A web-based assessment management solution that facilitates the collection and application of real-time assessment data.

CAOS

Formerly CHUB. Curricular and Academic Operational Data Store. A set of web service method calls available through the ESB that provides curricular, roster, and student data to the authorized recipient systems. Data is organized under two areas - Curricular and Class Roster.

CHUB

Curricular HUB. Now known as CAOS.

EBO[2]

Enterprise Business Object. A standard definition for a business object (e.g. a course, a student, a purchase order, etc.) in terms of reusable data components.

ESB[3]

Enterprise Service Bus. A layer between applications that enables communication regardless of language, framework, or data interfacing technique (e.g. SOAP, REST, etc.).

ETL

Data Extract, Transform, and Load. The set of activities performed when interfacing data between systems, typically as part of a bulk or batch process; manipulation of data format.

> **Commented [SG2]:** Acronym not used

LO

Learning Outcomes. Statements that describe the measurable skills, abilities, knowledge, or values that learners have achieved, and can reliably demonstrate at the end of a course or program. LOs are learner-centered rather than instructor-centered.

> **Commented [SG3]:** Acronym not used, but could title as LEARNIG OUTCOME since that is used above

Microservices

A form of service-oriented architecture style wherein applications are built as a collection of discrete smaller services rather than one monolithic app.

SaaS

Software as a Service.

SIS

Student Information System. UW-Madison's branding of Oracle PeopleSoft's Campus Solutions.

---

[2] https://docs.oracle.com/cd/E17904_01/doc.1111/e17363/chapter02.htm#FPCON168

[3] https://en.wikipedia.org/wiki/Enterprise_service_bus