Welcome!

2ND ANNUAL

Ed-Fi Technical Congress
Transition to 3.0 ODS and API, including using the new extensibility model

Ed-Fi Technical Congress 2018
Phoenix, AZ
Agenda

• **ODS/API v3.0 Release Themes** (9:30 am -10:30 am)
  – New & Updated Standards
    • Alignment With the Upcoming Ed-Fi Data Standard v3.0
    • Upgrade to OpenAPI 2.0 Specification – demo
    • Standardized Use of OAuth 2.0 Client Credentials Grant Type
  – New & Improved Features
    • Decoupling of Ed-Fi Core and Extensions in the API - demo
    • Improvements to Routes - demo
    • Support for Multiple Operational Contexts – demo - types & descriptors
  – Code Refactoring
  – Upgrade/Migration support

• **Extending ODS /API v3.0** (10:30 am – 10:45 am)

• **Benefits/Impacts** (10.45 am – 11.15  am)
  – Platform Hosts
  – API Consumers

• **Wrap-up** (11:15 am– 11:30 am)
ODS/API v3.0 – Release Themes
ODS/API v3.0 – Release Themes

• New & Updated Standards
  - Alignment With the Upcoming Ed-Fi Data Standard v3.0
  - Upgrade to OpenAPI 2.0 Specification
  - Standardized Use of OAuth 2.0 Client Credentials Grant Type

• New & Improved Features
  - Decoupling of Ed-Fi Core and Extensions in the API
  - Improvements to Routes
  - Support for Multiple Operational Contexts

• Code Refactoring

• Upgrade/Migration support
New & Updated Standards

Alignment with the

**Data Standard v3.0d - RFC**

- **Data Model Improvements and Enhancements**
- **Continued CEDS Alignment**
  - Alignment of Data Standard v3.0 with CEDS v7.0 is underway. A mapping will be published in Q2 2018
- **Changes to the Ed-Fi Extension Framework**
  - The Ed-Fi Extension Framework provides specific patterns implementers can use to customize the Ed-Fi Data Model while still retaining conformance to the Ed-Fi Data Standards.
  - Some issues with Ed-Fi Data Standard v2.0 guidance
    - Allowed required elements in the Ed-Fi Data Standard to be removed from an implementation.
    - Elements defined as optional could be changed to be required by an implementation.
  - Ed-Fi Data Standard v3.0 [Extensions Framework Guide](#)
Alignment with the Data Standard v3.0d - RFC

Relocation of Evaluation-based Student Attributes to StudentEducationOrganizationAssociation

(DATASTD-1082, DATASTD-1086, DATASTD-1119)
Impacts: Ed-Fi UDM Handbook, Ed-Fi ODS / API, Ed-Fi XSD for Bulk Data Exchange

This location is logical given that those properties are attributes typically assigned by an LEA. This solves that thrashing problem experienced in field.

- LimitedEnglishProficiency and Disability student properties moved to StudentEdOrgAssociation entity.
- EconomicDisadvantaged to become an StudentCharacteristic
- ProgramParticipation moved from Student onto StudentEdOrgAssociation
- StudentIndicator and StudentCharacteristic moved to StudentEdOrgAssociation
Alignment with the Data Standard v3.0d – RFC

Expansion of Support for Standardized Federal EDFacts Data Collection

(DATASTD-1162 + others)
Impacts: Ed-Fi UDM Handbook, Ed-Fi ODS / API, Ed-Fi XSD for Bulk Data Exchange

Data Standard v3.0 Improves the ability of the core model to represent standard EDFacts reporting.

- Four new specialized StudentProgramAssociations were added to the model with standard fields for EDFacts collection:
  - StudentLanguageInstructionProgramAssociation,
  - StudentHomelessProgramAssociation,
  - StudentNeglectedOrDelinquentProgramAssociation
  - StudentSchoolFoodServiceProgramAssociation
- Existing entities for CTE, Title I Part A, Special Education, and Migrant programs were trued-up to the current EDFacts data collection
- New descriptors (enumerations) and common types were added to the model to accompany this change.
Alignment with the Data Standard v3.0d – RFC

Expansion of and Changes to the Calendar Model

(DataSTD-914)
Impacts: Ed-Fi UDM Handbook, Ed-Fi ODS / API, Ed-Fi XSD for Bulk Data Exchange

Data Standard v2.0 only allowed for one calendar per school but in practice, different grades or cohorts of students may have different calendars.

