

SIS Vendor and District Test Plan for School Year 2021-2022 (Ed-Fi 3.1)

Minnesota Department of Education & Minnesota IT Services Automated Student Data Collection System

Revision date: 11/25/2020

Table of Contents

[Overview 1](#_Toc57886608)

[MN Sandbox Certification Testing 1](#_Toc57886609)

[Test Sandbox Environment URLs: 2](#_Toc57886610)

[Swagger 2](#_Toc57886611)

[Ed-Fi Sandbox Admin Website 2](#_Toc57886612)

[ODS/API base URL 3](#_Toc57886613)

[URL changes for addressing extension vs core resources 3](#_Toc57886614)

[ODS/API oauth URL 3](#_Toc57886615)

[Sandbox Environment Helpful Tips 3](#_Toc57886616)

[How LocalEducationAgencyId and SchoolId are derived from MDE stateOrganizationID 4](#_Toc57886617)

[2021-2022 Data Requirements - API Resources 5](#_Toc57886618)

[API Documentation 5](#_Toc57886619)

[Mapping Documentation 8](#_Toc57886620)

[Ed-FI Collection Updates Summary for School Year 21-22 9](#_Toc57886621)

[Education Organization Id usage by Resource 9](#_Toc57886622)

[MARSS and Ancestry of Ethnic Origin Certification Scenarios – Ed-Fi API Resources 10](#_Toc57886623)

[Resource: Students 10](#_Toc57886624)

[Resource: StudentSchoolAssociations 10](#_Toc57886625)

[Resource: StudentEducationOrganizationAssociation 11](#_Toc57886626)

[Resource: Calendar 11](#_Toc57886627)

[Resource: CalendarDate 12](#_Toc57886628)

[Resource: ClassPeriod 13](#_Toc57886629)

[Resource: GradingPeriod 13](#_Toc57886630)

[Resource: Session 14](#_Toc57886631)

[Resource: Course 14](#_Toc57886632)

[Resource: CourseOffering 17](#_Toc57886633)

[Resource: Section 19](#_Toc57886634)

[Resource: StaffSectionAssociation 19](#_Toc57886635)

[Resource: StudentSectionAssociation 20](#_Toc57886636)

[Resource: Grade 21](#_Toc57886637)

[Resource: Programs 21](#_Toc57886638)

[StudentProgramAssociations 22](#_Toc57886639)

[Resource: Student21stCenturyLearningCenterGrantProgramAssociation 22](#_Toc57886640)

[Resource: StudentCEISProgramAssociation 22](#_Toc57886641)

[Resource: StudentEarlyChildhoodScreeningProgramAssociations 22](#_Toc57886642)

[Resource: StudentGiftedTalentedProgramAssociation 22](#_Toc57886643)

[Resource: StudentADSISProgramAssociation 23](#_Toc57886644)

[Resource: StudentHomelessProgramAssociation 23](#_Toc57886645)

[Resource: StudentLanguageInstructionProgramAssociation 23](#_Toc57886646)

[Resource: StudentPSEOConcurrentProgramAssociation 24](#_Toc57886647)

[Resource: StudentPSEOProgramAssociation 24](#_Toc57886648)

[Resource: StudentSAAPProgramAssociation 24](#_Toc57886649)

[Resource: StudentSchoolFoodServiceProgramAssociation 25](#_Toc57886650)

[Resource: StudentSection504PlanProgramAssociation 25](#_Toc57886651)

[Resource: StudentSpecialEducationProgramAssociation 25](#_Toc57886652)

[Resource: StudentTitleIPartAProgramAssociations 26](#_Toc57886653)

[GRR Certification Scenarios - API Resources 26](#_Toc57886654)

[Resource: ExamReimbursementRequest 26](#_Toc57886655)

[MCCC Certification Scenarios - API Resources 27](#_Toc57886656)

[Resource: Students 27](#_Toc57886657)

[Resource: StudentSchoolAssociations 27](#_Toc57886658)

[Resource: StudentEducationOrganizationAssociation 27](#_Toc57886659)

[Resource: Calendar 28](#_Toc57886660)

[Resource: CalendarDate 28](#_Toc57886661)

[Resource: ClassPeriod 28](#_Toc57886662)

[Resource: GradingPeriod 29](#_Toc57886663)

[Resource: Session 29](#_Toc57886664)

[Resource: Course 30](#_Toc57886665)

[Resource: CourseOffering 32](#_Toc57886666)

[Resource: Section 34](#_Toc57886667)

[Resource: StaffSectionAssociation 35](#_Toc57886668)

[Resource: StudentSectionAssociation 35](#_Toc57886669)

[Resource: Grade 36](#_Toc57886670)

[Early Education Enrollment Certification Scenarios - API Resources 36](#_Toc57886671)

[Resource: StudentSchoolAssociations 36](#_Toc57886672)

[Resource: Parent 37](#_Toc57886673)

[Resource: StudentParentAssociation 38](#_Toc57886674)

[Resource: StudentEducationOrganizationAssociation 38](#_Toc57886675)

[Resource: StudentEarlyEducationProgramAssociations 38](#_Toc57886676)

[Resource: StudentHomelessProgramAssociation 39](#_Toc57886677)

[Resource: StudentLanguageInstructionProgramAssociation 40](#_Toc57886678)

[2021-2022 MDE Submitted Data Requirements - API Resources 40](#_Toc57886679)

[Resource: LocalEducationAgencies 40](#_Toc57886680)

[Resource: Schools 40](#_Toc57886681)

[Resource: Post-Secondary Institutions 40](#_Toc57886682)

[Resource: Course (State Level Only) 41](#_Toc57886683)

[Resource: Staff 41](#_Toc57886684)

[Resource: Program 41](#_Toc57886685)

[Ed-Fi Model Dependency Graph (MARSS collection) 42](#_Toc57886686)

[Ed-Fi Model Dependency Graph (MCCC collection) 43](#_Toc57886687)

[Staging Environment Load and Quality Check 44](#_Toc57886688)

[Ed-Fi / MARSS Identities API Integration Test 44](#_Toc57886689)

[Create Student ID Test 1 44](#_Toc57886690)

[Create Student ID Test 2 45](#_Toc57886691)

[Ed-Fi / MARSS Student Record Validation Test 46](#_Toc57886692)

[Student Validation Process Overview 46](#_Toc57886693)

[Validate Student ID Test 1 46](#_Toc57886694)

[Validate Student ID Test 2 46](#_Toc57886695)

**Change Log:**

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Description of Changes** |
| v1.0 | 11/25/20 | Initial Version for SY21-22 Collection |
| v1.1 | 12/03/20 | Minor edits and clarifications |

# Overview

Student Information Systems vendors integrating with the Minnesota Department of Education's (MDE) Ed-Fi Automated Student Data Collection system are required to pass 2 levels of testing prior to being granted a district's API key and secret to the production environment.

As in School year 20-21, the 21-22 School Year Certification will also be against the Ed-Fi 3.1 Data Standard and ODS/API with extensions customized for MDE.

First, vendors are required to demonstrate their ability to submit the specific API resources and elements encompassed within the MARSS Data elements. This first phase is executed in the MNIT Ed-Fi Test Sandbox environment.

**The Test Sandbox environment does not include integration with the Identities API endpoint.**

**API profiles are not enforced in the Test Sandbox environment, however you may be required to demonstrate appropriate implementation of API profile use in your API transactions.**

When all scenarios are verified in the Sandbox testing environment, the SIS vendor is granted access to the Staging Environment. Vendors must perform a test load of their district's real student demographic data in the Staging environment. During this test, Student IDs will be validated against the Ed-Fi Identities Service to ensure valid ids are being inserted to the Ed-Fi ODS. MARSS comparison checks and additional data quality verifications are executed in the Staging Environment.

Once the Staging data reaches a satisfactory level of quality as determined by MDE, MNIT, and the District, MNIT will enable the vendor to obtain a key and secret for the production environment.

This document includes the Certification Requirements for both the scenario-based sandbox testing and staging environment data loading.

*Note: many URLs in this document are not yet active, and thus marked with:*

Not currently active

*Please note that these URLs are also subject to change upon actual Ed-Fi implementation for the 2021-2022 school year.*

# MN Sandbox Certification Testing

When certification begins, vendors will be provided a key and secret to a sandbox ODS/API with no sample data pre-loaded. This ODS will contain the Minnesota Education School and District records synchronized with MDE-ORG, as well as the MN specific descriptors listed below in this document. When submitting data for each of the Certification scenarios, we ask that you use a district you have access to, and schools within the district.