Data Standard v3.0

• Adds the Calendar domain entity with an optional collection of grade levels.
• Adds reference to Calendar on Cohort, Cohorts in a school may also have different calendars independent of grade levels.
• CalendarDates are then no longer directly associated with Schools, but instead with Calendars, which are associated with Schools.
Alignment with the Data Standard v3.0d – RFC

Fixes to the Bell Schedule Model

(DATASTD-948, DATASTD-993, DATASTD-1054)
Impacts: Ed-Fi UDM Handbook, Ed-Fi ODS / API, Ed-Fi XSD for Bulk Data Exchange

Data Standard v2.0

• BellSchedule had a reference (1:1) to CalendarDate causing difficulties in signifying the dates for which a BellSchedule is active.
• GradeLevel was part of the key to BellSchedule causing systems to duplicate bell schedule for each grade level.

Data Standard v3.0

• Removes the reference to the CalendarDate domain entity, replacing it with an optional collection of date elements named "Date".
• Removes GradeLevel from the key to BellSchedule. Since a bell schedule often covers multiple grade levels (frequently a whole school), GradeLevel was changed to an optional array.
Alignment with the Data Standard v3.0d – RFC

Removal of Types

(DATASTD-1088)
Impacts: Ed-Fi UDM Handbook, Ed-Fi ODS / API, Ed-Fi XSD for Bulk Data Exchange

Background:

Changes:
• "Type"-class of entities removed from the data model, and replaced with Ed-Fi Descriptors.
• Mappings from Descriptors to types are removed
• Instead of type sets, standardized enumeration sets provided as part of the Ed-Fi Data Standard will be surfaced as Descriptor sets in the ed-fi.org namespace.
New & Updated Standards

Upgrade to OpenAPI 2.0 Specification
Upgrade to OpenAPI 2.0 Specification

- The ODS/API uses Swagger metadata (now renamed to Open API) to produce visual, interactive documentation & generate Client API SDK

- Ed-Fi ODS / API v2.x produced Swagger 1.2 metadata.
  - This specification is out-of-date, and tooling support is lacking going forward – which means that the SwaggerUI can no longer be updated without an upgrade to the metadata.

- Ed-Fi ODS / API v3.0 produces OpenAPI specification 2.0 metadata bringing in the following benefits
  - Moves to an actively supported and maintained version of SwaggerUI.
  - Adopts Swagger community tools for generating client SDKs for the ODS / API. This provides API SDK support for languages beyond C# and Java, as well as enables the Alliance to deprecate costly support and maintenance for the current Client API SDK.
  - Standardizes the return type for HTTP GETs.
The ODS/API v2.0 implements variable GET semantics to reflect the natural key model of the Ed-Fi standard:

- **“Get By Key”**: Supply natural key fields and return a single JSON object
- **“Get By Example”**: Uses supplied fields as search values and returns a JSON collection

Developers can inadvertently trip over this difference by not catching situations in which a natural key values are submitted (Tracker ODS-894).
New & Updated Standards

Upgrade to OpenAPI 2.0 Specification

Standardized return types for HTTP GETs

- In Open API 2.0 only one operation per HTTP method is permitted on a given REST path.
- This means that **GetByExample** will be the only “GET” operation that will be supported on the collections endpoints and it will always return an array of items.
- Additionally, the **GetAll** operation in the Ed-Fi ODS API 2.x will be just a specialized subset of **GetByExample** that provides no criteria. **GetByKey** will be another specialized subset of **GetByExample** where natural keys are provided as criteria.
- **GetByld, DeleteByld** and **Put** and **Post** operations are unaffected
New & Updated Standards

Upgrade to OpenAPI 2.0 Specification

ODS / API v3.0 metadata endpoints

• **Interactive Documentation use case**
• **SDK gen use case**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>Primary Target Audience</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resources</td>
<td>Includes all the data collection resources including Extensions (but excludes the Descriptors).</td>
<td>API consumers and application vendors using SwaggerUI.</td>
</tr>
<tr>
<td>Descriptors</td>
<td>Includes only the Ed-Fi Descriptors (including Descriptors added via Extensions).</td>
<td>API consumers and application vendors using SwaggerUI.</td>
</tr>
<tr>
<td>Bulk</td>
<td>Contains all operations related to bulk data loads.</td>
<td>API consumers and application vendors who are operating against an API that supports bulk loads.</td>
</tr>
<tr>
<td>Identity</td>
<td>Contains all operations supporting integration with an external Uniqueld system.</td>
<td>API consumers and application vendors working with APIs that have an integration with an external Uniqueld system.</td>
</tr>
<tr>
<td>Profile-Specific</td>
<td>Includes the relevant subset of the Standard API Resources (including extensions and any referenced Descriptors) with appropriately constrained surface areas.</td>
<td>Specialty vendors, for data collection and/or integration.</td>
</tr>
<tr>
<td>Composite-Specific</td>
<td>Includes read-only resources, as defined in the composite, including any referenced descriptors.</td>
<td>API consumers and application vendors focused on data integration activities in a specific subject area.</td>
</tr>
<tr>
<td>Data Collection API</td>
<td>Includes all Descriptors and Resources, including all Extensions.</td>
<td>SDK generators.</td>
</tr>
</tbody>
</table>
OAuth 2.0 Authorization Code Grant Type
(a.k.a. “3-Legged OAuth”)
Authentication (as implemented)  
(a.k.a. “2.5-Legged OAuth”)

• Issues:
  • Authorization code effectively “binds” the client with itself, which is pointless.
  • This is non-standard – It isn't 2-Legged OAuth or 3-Legged OAuth, but something in between.
OAuth 2.0 Client Credentials Grant Type (a.k.a. “2-Legged” OAuth)

- Recommended OAuth flow when the client is the resource owner.

- Token carries a fixed expiration time (**no longer has a rolling expiration**)

- Benefits:
  - A simpler process
  - Implements a standard
  - Works out-of-the-box with tooling (e.g. SwaggerUI and Postman)
Postman OAuth Demo
Decoupling of Ed-Fi Core and Extensions

ODS/ API v2.x

• In ODS/ API v2.x, extensions to the data standard are blended seamlessly into the API resource models and URL paths

• For example, in this JSON snippet it is not obvious to a developer that `tribalAffiliation` is not a core field in the Ed-Fi data model

• One consequence of this design ➔ confusion on the part of API developers as to whether or not an element is part of the Ed-Fi data standard or part of an extension
Decoupling of Ed-Fi Core and Extensions

ODS/ API v2.x (Cont.)

• A 2nd consequence of this design ➔ no way to signal to client developers that an
  implementation is using a standard set of extensions (e.g. talent management or
  transportation domain extensions)
• A 3rd consequence of this design ➔ If an API used more than one set of extensions, name
  collisions from those sets were possible
• The planned enhancement is to separate extensions - both new entities and new fields on
  entities - within the API representations and paths, and introduce namespaces to clarify
  the source
Decoupling of Ed-Fi Core and Extensions

**ODS/ API v3.x**

- Explicit support for extensions in JSON resource models
- Changes to API routes
  
  /data/v3/ed-fi/schools
  /data/v3/talentMgmt/applicants
  /data/v3/TX/leavers
Decoupling of Ed-Fi Core and Extensions Demo
## Improvements to Routes

New Routes support the new extensions model, and provide improved support for API versioning.

<table>
<thead>
<tr>
<th>Description</th>
<th>Current (ODS / API v2.0)</th>
<th>Future (ODS / API v3.0)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base URI</td>
<td>../api/v2.0/2018</td>
<td>/(api_area)/(version)/(year)*</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Examples:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/data/v3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/composites/v1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/identity/v2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/bulk/v1</td>
</tr>
<tr>
<td>Basic Relative URL Format</td>
<td></td>
<td>Data Management Resources:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.../(org)/(resource)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>API Composites:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>.../(org)/(category)/(resource)</td>
</tr>
<tr>
<td>Examples</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Resources (Ed-Fi)</td>
<td>/api/v2.0/2018/schools/</td>
<td>/data/v3/ed-fi/schools</td>
</tr>
<tr>
<td>Descriptors (Ed-Fi)</td>
<td>/api/v2.0/2018/termDescriptors</td>
<td>/data/v3/ed-fi/termDescriptors</td>
</tr>
<tr>
<td>Resources (Host Extensions)</td>
<td>/api/v2.0/2018/leavers</td>
<td>/data/v3/TX/leavers</td>
</tr>
<tr>
<td>Resources (Domain Extensions)</td>
<td>/api/v2.0/2018/applicants</td>
<td>/data/v3/talentMgmt/applicants</td>
</tr>
<tr>
<td>Composites (Host Organization)</td>
<td>Custom</td>
<td>/composites/v1/MI/enrollment/Schools</td>
</tr>
<tr>
<td>Composites Based on Domain or Host Extensions</td>
<td>N/A</td>
<td>Unsupported</td>
</tr>
<tr>
<td>Bulk</td>
<td>api/v2.0/2018/bulkOperations</td>
<td>/bulk/v1/bulkOperations</td>
</tr>
<tr>
<td>Identity</td>
<td>api/identity/v1/identities</td>
<td>/identity/v1/identities</td>
</tr>
<tr>
<td></td>
<td>api/identity/v2/identities</td>
<td>/identity/v2/identities</td>
</tr>
</tbody>
</table>