## Test Sandbox Environment URLs:

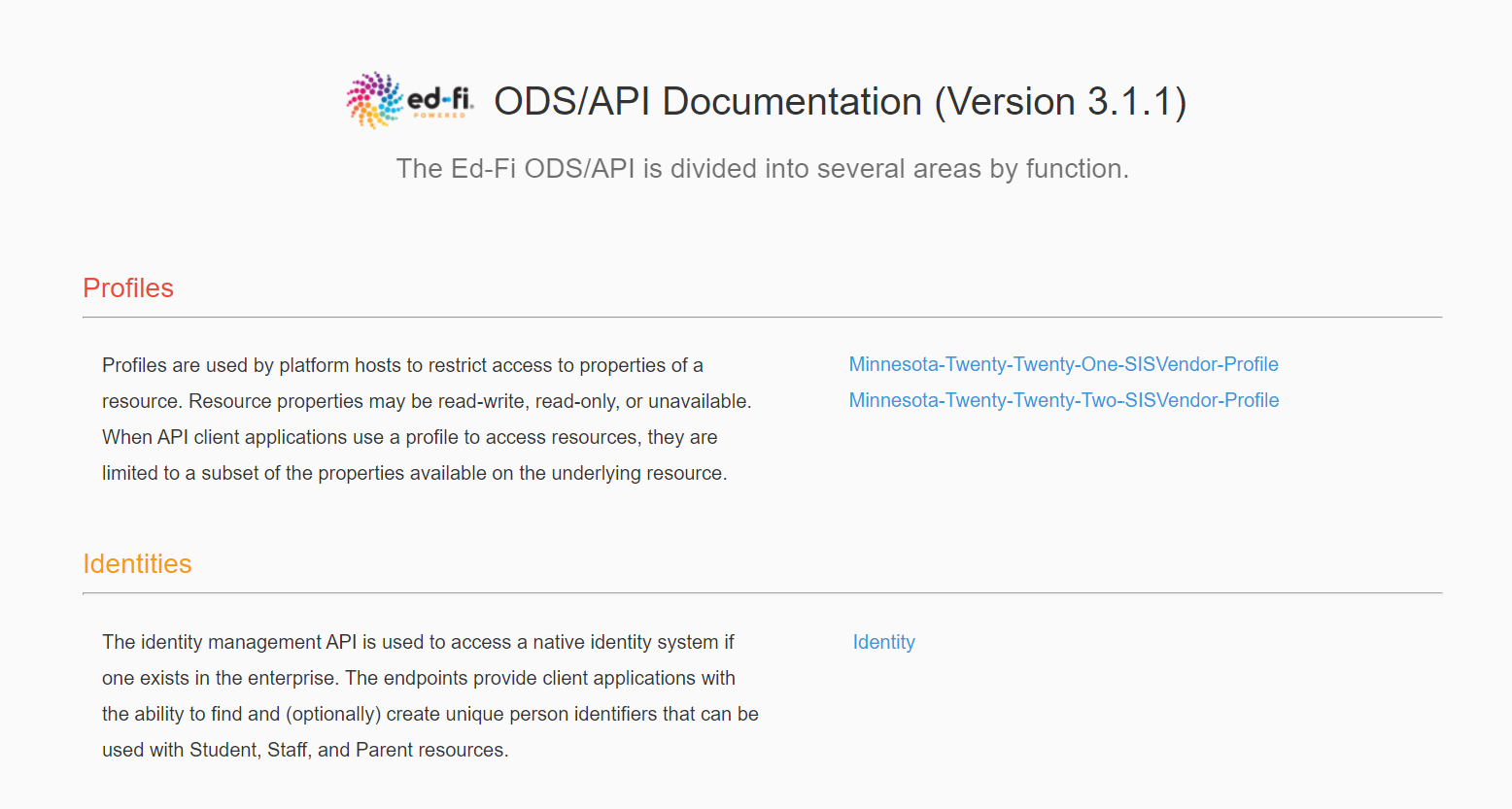
### Swagger

<https://test.edfi.education.mn.gov/sb21_/EdFi.Ods.SwaggerUI/>  
Not currently active

MDE’s Swagger website has been customized to remove the generic Resources section. All MN Specific resources can be found under the Profiles for each School Year:

* Minnesota-Twenty-Twenty-One-SISVendor-Profile – For Data submitted to the School Year 2020-2021 ODS
* Minnesota-Twenty-Twenty-Two-SISVendor-Profile – For Data submitted to the School Year 2021-2022 ODS

*Note: The Identities Section is provided as documentation only – and is not functional in the Sandbox environment*



### Ed-Fi Sandbox Admin Website

<https://test.edfi.education.mn.gov/sb21_/EdFi.Ods.Admin.Web>   
Not currently active

To request an account, please contact MDE.

In the Sandbox Admin Tool, you will create test sandbox ODS instances to use for development and testing of your integrations.

### ODS/API base URL

<https://test.edfi.education.mn.gov/sb21_/edfi.ods.webapi/data/v3/>  
Not currently active

#### URL changes for addressing extension vs core resources

##### Extensions

The URL for addressing MN extension resources should include ‘mn’ after v3, for example:

When addressing studentSection504PlanProgramAssociations, a Minnesota Dept of Education extension entity, the URL is:

…/edfi.ods.webapi/data/v3/**MN**/studentSection504PlanProgramAssociations

##### Core

The URL for addressing core resources should include ‘ed-fi’ after v3, for example:

When addressing StudentSchoolAssociation, a core ed-fi entity, the URL is:

…/edfi.ods.webapi/data/v3/**ed-fi**/StudentSchoolAssociation

**Important Note**: the school year must be included after “/v3/” and before the core/extension namespace in QA, Stage and Production, **but not in Sandbox.**

For example:

…/edfi.ods.webapi/data/v3/**2022**/ed-fi/StudentSchoolAssociation

…/edfi.ods.webapi/data/v3/**2022**/MN/studentSection504PlanProgramAssociations

### ODS/API oauth URL

<https://test.edfi.education.mn.gov/sb21_/edfi.ods.webapi/oauth/>  
Not currently active

Authentication in Ed-Fi 3.x ODS/API uses two-legged OAuth 2.0 Client Credentials Grant Flow. More information and sample API calls are located here:

<https://techdocs.ed-fi.org/display/ODSAPI32/Authentication>

## Sandbox Environment Helpful Tips

* In the staging and production environments, SIS vendor API keys will be associated with a claim set limiting access to only the API resources included in this document. This claim set has been enabled in the Sandbox environment. When you access a resource not included in the claim set, you will receive the following message in the response "Access to the resource could not be authorized. Are you missing a claim?”
* A SIS Vendor profile has been created limiting access to specific properties on each of the resources required in each School Year. Profiles are tied to your key and secret and auto-enabled in Staging and Production – meaning you will not have to include the profile in the “accept” or “content-type” headers of the request for the profile to take effect.
* When creating a sandbox, you have the option of including a set of sample data in the sandbox. The sample data used, is associated with St. Paul Public School District (LocalEducationAgencyId = 10625000). When you do not select the option to include sample data, your sandbox database will be loaded with an essentially empty ODS except that it will include MN Schools and Districts, and the Descriptors that MDE has customized for this implementation.
* The actual certification test will be conducted in a sandbox with no sample data loaded and you will reference your own MDE Schools, Districts and Descriptors in the Scenarios.
* The Sandbox Environment ignores the School Year included in the API base URL. Since you will be submitting data to the 2021-2022 School year, to prep for submissions to Staging and Production, remember to include the year, i.e. “2022”.
* MN custom descriptor codes have been loaded under the namespace "uri://education.mn.gov/”
* All Descriptor references require namespaces, and do not rely on the concept of a default operational context. Descriptor references must be formatted as follows: uri://[organization indicator]/[name of descriptor]#[code value]
  + For example: **uri://education.mn.gov/ProgramTypeDescriptor#Special Education**
* The School Ids and District Ids used for the ODS/API resources are the MN State Education Organization IDs.
* StaffUniqueId will be loaded using the FFN in the Educator Licensing System.
* Post-Secondary Institution ID will be pre-loaded by MDE – these college identifiers will come from the MDE-ORG system.

## How LocalEducationAgencyId and SchoolId are derived from MDE stateOrganizationID

The MDE stateOrganizationID is formatted as follows: ttddddsssmmm

Where:

* tt = district type
* dddd = district number, left zero filled
* sss = school number, left zero filled, 000 for districts
* mmm = 000 for all organizations reported in Ed-Fi

The MDE stateOrganizationID value is stored in Ed-Fi on the Ed-Fi **EducationOrganizationIdentificationCodes** collection and is surfaced via the Ed-Fi **LocalEducationAgencies** and **Schools** Resource endpoints.

The Ed-Fi LocalEducationAgencyId and SchoolId are derived as follows ttddddsss. ‘sss’ is Zero filled for the LocalEducationAgencyId.

For Example:

* MN State Organization ID for a school is 10535315000 – Ed-Fi SchoolId = 10535315
* MN State Organization ID for a district is 10535000000 – Ed-Fi LocalEducationAgencyId = 10535000

# 2021-2022 Data Requirements - API Resources

## API Documentation

For each of the following resources, the elements/properties required are included in Swagger under the current Profile:

<https://test.edfi.education.mn.gov/sb21_/EdFi.Ods.SwaggerUI/>   
Not currently active

For example, to view required resource properties for StudentSchoolAssociation, open the post action in Swagger:

example of post action in Swagger

Note: ed-fi in the path above indicates that this is a core resource.

Properties in the StudentSchoolAssociation can be viewed as the JSON object sample by selecting “Example Value”:



Definitions and Data Types in the StudentSchoolAssociation can be viewed by selecting “Model”:



## Mapping Documentation

School Year 21-22 will introduce the Ed-Fi collection of Minnesota Common Course Catalogue (MCCC) and Graduation Requirements (GRR).

In addition, MDE is beginning to work toward the eventual collection of Assessment pre-code and Assessment results data for the MCA and MTAS Assessment.

In School Year 19-20, MDE implemented an Ed-Fi based MARSS collection.

School Year 20-21 included the introduction of the Early Education data collection.

For context around how each MDE collection maps to the Ed-Fi Data Standard please view the following spreadsheets:

[2020-2021 Mapping Review Matrix Ed-Fi 3.1.xls spreadsheet](https://github.com/mn-mde-edfi/MDE-EdFi-Documentation/blob/master/2020-21%20MDE%20Ed-Fi%20Documentation/2020-21%20Student%20Enrollment%20Data%20Mapping%20Matrix%20Ed-Fi%203.1.xlsx) (MARSS & EE)

MCCC Mapping Matrix (TBD)

GRR Mapping Matrix (TBD)

The Mapping Review Matrix documents include the mappings between MDE elements and Ed-Fi core and extension entities and elements.

Additionally, the Descriptor values applicable to each Collection are also included in these documents for reference.



**Data Element Mappings** **Descriptors**

## Ed-FI Collection Updates Summary for School Year 21-22

* The Minnesota Common Course Catalogue data collection has been implemented in Ed-Fi. Most MCCC elements are collected in the following core Ed-Fi Entities:
  + Course, CourseOffering, Section, StudentSectionAssociation, StaffSectionAssignment, ClassPeriod, Calendar and Grade.
  + An extension CourseCourseAssociation has been added to allow associations between District, State and College Level Courses.
* The ExamReimbursementRequest has been added to Collect GRR requests.

## Education Organization Id usage by Resource

Education Organization References in the Ed-Fi API allow an API client to submit either a Local Education Agency, School Id, PostSecondaryInstitutionId; however the MDE implementation requires that the following ids must be used for Education Organization References:

| **Resource** | **Element** | **ID** |
| --- | --- | --- |
| StudentEducationOrganizationAssociation | educationOrganizationReference.educationOrganizationId | School Id |
| StudentProgramAssociation (all program types) | educationOrganizationReference. educationOrganizationId | School Id |
| StudentProgramAssociation (all program types) | programReference.educationOrganizationId | State Education Agency Id |
| Course | educationOrganizationReference.educationOrganizationId | District Course –**LocalEducationAgencyId**  State Course (loaded by MDE) – **StateEducationAgencyId**  College Course –**postSecondaryInstitutionId** |
| CourseOffering | SchoolReference  CourseReference | SchoolId  EducationOrganizationId on the Course record |
| ClassPeriod | SchoolReference | SchoolId |
| Section | CourseOfferingReference  ClassPeriodReference | SchoolId  SchoolId |
| studentSectionAssociation | sectionReference>CourseOfferingReference | SchoolId |
| StaffSectionAssociation | sectionReference>CourseOfferingReference | SchoolId |
| Grade | StudentSectionAssociationReference> sectionReference>CourseOfferingReference | SchoolId |
| GradingPeriod | SchoolReference | SchoolId |

## MARSS and Ancestry of Ethnic Origin Certification Scenarios – Ed-Fi API Resources

### Resource: Students

**Description**

This entity represents an individual for whom instruction, services, and/or care are provided in an early childhood, elementary, or secondary educational program under the jurisdiction of a school, education agency or other institution or program. A student is a person who has been enrolled in a school or other educational institution.

**Prerequisite Data**

* None

**Certification Scenarios**

1. Create 14 student records including the following elements.
   1. studentUniqueID
   2. birthdate
   3. birthSexDescriptor
   4. firstName
   5. generationCodeSuffix
   6. lastSurname
   7. middleName

### Resource: StudentSchoolAssociations

**Description**

This association represents the School in which a student is enrolled. The semantics of enrollment may differ slightly by state. Non-enrollment relationships between a student and an education organization may be described using the StudentEducationOrganizationAssociation.

**Prerequisite Data**

* Students
* Schools

**Certification Scenarios**

1. Enroll student 1 at Elementary School, including
   1. homeboundServiceIndicator
   2. specialPupilIndicator
2. Enroll student 2 at Middle School.
3. Withdraw student from Middle School.
4. Enroll student 2 at High School.
5. Create the following records for Student 2
   1. An enrollment
   2. Withdrawal
   3. Re-enrollment of a student who changes resident districts but does not change schools. Place a 20 day gap between records.
6. Update Student 1’s Percent Enrolled to 50% and set membershipAttendanceUnitDescriptor to days, set membership and attendance.
7. Update Student **1’s Percent Enrolled to 100%** membershipAttendanceUnitDescriptor to hours and adjust membership and attendance accordingly.
8. Update Student 2’s Transportation Category.
9. Update Student 3’s StateAid Category.
10. Create 2 enrollment records (a, b) for a student that is in two different schools within the same district at the same time. Enrollment dates 09/10/17 through 06/10/18.
11. Create the records for
    1. An enrollment,
    2. Withdrawal and
    3. Re-enrollment for a kindergarten student who receives an IEP mid-year – send updated special education evaluation status. (Grade change from KA to HK, and specialEducationEvaluationStatus updated)

### Resource: StudentEducationOrganizationAssociation

**Description**

This association indicates any relationship between a student and an education organization other than how the state views enrollment. Enrollment relationship semantics are covered by StudentSchoolAssociation.

MDE allows for the capture of student demographic data by school enrollment. Therefore, a StudentEducationOrganizationAssociation record must be submitted for each **school** in which the student is enrolled to provide the student demographic data provided to the enrolling school by the parent(s).

**Prerequisite Data**

* Students
* EducationOrganizations (Schools)

**Scenarios**

1. Create a StudentEducationOrganizationAssociation between Student 1 and an Elementary School, Include the following data points:
   1. EthnicCode by sending a StudentCharacteristic = ‘American Indian - Alaskan Native (Minnesota)’
   2. Race = ‘American Indian - Alaska Native’
   3. Birthdate
   4. sexDescriptor
   5. firstName, middleName, lastName, generationCodeSuffix
   6. hispanicLatinoEthnicity
   7. languageDescriptor
   8. languageUseDescriptor = ‘Home Language’
   9. studentIdentificationCodes
   10. studentIdentificationSystemDescriptor = ‘Local’
   11. assigningOrganizationIdentificationCode = ‘District Id’
   12. identificationCode = local use code
2. Update Student 1’s record to include Ancestry of Ethnic Origin = ai-cherokee
3. Update Student 1’s record to include a second Ancestry of Ethnic Origin = as-burmese
4. Create a StudentEducationOrganizationAssociation between Student 2 and Middle School.
5. Create a StudentEducationOrganizationAssociation between Student 2 and High School.
6. Update Student 3’s OptOutIndicator

### Resource: Calendar

**Description**

Ed-Fi Description: A set of dates associated with an organization. MDE is not using the Calendar entity as collection of dates, rather MDE captures the following key pieces of Calendar Metadata in the Calendar file: Instructional Days, Length of Day, and Kindergarten Schedule (when applicable). Calendar is captured at the School Level by grade.

**Prerequisite Data**

* Schools (published to ODS by MDE)

**Changes from 19-20 to 20-21**

New properties:

* DaysInSession
* Description
* New CalendarTypeDescriptor for MCCC

**Scenarios**

**MARSS**

1. Create calendar record for Elementary School grades 1 through 5 (do not include kindergarten as grade level)
   1. Length of day = 360
   2. Instructional Days = 167
2. Update calendar created in scenario one by reducing the Instructional Days from 167 to 166
3. Create calendar record for Elementary School for Kindergarten only - use grade KG
   1. Length of day = 360
   2. Instructional Days = 165
4. Create Full Year School Readiness Plus calendar for Elementary School grade RA
   1. Length of day = 150
   2. Instructional Days = 148
5. Create Half Year School Readiness Plus calendar for Elementary School grade RB
   1. Length of day = 150
   2. Instructional Days = 78
6. Create Middle School Calendar
7. Create High School Calendar

**MCCC**

1. Create MCCC Calendar including
   1. Calendar Code
   2. School ID
   3. School Year
   4. CalendarType
   5. Days In Session
   6. Instructional Days
   7. Description

### Resource: CalendarDate

**Description**

Ed-Fi Description: The type of scheduled or unscheduled event for the day. MDE will be collecting the MCCC ScheduleInfo Start and End Dates using Calendar Date

**Prerequisite Data**

* Schools (published to ODS by MDE)
* MCCC Calendar

**Scenarios**

**MCCC**

1. Create 2 Calendar Dates
   1. Date 1 – calendarEventDescriptor = STARTDT
   2. Date 2 – calendarEventDescriptor = ENDDT
   3. Both Dates should reference the MCCC Calendar created above

### Resource: ClassPeriod

**Description**

Ed-Fi Description: This entity represents the designation of a regularly scheduled series of class meetings at designated times and days of the week. MDE will be collecting the MCCC PeriodInfo Data in this resource. This data must be submitted for each school.

**Prerequisite Data**

* Schools (published to ODS by MDE)

**Scenarios**

**MCCC**

1. Create a series of Class Periods for a High School
   1. Each Period must include:
      1. ClassPeriodName
      2. ClassPeriodDescriptor
      3. SchoolID
      4. ClassPeriodDescription
      5. StartTime
      6. EndTime

### Resource: GradingPeriod

**Description**

Ed-Fi Description: This entity represents the time span for which grades are reported. MCCC collection in Ed-Fi requires the creation of GradingPeriods in order to allow submission of grades. Grading Periods are defined at the School Level.

**Prerequisite Data**

* Schools (published to ODS by MDE)

**Scenarios**

**MCCC**

1. Create a series of GradingPeriods for a High School
   1. Each Period must include:
      1. GradingPeriodDescriptor
      2. PeriodSequence
      3. SchoolId
      4. SchoolYear

### Resource: Session

**Description**

Ed-Fi Description: A term in the school year, generally a unit of time into which courses are scheduled, instruction occurs and by which credits are awarded. Sessions may be interrupted by vacations or other events. The MCCC collection in Ed-Fi will use Session for collection of TermInfo.

**Prerequisite Data**

* Schools (published to ODS by MDE)

**Scenarios**

**MCCC**

1. Create Sessions for a High School (Fall, Spring, and non-scheduled)
   1. Each Period must include:
      1. TermDescriptor (S for Semester, NS for non-scheduled)
      2. School Year
      3. SchoolId
      4. DaysInSession
      5. TotalInstructionalDays
      6. SessionName
      7. BeginDate
      8. EndDate
      9. TermNumber

### Resource: Course

**Description**

Ed-Fi Description: This educational entity represents the organization of subject matter and related learning experiences provided for the instruction of students on a regular or systematic basis. The MCCC collection in Ed-Fi will use Course for collection of District, College and State Courses.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* State Courses (published to ODS by MDE)

**Scenarios**

**MCCC**

1. Create 8 **District** Courses
   1. Course 1 Course Level Type = B
   2. Course 2 Course Level Type = G
   3. Course 3 Course Level Type = E
   4. Course 4 Course Level Type = D
   5. Course 5 Course Level Type = A
   6. Course 6 Course Level Type = C
   7. Course 7 Course Level Type = X
   8. Course 8 Course Level Type = N

Including the following elements:

* + 1. CourseDescription (only when associated with an Unclassified State Course)
    2. HighSchoolCourseRequirement
    3. CourseLevelCharacteristic for End of Course Indicator
    4. Course Code (Local)
    5. SequenceLimit
    6. LocalEducationAgencyId
    7. CourseTitle
    8. CourseDefinedByDescriptorId = “LEA”
    9. CourseIdentificationCode – repeat the CourseCode (this is an ed-fi requirement)
    10. CourseIdentificationSystemDescriptor = ‘LEA course code’
    11. *LearningStandardid* (not currently required)

1. Create an EE District Course   
   Elements that must have valid values:
   1. Course Description (only required when associated with Unclassified State Course)
   2. Course Code (local)
   3. Course level Type (P)
   4. Sequence Limit (1)
   5. StandardId (State)
   6. High School Course Requirement (False)
   7. **Must NOT include CourseLevelCharacteristic for End of Course Indicator**
   8. CurriculumUsed
      1. CurriculumUsedDescriptor (CRCUE, HWOT)
      2. ImplementationStatusDescriptor (FULL, CEXP)
   9. AssessmentTool
      1. AssessmentToolDescriptor
      2. ImplementationStatusDescriptor
   10. CourseDefinedByDescriptorId = ‘LEA”
       1. CourseIdentificationCode – repeat the CourseCode (this is an ed-fi requirement)
       2. CourseIdentificationSystemDescriptor = ‘LEA course code’
   11. LearningStandardid (99.E5.1)
2. District Courses to associate with College courses
   1. Create additional District courses to be associated with college course levels D and A
3. Create **College** Courses
   1. 1 with Course Level Type = D
   2. 1 with Course Level Type = A
   3. 2 college courses to be used for Direct Pay PSEO, including the following elements:
      1. PostSecondaryInstitutionId
      2. MaximumAvailableCredits
      3. CourseCode (College Course Identifier – generally dept letters & course number)
      4. CourseTitle
      5. CourseDefinedByDescriptorId = ‘College”
      6. CourseIdentificationCode – repeat the CourseCode (this is an ed-fi requirement)
      7. CourseIdentificationSystemDescriptor = ‘University course code’
4. Create a District Course for Independent Study

**NOTE:** Independent Study course section enrollments did not previously require the setup of a course section – the Ed-Fi Model enforces the full set of master schedule entities (course, course offering and section) in order to associate a student with a course and record grades for that course.