- **School Year is Optional**
- **Separate versioning of bulk and identity API**
- **Composites is reserved word**
- **Deprecated**
Redesigning the Ed-Fi Enumeration Sets

- Redesigns the Ed-Fi Enumerations to be semantically unambiguous across different implementations which is essential for data integration and interoperability.
- The Alliance envisions that upcoming (ODS/API) release will provide mappings between different Descriptor sets, allowing for **support of multiple "operational contexts"** (contexts in which controlled vocabularies are different for the same underlying concept), and allowing for explicit declaration of operational context. ODS / API v3.0 sets the stage to support multiple operational contexts.
- Ed-Fi Descriptor references require Fully Qualified Namespaces
- Descriptor namespace format - uri://[organization indicator]/[name of descriptor][code value]
  - uri://ed-fi.org/AcademicSubjectDescriptor#Chemistry
  - uri://ed-fi.org/GradeLevelDescriptor#Fifth grade
Ed-Fi Descriptor References Require Fully Qualified Namespaces

Demo
Code Refactoring

• Source code reorganization
  – Address complexity and establish foundation for more manageable, composable releases (e.g. distributing ODS/API core via NuGet packages)

• API code generation template re-engineering and semantic model
  – Address technical debt and prepare for API code generation support in MetaEd

• Note: these changes were also applied to the ODS / API v2.3 release.
Upgrade/Migration support

- Out of the Box core migration support
- Migrating Extensions need custom scripts
- Designed for multi-year ODS implementation. Year-Specific instances will simply switch over to new ODS
Extending ODS /API v3.0

MetaEd IDE 1.1.3
Ed-Fi Extension Framework
How To: Extend the Ed-Fi ODS / API - Student Transportation Example
MetaEd IDE Version Targeting
ODS/API v3.0 Benefits/Impacts
Platform Hosts

- Engage your Vendors early on. Apply and test Extensions, deploy sandboxes and coordinate roll-out Review new Extension capabilities
- Review down stream applications that may require changes
- Switch to MetaEd and verify that your extensions confirm to guidelines provided by Ed-Fi Extension Framework
- Review current Extensions to see if any elements are included in the new Data Standard.
Platform Hosts

• Review new extension capabilities. It supports an Ed-Fi Core plus composable Ed-Fi Extensions
• Platform hosts can apply multiple extensions, such as host-specific extensions (e.g., an SEA-specific accountability collection), along with domain-specific extensions (e.g., talent management, transportation).
Platform Hosts

- Review all Ed-Fi Descriptors, ensure any localized descriptors (i.e., those with values other than the as-shipped Ed-Fi Descriptor sets) retain the namespace of the defining organization.
- Make sure localized descriptors adhere to the fully qualified namespace pattern.
- Publish updated documentation, notify client systems.
Platform Hosts

• Review and consume the refactored code.
• If applicable, re-apply customizations.
ODS/API v3.0 Benefits/Impacts

API Consumers

• Review the client application for data standard v3.0 changes to breaking JSON payloads
API Consumers

• Review new Extension capabilities.
• Review and Update the client applications to accommodate changes to extension JSON payload.
• Review and Update the client applications to accommodate changes to Routes
API Consumers

• If applicable (e.g., if developers have used the customized Ed-Fi SDKs for C# or Java), migrate to the latest standard Swagger Codegen client SDK tooling compatible with OpenAPI 2.0.

• If applicable (e.g., if developers have used the GetByKey or GetByExample), modify code to adapt to standardized return types for GETs.
API Consumers

- Adapt client code to OAuth Client Credential Flow.
- OAuth Client Credential Flow doesn’t allow for rolling expiration for tokens. Adapt client code to request a new access token upon token expiration.
API Consumers

• All descriptor reference require namespaces now. Adapt client code to the new specifications.
Wrap-up
Prepare for the switch

- Download and work with the release candidate
- Provide feedback - RFC
- Upgrading to Ed-Fi 3.0 for 2019-2020 School Year?
  - Working Session with Sayee Srinivasan at 2:15pm
Resources

- **Data Standard 3.0(RC1)**
- **UDM Handbook**
- **Swagger Codegen Tool**
- **Hands-on exercise on how to use Swagger Codegen tool**
- **Webinars** - “What’s New With Ed-Fi 3.0: Part 1” and “What’s New With Ed-Fi 3.0: Part 2”
- **MetaEd Labs** – From 2017 Bootcamp