Separate course, course offering and section records are required for your district to report ALC Independent Study records; however, Course Offering and Section will not be used by MDE for reporting. (The Ed-Fi Model requires an entry in Course Offering linked to the non-scheduled term, and a section default.)

* 1. Include the following elements:
     1. CourseLevelCharacteristic = IS
     2. Course Code
     3. CourseDescription
     4. HighSchoolCourseRequirement
     5. SequenceLimit
     6. LocalEducationAgencyId
     7. CourseTitle
     8. CourseDefinedByDescriptorId = ‘LEA”
     9. CourseIdentificationCode – repeat the CourseCode (this is an ed-fi requirement)
     10. CourseIdentificationSystemDescriptor = ‘LEA course code’
     11. LearningStandardid (k-12 Courses)

1. **Create a course** for Direct Pay PSEO section enrollments – the Ed-Fi Model requires course, course offering and section. At minimum, a **single Placeholder course, course offering, and section will be required for your district** to report Direct Pay PSEO Student Section Association records.
   1. Ed-Fi required elements:
      1. Course Code: Direct Pay PSEO
      2. Course Title
      3. Course Identification Code
      4. Number of Parts
      5. CourseDefinedByDescriptor = “LEA’
2. **Not applicable to SIS Vendors - Project Foundry Only -** Create a course for Project Based section enrollments. While these types of enrollments do not include MDE course, course offering or section requirements, the Ed-Fi Model enforces these entities in the Master Schedule. At minimum, a **single Placeholder course, course offering, and section will be required for your district** to report Project Based Student Section Association records.
   1. Ed-Fi required elements:
      1. Course Code: Project Based
      2. CourseLevelCharacteristic = PBL
      3. Course Title
      4. Course Identification Code
      5. Number of Parts
      6. CourseDefinedByDescriptor = “LEA’
3. Create CourseCourseAssociation records between the district Courses and State Level Courses (including IS, and Project Based if applicable)
   1. EducationOrganizationId = District ID
   2. CourseCode = District Course Code
   3. ToCourseEducationOrganizationId = State Course EducationOrganizationId
   4. ToCourseCode = State Course Code
      1. IS can only use state codes with CourseLevelCharacteristicDescription = IS and Project Based can only use state codes with CourseLevelCharacteristicDescription = PBL.
      2. Scheduled course work type cannot use CourseLevelCharacteristicDescription = PBL.
4. Create CourseCourseAssociation records between the College Courses A and D and the District Courses created for this association.
   1. EducationOrganizationId = District ID
   2. CourseCode = District Course Code
   3. ToCourseEducationOrganizationId = College Course’s PostSecondaryInstitutionId
   4. ToCourseCode = College Course Code
5. Create CourseCourseAssociation records between the Direct Pay College Courses and the applicable State Course (PSEO Direct Pay)
   1. EducationOrganizationId = College Education Organization ID
   2. CourseCode = College Course Code
   3. ToCourseEducationOrganizationId = State SEA id
   4. ToCourseCode = State Course Code

### Resource: CourseOffering

**Description**

Ed-Fi Description: This entity represents an entry in the course catalog of available courses offered by the school during a session. The MCCC collection in Ed-Fi will use Course Offering for collection of LocalCourseInformation that is not captured within the Ed-Fi Course record. The CourseOffering also links up the Course to the term and section, student and staff allowing grades and staff assignments to be captured.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* State, College and District Courses
* Sessions

**Scenarios**

**MCCC**

Create the following CourseOffering Records

1. CourseOffering 1 References the District Course with Level Type = B
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
2. CourseOffering 2 References the District Course with Course Level Type = G
   * 1. CourseReference
     2. LocalCourseCode (this can match the District Course’s Course Code – at the discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
3. CourseOffering 3 References the District Course with Course Level Type = E
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
4. CourseOffering 4 References the District Course with Course Level Type = D
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
5. CourseOffering 5 References the District Course with Course Level Type = A
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
6. CourseOffering 6 References the District Course Associated with Course Level Type = C CourseReference
   * 1. LocalCourseCode (this can match the District Course’s Course Code – at the discretion of district)
     2. SchoolId
     3. SessionReference (SchoolYear, SchoolId, SessionName)
     4. InstructionMinutesPerTerm
7. CourseOffering 7 References the District Course with Course Level Type = X
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
8. CourseOffering 8 References the District Course with Course Level Type = N
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
9. CourseOffering 9 References the District Course Associated with Course Level Type = P
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
     6. InstructionalApproach
        1. InstructionalApproachDescriptor
        2. ImplementationStatusDescriptor
     7. siteBasedInitiative
        1. siteBasedInitiativeDescriptor
        2. ImplementationStatusDescriptor
10. CourseOffering 10 for Independent Study **(you will have one course offering for every Independent Study district course)**
    * 1. CourseReference (reference to district course)
      2. LocalCourseCode (District Course’s Code - at discretion of district)
      3. Schoolid
      4. SessionReference (SchoolYear, SchoolId, SessionName = “Unscheduled”)
11. CourseOffering 11 for Direct Pay PSEO **(you will have a single course offering linked to a single district course set up as a placeholder)**
    * 1. CourseReference (reference to district course)
      2. LocalCourseCode (District Course’s Code - at discretion of district)
      3. Schoolid
      4. SessionReference (SchoolYear, SchoolId, SessionName = “Unscheduled”)
12. CourseOffering 12 for Project Based **(not submitted by SIS vendors) - (Requires** **a single course offering linked to a single course which is associated to all project-based college courses)**
    * 1. CourseReference (reference to district course)
      2. LocalCourseCode (District Course’s Code - at discretion of district)
      3. Schoolid
      4. SessionReference (SchoolYear, SchoolId, SessionName = “Unscheduled”)

### Resource: Section

**Description**

Ed-Fi Description: This entity represents a setting in which organized instruction of course content is provided, in-person or otherwise, to one or more students for a given period of time. A course offering may be offered to more than one section. Most MCCC CourseSectionInfo elements will be collected in the Ed-Fi Section Entity.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* State, College and District Courses (published to ODS by MDE)
* Sessions
* CourseOffering

**Scenarios**

**MCCC**

1. Create 9 Section Records to associate with the first 9 course offerings (not the IS, Project Based or Direct Pay PSEO courses yet – those will be done in step 2)
   1. Include the following elements:
      1. SectionIdentifier
      2. CourseOfferingReference
      3. ClassPeriod (for Scheduled Section Enrollments)
      4. SectionCharacteristicDescriptor (FP “Fixed Period Indicator” for courses associated to a fixed period, or MA “Marking Indicator” for sections where grades are recorded)
      5. InstructionLanguageDescriptor (only for non-English, using MARSS language descriptor)
      6. MediumOfInstructionDescriptor (required)
      7. SequenceOfCourse (must be less than or equal to Section Limit of course referenced)
2. Create 3 Section Records. 1 for each of the following course offerings:
   1. Independent Study (a section is required for every course offering)
      1. SectionIdentifier: IS\_<LocalCourseCode>\_Section
      2. CourseOfferingReference
   2. Direct Pay PSEO (a single placeholder section is required all Direct Pay PSEO)
      1. SectionIdentifier: DirectPayPSEO\_Section
      2. CourseOfferingReference
   3. Project Based (a single placeholder section is required all Project Based)
      1. SectionIdentifier: ProjectBased\_Section
      2. CourseOfferingReference

### Resource: StaffSectionAssociation

**Description**

Ed-Fi Description: This association indicates the class sections to which a staff member is assigned.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* Staff (published to ODS by MDE)
* Sections

**Scenarios**

**MCCC**

1. Create StaffSectionAssociation for sections created for Course Level Types (A, B, C, D, E, G, N, X, P)
   1. Include the following elements:
      1. StaffReference.staffUniqueId (FFN)
      2. ClassroomPositionDescriptor = ‘TOR’
      3. SectionReference

### Resource: StudentSectionAssociation

**Description**

Ed-Fi Description: This association indicates the course sections to which a student is assigned.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* Students
* Sections

**Scenarios**

**MCCC**

1. Create 1 StudentSectionAssociation Record for each Section loaded
   1. Include the following elements:
      1. StudentReference.studentUniqueId
      2. SectionReference
      3. BeginDate
      4. College Course Reference (only applicable to Direct Pay PSEO)
      5. Early Education fields (only for Couse Level Type ‘P’)
         1. instructionalApproachDescriptor
         2. implementationStatusDescriptor
         3. siteBasedInitiativeDescriptor
         4. implementationStatusDescriptor
      6. SectionEnrollmentType
         1. Match the enrollment type with the appropriate course/section.
            1. Scheduled
            2. Independent Study
            3. Direct Pay PSEO
            4. Project Based (if available)

### Resource: Grade

**Description**

Ed-Fi Description: This educational entity represents an overall score or assessment tied to a course over a period of time (i.e., the grading period). Student grades are usually a compilation of marks and other scores. Most MCCC StudentSectionMark elements in Ed-Fi will be collected in the Ed-Fi Grade Entity.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* Students
* Sections
* StudentSectionAssociations

**Scenarios**

**MCCC**

1. Create 1 Grade Record for each StudentSectionAssociation loaded
   1. Include the following elements:
      1. StudentSectionAssociationReference
      2. CollegeCreditsEarned
      3. CollegeGradeEarned
      4. LocalCreditEarned
      5. NumericGradeEarned
      6. LetterGradeEarned
      7. SectionEnrollmentType

### Resource: Programs

**Program Records for each of the following program types will be loaded by MDE with the StateEducationAgencyid =** 999999000

* The educationOrganizationReference for the StudentProgramAssociation is the **SchoolId**.
* The educationOrganizationReference on the StudentProgramAssociation’s programReference is the **StateEducationAgencyid (**999999000).

## StudentProgramAssociations

### Resource: Student21stCenturyLearningCenterGrantProgramAssociation

**Description**

This association represents Students in the 21st Century Community Learning Centers Grant Program.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “21st Century Learning Center Grant”
* Students

**Scenarios**

* Associate Student 1 with this StudentProgramAssociation

### Resource: StudentCEISProgramAssociation

**Description**

This association represents Students in the Coordinated Early Intervening Service Program.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “Coordinated Early Intervening Services”
* Students

**Scenarios**

* Associate Student 2 with this StudentProgramAssociation
* Delete this Student Program Association
* Delete Student 2’s Middle School StudentEducationOrganizationAssociation record.
* Delete student 2’s Middle School enrollment record.

### Resource: StudentEarlyChildhoodScreeningProgramAssociations

**Description**

This association represents Students in the Early Childhood Screening Association Program.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “Early Childhood Screening”
* Students

**Scenarios**

1. Associate Student 3 with this StudentProgramAssociation
2. Change the earlyChildhoodScreenerDescriptor to ‘Head Start’ and earlyChildhoodScreeningExitStatusDescriptor to ‘Rescreen’

### Resource: StudentGiftedTalentedProgramAssociation

**Description**

This association represents Students in the Gifted Talented Program.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “Gifted and Talented”
* Students

**Scenarios**

1. Associate Student 4 with this StudentProgramAssociation
2. Change the gifted talented participation code to ‘Full-time services’

### Resource: StudentADSISProgramAssociation

**Description**

This association represents Students in the Alternative Delivery of Specialized Instruction (ADSIS) Program.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “Alternative Delivery Of SIS”
* Students

**Scenarios**

1. Associate Student 5 with this StudentProgramAssociation

### Resource: StudentHomelessProgramAssociation

**Description**

This association represents the McKinney-Vento Homeless Program program(s) that a student participates in or from which the Student receives services.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “Homeless”
* Students

**Scenarios**

1. Associate Student 6 with this StudentProgramAssociation
2. Change the primary nighttime residence code to ‘Doubled-Up’ and Unaccompanied Youth to True
3. Submit a StudentHomelessProgramAssociation for the EE enrolled student
   1. Primary Nighttime Residence = SHR
   2. Begin Date 9/5/2019
   3. End Date 12/25/2020

### Resource: StudentLanguageInstructionProgramAssociation

**Description**

This association represents the Title III Language Instruction for Limited English Proficient and Immigrant Students program(s) that a student participates in or from which the Student receives services.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “English Learner”
* Students

**Scenarios**

1. Associate Student 7 with this StudentProgramAssociation where English Learner Participation is set to True (Indicates EL Served)
2. Change the English Learner Participation to False, to Indicate EL Identified, but not served
3. Associate another student with this program and set the language service code to ‘Newcomer Program’

### Resource: StudentPSEOConcurrentProgramAssociation

**Description**

This association represents Students in the PSEO Concurrent Enrollment Program.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “PSEO Concurrent”
* Students

**Scenarios**

1. Associate Student 8 with this StudentProgramAssociation

### Resource: StudentPSEOProgramAssociation

**Description**

This association represents Students in the PSEO Program.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “PSEO Program”
* Students

**Scenarios**

1. Update PSEO High School Hours to 650
2. Associate Student 9 with this StudentProgramAssociation

### Resource: StudentSAAPProgramAssociation

**Description**

This association represents Students in a State Approved Alternative Program.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “SAAP”
* Students

**Scenarios**

1. Associate Student 10 with this StudentProgramAssociation
2. Change SAAP Credits to 5.75 and Set Independent Study indicator to True and SAAP concurrent indicator to Yes

### Resource: StudentSchoolFoodServiceProgramAssociation

**Description**

This association represents the school food services program(s), such as the Free or Reduced Lunch Program, that a student participates in or from which the Student receives services.

Note: in School Year 18-19 schoolFoodServicesEligibility was tracked as a separate field under StudentSchoolAssociation. This element is now tracked using SchoolFoodServiceProgramService on studentSchoolFoodServicesProgramAssociation.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “School Food Service”
* Students

**Scenarios**

1. Associate Student 11 with this StudentProgramAssociation with SchoolFoodServiceProgramService = 2
2. Change FRP Meal Code to 1

### Resource: StudentSection504PlanProgramAssociation

**Description**

Students who have a Section 504 plan.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “Section 504 Plan”
* Students

**Scenarios**

1. Associate Student 12 with this StudentProgramAssociation

### Resource: StudentSpecialEducationProgramAssociation

**Description**

This association represents the special education program(s) that a student participates in or receives services from. The association is an extension of the StudentProgramAssociation particular for special education programs.

**Important Notes on Order of Disability**

MDE collects the primary disability code for a student during an enrollment period.

* Disability codes must be reported to Ed-Fi with the 'order of disability' as 1 to designate the primary disability.
* Although additional disability codes for a student might be reported for a particular begin date, they will be ignored by MDE and not collected or stored in the MDE transformations. In other words, only the disability code with ‘order of disability’ 1 will be used for any begin date.
* If multiple disability codes for a student are reported with the same begin date, there must only be one with an order of disability = 1.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “Special Education”
* Students

**Scenarios**

1. Associate Student 13 with this StudentProgramAssociation include a disability with order of priority = 1
2. Change disability code to 12, set special education service hours to 1100.5

### Resource: StudentTitleIPartAProgramAssociations

**Description**

This association represents the Title I Part A program(s) that a student participates in or from which the Student receives services. The association is an extension of the StudentProgramAssociation particular for Title I Part A programs.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “Title I Part A”
* Students

**Scenarios**

1. Associate Student 14 with this StudentProgramAssociation

## GRR Certification Scenarios - API Resources

### Resource: ExamReimbursementRequest

**Description**

MN Extension: Exam reimbursement request (formerly known as Graduate Required Records).

**Prerequisite Data**

* Student
* StudentSchoolAssociation (enrollment)
* StudentEducationOrganizationAssociation (demographic record)

**Scenarios**

1. Submit an examReimbursementRequest for a high school student for the ACT writing assessment who is unable to pay
   1. assessmentCategoryDescriptor.codevalue = 0008
   2. reimbursementRequested = True
   3. Unable to pay = true
2. Submit an examReimbursementRequest for a high school student for the ACT assessment who is unable to pay
   1. assessmentCategoryDescriptor.codevalue = 0007
   2. reimbursementRequested = True
   3. Unable to pay = true
3. Delete the first record
4. Update the second record as follows:
   1. reimbursementRequested = False
   2. Unable to pay = False

## MCCC Certification Scenarios - API Resources

### Resource: Students

**Description**

This entity represents an individual for whom instruction, services, and/or care are provided in an early childhood, elementary, or secondary educational program under the jurisdiction of a school, education agency or other institution or program. A student is a person who has been enrolled in a school or other educational institution.

**Prerequisite Data**

* None

**Certification Scenarios**

1. Create 12 student records including the following elements.
   1. studentUniqueID
   2. birthdate
   3. birthSexDescriptor
   4. firstName
   5. generationCodeSuffix
   6. lastSurname
   7. middleName

### Resource: StudentSchoolAssociations

**Description**

This association represents the School in which a student is enrolled. The semantics of enrollment may differ slightly by state. Non-enrollment relationships between a student and an education organization may be described using the StudentEducationOrganizationAssociation.

**Prerequisite Data**

* Students
* Schools

**Scenarios**

1. Enroll the 12 students (spread across elementary, middle and high school)

### Resource: StudentEducationOrganizationAssociation

**Description**

This association indicates any relationship between a student and an education organization other than how the state views enrollment. Enrollment relationship semantics are covered by StudentSchoolAssociation.

MDE allows for the capture of student demographic data by school enrollment. Therefore, a StudentEducationOrganizationAssociation record must be submitted for each **school** in which the student is enrolled to provide the student demographic data provided to the enrolling school by the parent(s).

**Prerequisite Data**

* Students
* EducationOrganizations (Schools)

**Scenarios**

1. Create StudentEducationOrganizationAssociation records for each of the students

### Resource: Calendar

**Description**

Ed-Fi Description: A set of dates associated with an organization. MDE is not using the Calendar entity as collection of dates, rather MDE captures the following key pieces of Calendar Metadata in the Calendar file: Instructional Days, Length of Day, and Kindergarten Schedule (when applicable). Calendar is captured at the School Level by grade.

**Prerequisite Data**

* Schools (published to ODS by MDE)

**Scenarios**

1. Create MCCC Calendar for elementary, middle and high schools including
   1. Calendar Code
   2. School ID
   3. School Year
   4. CalendarType = “MCCC”
   5. Days In Session
   6. Instructional Days
   7. Description

### Resource: CalendarDate

**Description**

Ed-Fi Description: The type of scheduled or unscheduled event for the day. MDE will be collecting the MCCC ScheduleInfo Start and End Dates using Calendar Date.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Calendar

**Scenarios**

1. Create 2 Calendar Dates for each of the 3 calendars
   1. Date 1 – calendarEventDescriptor = STARTDT
   2. Date 2 – calendarEventDescriptor = ENDDT
   3. Both Dates should reference the MCCC Calendar created above

### Resource: ClassPeriod

**Description**

Ed-Fi Description: This entity represents the designation of a regularly scheduled series of class meetings at designated times and days of the week. MDE will be collecting the MCCC PeriodInfo Data in this resource. This data must be submitted for each school.

**Prerequisite Data**

* Schools (published to ODS by MDE)

**Scenarios**

1. Create a series of Class Periods for elementary, middle and high school
   1. Each Period must include:
      1. ClassPeriodName
      2. ClassPeriodDescriptor
      3. SchoolID
      4. ClassPeriodDescription
      5. StartTime
      6. EndTime

### Resource: GradingPeriod

**Description**

Ed-Fi Description: This entity represents the time span for which grades are reported. MCCC collection in Ed-Fi requires the creation of GradingPeriods in order to allow submission of grades. Grading Periods are defined at the School Level.

**Prerequisite Data**

* Schools (published to ODS by MDE)

**Scenarios**

1. Create a series of GradingPeriods for elementary, middle and high school

Each Period must include:

GradingPeriodDescriptor

PeriodSequence

SchoolId

SchoolYear

### Resource: Session

**Description**

Ed-Fi Description: A term in the school year, generally a unit of time into which courses are scheduled, instruction occurs and by which credits are awarded. Sessions may be interrupted by vacations or other events. The MCCC collection in Ed-Fi will use Session for collection of TermInfo.

**Prerequisite Data**

* Schools (published to ODS by MDE)

**Scenarios**

1. Create Sessions for elementary, middle and high school (Fall, Spring, and non-scheduled)
   1. Each Period must include:
      1. TermDescriptor (S for Semester, NS for non-scheduled)
      2. School Year
      3. SchoolId
      4. DaysInSession
      5. TotalInstructionalDays
      6. SessionName
      7. BeginDate
      8. EndDate
      9. TermNumber

### Resource: Course

**Description**

Ed-Fi Description: This educational entity represents the organization of subject matter and related learning experiences provided for the instruction of students on a regular or systematic basis. The MCCC collection in Ed-Fi will use Course for collection of District, College and State Courses.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* State Courses (published to ODS by MDE)

**Scenarios**

1. Create 8 **District** Courses (set up at least 1 course per school type – Elementary, Middle and High School)
   1. Course 1 Course Level Type = B
   2. Course 2 Course Level Type = G
   3. Course 3 Course Level Type = E
   4. Course 4 Course Level Type = D
   5. Course 5 Course Level Type = A
   6. Course 6 Course Level Type = C
   7. Course 7 Course Level Type = X
   8. Course 8 Course Level Type = N

Including the following elements:

* + 1. CourseDescription (only when associated with an Unclassified State Course)
    2. HighSchoolCourseRequirement
    3. CourseLevelCharacteristic for End of Course Indicator
    4. Course Code (Local)
    5. SequenceLimit
    6. LocalEducationAgencyId
    7. CourseTitle
    8. CourseDefinedByDescriptorId = “LEA”
    9. CourseIdentificationCode – repeat the CourseCode (this is an ed-fi requirement)
    10. CourseIdentificationSystemDescriptor = ‘LEA course code’
    11. *LearningStandardid* (not currently required)

1. Create an EE District Course   
   Elements that must have valid values:
   1. Course Description (only required when associated with Unclassified State Course)
   2. Course Code (local)
   3. Course level Type (P)
   4. Sequence Limit (1)
   5. StandardId (State)
   6. High School Course Requirement (False)
   7. **Must NOT include CourseLevelCharacteristic for End of Course Indicator**
   8. CurriculumUsed
      1. CurriculumUsedDescriptor (CRCUE, HWOT)
      2. ImplementationStatusDescriptor (FULL, CEXP)
   9. AssessmentTool
      1. AssessmentToolDescriptor
      2. ImplementationStatusDescriptor
   10. CourseDefinedByDescriptorId = ‘LEA”
       1. CourseIdentificationCode – repeat the CourseCode (this is an ed-fi requirement)
       2. CourseIdentificationSystemDescriptor = ‘LEA course code’
   11. LearningStandardid (99.E5.1)
2. District Courses to associate with College courses
   1. Create additional District courses to be associated with college course levels D and A
3. Create **College** Courses
   1. 1 with Course Level Type = D
   2. 1 with Course Level Type = A
   3. 2 college courses to be used for Direct Pay PSEO

Including the following elements:

* + 1. PostSecondaryInstitutionId
    2. MaximumAvailableCredits
    3. CourseCode (College Course Identifier – generally dept letters & course number)
    4. CourseTitle
    5. CourseDefinedByDescriptorId = ‘College”
    6. CourseIdentificationCode – repeat the CourseCode (this is an ed-fi requirement)
    7. CourseIdentificationSystemDescriptor = ‘University course code’

1. Create a District Course for Independent Study

**NOTE:** Independent Study course section enrollments did not previously require the setup of a course section – the Ed-Fi Model enforces the full set of master schedule entities (course, course offering and section) in order to associate a student with a course and record grades for that course.

Separate course, course offering and section records are required for your district to report ALC Independent Study records; however Course Offering and Section will not be used by MDE for reporting. (The Ed-Fi Model requires an entry in Course Offering linked to the non-scheduled term, and a section default.)

* 1. Include the following elements:
     1. CourseLevelCharacteristic = IS
     2. Course Code
     3. CourseDescription
     4. HighSchoolCourseRequirement
     5. SequenceLimit
     6. LocalEducationAgencyId
     7. CourseTitle
     8. CourseDefinedByDescriptorId = ‘LEA”
     9. CourseIdentificationCode – repeat the CourseCode (this is an ed-fi requirement)
     10. CourseIdentificationSystemDescriptor = ‘LEA course code’
     11. LearningStandardid (k-12 Courses)

1. **Create a course** for Direct Pay PSEO section enrollments – the Ed-Fi Model requires course, course offering and section. At minimum, a **single Placeholder course, course offering, and section will be required for your district** to report Direct Pay PSEO Student Section Association records.
   1. Ed-Fi required elements:
      1. Course Code: Direct Pay PSEO
      2. Course Title
      3. Course Identification Code
      4. Number of Parts
      5. CourseDefinedByDescriptor = “LEA’
2. **Not applicable to SIS Vendors - Project Foundry Only -** Create a course for Project Based section enrollments – While these types of enrollments do not include MDE course, course offering or section requirements – the Ed-Fi Model enforces these entities in the Master Schedule. At minimum, a **single Placeholder course, course offering, and section will be required for your district** to report Project Based Student Section Association records.
   1. Ed-Fi required elements:
      1. Course Code: Project Based
      2. CourseLevelCharacteristic = PBL
      3. Course Title
      4. Course Identification Code
      5. Number of Parts
      6. CourseDefinedByDescriptor = “LEA’
3. Create CourseCourseAssociation records between the district Courses and State Level Courses (including IS, and Project Based if applicable)
   1. EducationOrganizationId = District ID
   2. CourseCode = District Course Code
   3. ToCourseEducationOrganizationId = State Course EducationOrganizationId
   4. ToCourseCode = State Course Code
      1. IS can only use state codes with CourseLevelCharacteristicDescription = IS and Project Based can only use state codes with CourseLevelCharacteristicDescription = PBL.
      2. Scheduled course work type cannot use CourseLevelCharacteristicDescription = PBL.
4. Create CourseCourseAssociation records between the College Courses A and D and the District Courses created for this association.
   1. EducationOrganizationId = District ID
   2. CourseCode = District Course Code
   3. ToCourseEducationOrganizationId = College Course’s PostSecondaryInstitutionId
   4. ToCourseCode = College Course Code
5. Create CourseCourseAssociation records between the Direct Pay College Courses and the applicable State Course (PSEO Direct Pay)
   1. EducationOrganizationId = College Education Organization ID
   2. CourseCode = College Course Code
   3. ToCourseEducationOrganizationId = State SEA id
   4. ToCourseCode = State Course Code

### Resource: CourseOffering

**Description**

Ed-Fi Description: This entity represents an entry in the course catalog of available courses offered by the school during a session. The MCCC collection in Ed-Fi will use Course Offering for collection of LocalCourseInformation that is not captured within the Ed-Fi Course record. The CourseOffering also links up the Course to the term and section, student and staff allowing grades and staff assignments to be captured.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* State, College and District Courses
* Sessions

**Scenarios**

Create the following CourseOffering Records

1. CourseOffering 1 References the District Course with Level Type = B
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
2. CourseOffering 2 References the District Course with Course Level Type = G
   * 1. CourseReference
     2. LocalCourseCode (this can match the District Course’s Course Code – at the discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
3. CourseOffering 3 References the District Course with Course Level Type = E
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
4. CourseOffering 4 References the District Course with Course Level Type = D
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
5. CourseOffering 5 References the District Course with Course Level Type = A
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
6. CourseOffering 6 References the District Course Associated with Course Level Type = C CourseReference
   * 1. LocalCourseCode (this can match the District Course’s Course Code – at the discretion of district)
     2. SchoolId
     3. SessionReference (SchoolYear, SchoolId, SessionName)
     4. InstructionMinutesPerTerm
7. CourseOffering 7 References the District Course with Course Level Type = X
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
8. CourseOffering 8 References the District Course with Course Level Type = N
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
9. CourseOffering 9 References the District Course Associated with Course Level Type = P
   * 1. CourseReference
     2. LocalCourseCode (District Course’s Code - at discretion of district)
     3. SchoolId
     4. SessionReference (SchoolYear, SchoolId, SessionName)
     5. InstructionMinutesPerTerm
     6. InstructionalApproach
        1. InstructionalApproachDescriptor
        2. ImplementationStatusDescriptor
     7. siteBasedInitiative
        1. siteBasedInitiativeDescriptor
        2. ImplementationStatusDescriptor
10. CourseOffering 10 for Independent Study **(you will have one course offering for every Ind Study district course)**
    * 1. CourseReference (reference to district course)
      2. LocalCourseCode (District Course’s Code - at discretion of district)
      3. Schoolid
      4. SessionReference (SchoolYear, SchoolId, SessionName = “Unscheduled”)
11. CourseOffering 11 for Direct Pay PSEO **(you will have a single course offering linked to a single district course set up as a placeholder)**
    * 1. CourseReference (reference to district course)
      2. LocalCourseCode (District Course’s Code - at discretion of district)
      3. Schoolid
      4. SessionReference (SchoolYear, SchoolId, SessionName = “Unscheduled”)
12. CourseOffering 12 for Project Based **(not submitted by SIS vendors) - (Requires** **a single course offering linked to a single course which is associated to all project-based college courses)**
    * 1. CourseReference (reference to district course)
      2. LocalCourseCode (District Course’s Code - at discretion of district)
      3. Schoolid
      4. SessionReference (SchoolYear, SchoolId, SessionName = “Unscheduled”)

### Resource: Section

**Description**

Ed-Fi Description: This entity represents a setting in which organized instruction of course content is provided, in-person or otherwise, to one or more students for a given period of time. A course offering may be offered to more than one section. Most MCCC CourseSectionInfo elements will be collected in the Ed-Fi Section Entity.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* State, College and District Courses (published to ODS by MDE)
* Sessions
* CourseOffering

**Scenarios**

1. Create 9 Section Records to associate with the first 9 course offerings (not the IS, Project Based or Direct Pay PSEO courses yet – those will be done in step 2)
   1. Include the following elements:
      1. SectionIdentifier
      2. CourseOfferingReference
      3. ClassPeriod (for Scheduled Section Enrollments)
      4. SectionCharacteristicDescriptor (FP “Fixed Period Indicator” for courses associated to a fixed period, or MA “Marking Indicator” for sections where grades are recorded)
      5. InstructionLanguageDescriptor (only for non-English, using MARSS language descriptor)
      6. MediumOfInstructionDescriptor (required)
      7. SequenceOfCourse (must be less than or equal to Section Limit of course referenced)
2. Create 3 Section Records. 1 for each of the following course offerings:
   1. Independent Study (a section is required for every course offering)
      1. SectionIdentifier: IS\_<LocalCourseCode>\_Section
      2. CourseOfferingReference
   2. Direct Pay PSEO (a single placeholder section is required all Direct Pay PSEO)
      1. SectionIdentifier: DirectPayPSEO\_Section
      2. CourseOfferingReference
   3. Project Based (a single placeholder section is required all Project Based)
      1. SectionIdentifier: ProjectBased\_Section
      2. CourseOfferingReference

### Resource: StaffSectionAssociation

**Description**

Ed-Fi Description: This association indicates the class sections to which a staff member is assigned.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* Staff (published to ODS by MDE)
* Sections

**Scenarios**

1. Create StaffSectionAssociation for sections created for Course Level Types (A, B, C, D, E, G, N, X, P)
   1. Include the following elements:
      1. StaffReference.staffUniqueId (FFN)
      2. ClassroomPositionDescriptor = ‘TOR’
      3. SectionReference

### Resource: StudentSectionAssociation

**Description**

Ed-Fi Description: This association indicates the course sections to which a student is assigned.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* Students
* Sections

**Scenarios**

1. Create at least 1 StudentSectionAssociation Record for each Section loaded
   1. Include the following elements:
      1. StudentReference.studentUniqueId
      2. SectionReference
      3. BeginDate
      4. College Course Reference (only applicable to Direct Pay PSEO)
      5. Early Education fields (only for Couse Level Type ‘P’)
         1. instructionalApproachDescriptor
         2. implementationStatusDescriptor
         3. siteBasedInitiativeDescriptor
         4. implementationStatusDescriptor
      6. SectionEnrollmentType
         1. Match the enrollment type with the appropriate course/section.
            1. Scheduled
            2. Independent Study
            3. Direct Pay PSEO
            4. Project Based (if available)

### Resource: Grade

**Description**

Ed-Fi Description: This educational entity represents an overall score or assessment tied to a course over a period of time (i.e., the grading period). Student grades are usually a compilation of marks and other scores. Most MCCC StudentSectionMark elements in Ed-Fi will be collected in the Ed-Fi Grade Entity.

**Prerequisite Data**

* Schools (published to ODS by MDE)
* Districts (published to ODS by MDE)
* Students
* Sections
* StudentSectionAssociations

**Scenarios**

1. Create 1 Grade Record for each StudentSectionAssociation loaded
   1. Include the following elements:
      1. StudentSectionAssociationReference
      2. CollegeCreditsEarned
      3. CollegeGradeEarned
      4. LocalCreditEarned
      5. NumericGradeEarned
      6. LetterGradeEarned
      7. SectionEnrollmentType

## Early Education Enrollment Certification Scenarios - API Resources

### Resource: StudentSchoolAssociations

**Description**

This association represents the School in which a student is enrolled. The semantics of enrollment may differ slightly by state. Non-enrollment relationships between a student and an education organization may be described using the StudentEducationOrganizationAssociation.

**Prerequisite Data**

* Students
* Schools

**Certification Scenarios**

1. Submit an Early Education Enrollment record (StudentSchoolAssociation with entrygradelevel = EE)
   1. Required elements
      1. School Id
      2. Student Unique Id
      3. Entry Grade Level = EE
      4. Entry Type (Last Location Code) = 0
      5. Entry Date = 9/5/2020
      6. Exit Withdraw Date (Defaulted to end of school year)
      7. Exit Withdraw Type = 20
      8. Homebound Service Indicator = N
      9. Resident Local Education Agency
      10. Special Pupil Indicator = N
   2. Data **not** collected for Early Education
      1. State Aid Category
      2. Membership Attendance Units
      3. Membership Attendance Percent Enrolled
      4. Special Education Eval Status
      5. Transportation
2. Submit an Early Education Enrollment record (StudentSchoolAssociation with entrygradelevel = 1)
   * 1. School Id
     2. Student Unique Id
     3. Entry Grade Level = 1
     4. Entry Type (Last Location Code) = 4
     5. Entry Date = 9/6/2020
     6. Exit Withdraw Date (Defaulted to end of school year)
     7. Exit Withdraw Type = 40
     8. Homebound Service Indicator = N
     9. Resident Local Education Agency
     10. Special Pupil Indicator = N
3. Submit an Early Education Enrollment record (StudentSchoolAssociation with entrygradelevel = PA)
   * 1. School Id
     2. Student Unique Id
     3. Entry Grade Level = PA
     4. Entry Type (Last Location Code) = 4
     5. Entry Date = 9/6/2020
     6. Exit Withdraw Date (Defaulted to end of school year)
     7. Exit Withdraw Type = 40
     8. Homebound Service Indicator = N
     9. Resident Local Education Agency
     10. Special Pupil Indicator = N

### Resource: Parent

**Description**

This entity represents a parent or guardian of a student, such as mother, father, or caretaker.

**Prerequisite Data**

None

**Scenarios**

1. Submit Parent Records for the following Early Ed Student in Gradelevel EE
   * 1. Required Elements:
        1. Parent Unique Id
        2. firstName
        3. lastSurname
        4. MiddleName
        5. generationCodeSuffix
        6. highestCompletedLevelOfEducationDescriptor
        7. householdIncome
        8. householdSize
        9. receivingInterpreterAssistance
        10. classroomVolunteerDescriptor
        11. birthDate
        12. sexDescriptor
   1. Submit Parent Records for the following Early Ed Student in Gradelevel 1
   2. Submit Parent Records for the following Early Ed Student in Gradelevel PA

### Resource: StudentParentAssociation

**Description**

This association relates students to their parents, guardians, or caretakers.

**Prerequisite Data**

* Students
* Parents
* StudentSchoolAssociations

**Scenarios**

1. Submit Student Parent Association Record for the Early Education Student in Gradelevel EE
2. Required Elements:
   * 1. Parent Unique Id
     2. Student Unique Id
     3. Relation Descriptor
3. Submit Student Parent Association Record for the Early Education Student in Gradelevel 1
4. Submit Student Parent Association Record for the Early Education Student in Gradelevel PA

### Resource: StudentEducationOrganizationAssociation

**Description**

This association indicates any relationship between a student and an education organization other than how the state views enrollment. Enrollment relationship semantics are covered by StudentSchoolAssociation.

MDE allows for the capture of student demographic data by school enrollment. Therefore, a StudentEducationOrganizationAssociation record must be submitted for each **school** in which the student is enrolled to provide the student demographic data provided to the enrolling school by the parent(s).

**Prerequisite Data**

* Students
* EducationOrganizations (Schools)

**Scenarios**

1. Create a StudentEducationOrganizationAssociation for the Student Enrolled in Gradelevel EE
   1. Include all elements except StudentCharacteristicsDescriptors (ADP, RAEL, IMMIGRANT, SLIFE)
2. Create a StudentEducationOrganizationAssociation for the Student Enrolled in Gradelevel 1
3. Create a StudentEducationOrganizationAssociation for the Student Enrolled in Gradelevel PA

### Resource: StudentEarlyEducationProgramAssociations

**Description**

This association represents Students in either School Readiness or Early Childhood Family Education programs. Notes:

* The "EE-SR" and "EE-ECFE" program types are the programs intended for use with Early Education Data
* "End Reason Code" below correlates to "reasonExitedDescriptor" (see the similarly named Data Mapping Matrix tab)

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “SR”
* Program - where ProgramTypeDescriptor = “ECFE”
* Students

**Scenarios**

1. Associate gradelevel EE Enrolled Student with this SR StudentEarlyEducationProgramAssociation
   1. Begin Date: 9/5/2020
   2. End Date: 6/11/2021 (default to last day in the school year)
   3. End Reason Code = PE-01
   4. Membership: 640
   5. Attendance: 580
   6. Submit funding sources PF and CC
2. Associate same gradelevel EE Enrolled Student with this ECFE StudentEarlyEducationProgramAssociation
   1. Begin Date = 9/5/2020
   2. End Date = 10/15/2020
   3. End Reason Code = PE-02
   4. Membership Hours = 40
   5. Attendance Hours = 30
   6. Submit funding sources PF, CC and TITLE
3. Associate gradelevel 1 Enrolled Student with an ECFE StudentEarlyEducationProgramAssociation
   1. Begin Date: 10/6/2020
   2. End Date: 6/11/2020 (default to last day in the school year)
   3. Membership: 640
   4. Attendance: 580
   5. Submit funding sources ECFE and CC
4. Associate gradelevel PA Enrolled Student with an SR StudentEarlyEducationProgramAssociation
   1. Begin Date: 9/7/2020
   2. End Date: 6/11/2020 (default to last day in the school year)
   3. Membership: 640
   4. Attendance: 580
   5. Submit funding sources CSPF and OD

### Resource: StudentHomelessProgramAssociation

**Description**

This association represents the McKinney-Vento Homeless Program program(s) that a student participates in or from which the Student receives services.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “Homeless”
* Students

**Scenarios**

1. Submit a StudentHomelessProgramAssociation for the EE enrolled student
   1. Primary Nighttime Residence = SHR
   2. Begin Date = 9/5/2019
   3. End Date = 12/25/2020

### Resource: StudentLanguageInstructionProgramAssociation

**Description**

This association represents the Title III Language Instruction for Limited English Proficient and Immigrant Students program(s) that a student participates in or from which the Student receives services.

**Prerequisite Data**

* Schools
* Program - where ProgramTypeDescriptor = “English Learner”
* Students

**Scenarios**

1. Submit StudentLanguageInstructionProgramAssociation for EE enrolled Student
   1. Begin Date = 9/5/2019
   2. End Date = 5/25/2021
   3. Lang Service Code = Transitional Bilingual
   4. EL Participation = 1

## 2021-2022 MDE Submitted Data Requirements - API Resources

### Resource: LocalEducationAgencies

**Description**

This entity represents an administrative unit at the local level which exists primarily to operate schools or to contract for educational services. It includes school districts, charter schools, charter management organizations, or other local administrative organizations.

**Prerequisite Data**

* None

Note: All required Local Education Agency data will be loaded by MDE

### Resource: Schools

**Description**

This entity represents an educational organization that includes staff and students who participate in classes and educational activity groups.

**Prerequisite Data**

* None

Note: All required School data will be loaded by MDE.

### Resource: Post-Secondary Institutions

**Description**

An organization that provides educational programs for individuals who have completed or otherwise left educational programs in secondary school(s).

**Prerequisite Data**

* None

Note: All required College data will be loaded by MDE.

### Resource: Course (State Level Only)

**Description**

This educational entity represents the organization of subject matter and related learning experiences provided for the instruction of students on a regular or systematic basis.

**Prerequisite Data**

* Education Organization for SEA

Note: All required State Course data will be loaded by MDE.

### Resource: Staff

**Description**

This entity represents an individual who performs specified activities for any public or private education institution or agency that provides instructional and/or support services to students or staff at the early childhood level through high school completion. For example, this includes:

1. An "employee" who performs services under the direction of the employing institution or agency is compensated for such services by the employer and is eligible for employee benefits and wage or salary tax withholdings.
2. A "contractor" or "consultant" who performs services for an agreed upon fee or an employee of a management service contracted to work on site.
3. A "volunteer" who performs services on a voluntary and uncompensated basis.
4. An in-kind service provider.
5. An independent contractor or businessperson working at a school site.

**Prerequisite Data**

* none

Note: Sync process will be created by MDE to populate Staff Tables

### Resource: Program

**Description**

Ed-Fi Description: This entity represents any program designed to work in conjunction with, or as a supplement to, the main academic program. Programs may provide instruction, training, services, or benefits through federal, state, or local agencies. Programs may also include organized extracurricular activities for students. MDE will create the following programs with these program types:

* Section 504 Plan
* Title I Part A
* English Learner
* Gifted and Talented
* Special Education
* 21st Century Learning Center Grant
* Alternative Delivery Of SIS
* Coordinated Early Intervening Services
* Homeless
* PSEO Concurrent
* PSEO Program
* School Food Service
* Early Childhood Screening
* SAAP
* School Readiness
* Early Childhood Family Education

## Ed-Fi Model Dependency Graph (MARSS collection)

Descriptors

Calendar

Schools and LEAs

Student Records

Programs

StudentSchoolAssociation

(Enrollment Records)

Student Program Associations

StudentEducationOrganizationAssociation (Demographic Records)

1. Descriptors must be loaded first, as all other resources contain references to descriptor values.
2. Once descriptors are loaded, Education Organization data must be loaded by MDE.
3. Though MARSS is centered around student demographics and enrollment, the core Student records must be loaded before students may be enrolled.
4. Student enrollment data must be provided via StudentSchoolAssociation in order to establish a valid security claim before any other updates may be made to student records.
5. Once student enrollment data is loaded, student demographics may be provided via StudentEducationOrganizationAssociation.
6. Programs must be loaded for all districts submitting data to the ODS.
7. StudentProgramAssociations can be loaded once Programs and StudentSchoolAssociations have been loaded.

## Ed-Fi Model Dependency Graph (MCCC collection)

Descriptors

Calendar

Education Organizations (Schools, LEAs, Colleges, SEA)

Student Records

ClassPeriod

Courses

Course Offerings

Staff

Sections

Grade

Grading Period

Staff Section Association

Student Section Association

# Staging Environment Load and Quality Check

At the completion of Scenario based testing in the Sandbox environment, MNIT will grant vendors a key and secret for the district they are working with in the staging environment. In the staging environment, vendors will load actual student data, with student enrollment (studentSchoolAssociation), demographic (studentEducationOrganizationAssociation), program and calendar information.

Staging Environment URLs

[**https://stage.edfi.education.mn.gov/edfi.ods.webapi/**](https://stage.edfi.education.mn.gov/edfi.ods.webapi/)

## Ed-Fi / MARSS Identities API Integration Test

Vendors have the option of building in support for interfacing with the MN Student Identity System to create new MARSS Student IDs (Ed-FI Unique Id) through an Ed-Fi Identities API implementation. Vendors will be able to test their State ID creation integrations in the Staging environment.

Vendors must demonstrate their ability to create new Student IDs when a Student is created in the Student Information System.

This would be accomplished by posting to the /identities endpoint.

The URL for the identities API in the Staging environment will be:

<https://stage.edfi.education.mn.gov/edfi.ods.webapi/Identity/v2/2022/identities>

The general process for identity creation is as follows:

* A Vendor posts a new Student Identity including the following properties to the ODS/API identities end-point:

{

"stateStudentId": "000010000000",

"educationOrganizationId: "255901",

"lastSurname": "LastSurname",

"firstName": "FirstName",

"middleName": "MiddleName",

"generationCodeSuffix": "Suffix",

"birthDate": "2005-12-15T00:00:00",

"sexType": "Male"

}

* The identities API validates the student information against the MN Student Identity System.
* If no match is found in the Student Id System – the student id is created
* An HTTP response code 201 created is returned from the ODS/API.
* If the Student ID System returns an error the student id will fail to be created and an HTTP status code 400 Bad Request with the corresponding message for the error given is returned from the ODS/API (see specific errors below under Test 2)

### Create Student ID Test 1

Description: Vendor posts a new Student Identity to the ODS/API identities end-point.

Vendor submits Id with additional identifying attributes, and ID is created successfully

**Sample JSON**

{

"stateStudentId": "000010000000",

"educationOrganizationId: "255901",

"lastSurname": "LastSurname",

"firstName": "FirstName",

"middleName": "MiddleName",

"generationCodeSuffix": "Suffix",

"birthDate": "2005-12-15T00:00:00",

"sexType": "Male"

}

**Response**

201: created

### Create Student ID Test 2

Description: Vendor posts a new Ed-Fi Student record to the Identities endpoint for the following error conditions:

| **Error** | **Description** | **Details** |
| --- | --- | --- |
| 3000 | (Input SSID not equal SSID) | Student identifying information given matches a single student, but the State Student Id specified as input does not match the official State Student Id of the matched student. |
| 3001 | (Multiple SSIDs found) | Student identifying information given matches multiple existing students. |
| 3003 | (Possible Student Already Exists) | Student identifying information given does not match any students. However suspect student matches found using Clean Last Name Initial, Clean First Name Initial, Birth Date and gender equal. Review needed. |
| 3004 | (Possible Student Already Exists) | Student identifying information given does not match any students. However suspect student matches found using Clean Last Name, Clean First Name equal. Exact match. Review needed. |
| 3005 | (Possible Student Already Exists) | Student identifying information given does not match any students. However suspect student matches found using Soundex of Clean Last Name, Soundex of Clean First Name, Gender, and Birth Date + or - 1 year. Review needed. |
| 3006 | (Possible Student Already Exists) | Student identifying information given does not match any students. However, the State Student Id is already in use by a non-matched student. Review needed. |

**Response**

400: BAD REQUEST { “message”: “<error number> <error description><error details>” }

Vendors will be expected to demonstrate a mechanism for timely reporting of ID creation errors back to users, either through the SIS user interface, or through a separate interface, or reporting method.

## Ed-Fi / MARSS Student Record Validation Test

When Student Records are created or updated in the Ed-Fi ODS, the API will perform a look up against the MN State Student ID System to determine whether the student is associated to a valid student id.

This validation feature will be enabled in the Staging Environment. Vendors must capture validation errors returned in the API response and surface these errors to district users.

### Student Validation Process Overview

* SIS posts a new or updated Student record to the Ed-Fi API.
* The Ed-Fi student validator then calls the MDE Student ID System validation process with StateStudentId (StudentUniqueId), LastName, FirstName, BirthDate, and BirthSex.
  + If the MDE Student ID System returns 0 (Exact Match), the student identity matches all the information exactly and only a single student is found. The Ed-Fi API will allow the Student record to be stored in the ODS. The Ed-Fi API returns HTTP response code 204.
  + Any other return value from the MDE Student ID system validation process results in HTTP response code 400 Bad Request, which also passes back the error code and message from the MDE Student ID System. The SIS should present this error code and message to the district user, informing them of the Student ID issue which must be resolved in the MDE Student ID system.

### Validate Student ID Test 1

Description: Vendor posts a new valid Ed-Fi Student record to the ODS/API Student end-point.

Vendor submits Ed-Fi student record to the student’s resource endpoint using a valid student id and associated to the same student.

**Response**

201: created

### Validate Student ID Test 2

Description: Vendor posts a new Ed-Fi Student record to the ODS/API Student end-point for the following error conditions:

| **Error** | **Description** | **Details** |
| --- | --- | --- |
| 3000 | (Input SSID not equal SSID) | Student identifying information given matches a single student, but the State Student Id specified as input does not match the official State Student Id of the matched student. |
| 3001 | (Multiple SSIDs found) | Student identifying information given matches multiple existing students. |
| 3002 | (No match found0 | Student identifying information given does not match any students. Also no possible suspect student matches found. (May be safe to add as a new student.) |
| 3003 | (Possible Student Already Exists) | Student identifying information given does not match any students. However suspect student matches found using Clean Last Name Initial, Clean First Name Initial, Birth Date and gender equal. Review needed. |
| 3004 | (Possible Student Already Exists) | Student identifying information given does not match any students. However suspect student matches found using Clean Last Name, Clean First Name equal. Exact match. Review needed. |
| 3005 | (Possible Student Already Exists) | Student identifying information given does not match any students. However suspect student matches found using Soundex of Clean Last Name, Soundex of Clean First Name, Gender, and Birth Date + or - 1 year. Review needed. |
| 3006 | (Possible Student Already Exists) | Student identifying information given does not match any students. However, the State Student Id is already in use by a non-matched student. Review needed. |

**Response**

* The Ed-Fi Student record is unable to post.
* An HTTP status code 400 Bad Request and the corresponding message for the error given is returned from the ODS/API.
* Vendor Captures error and displays for district user.