

Ed-Fi 2.x Graduation Plan Entity Mapping

Overview

Use this page to help understand and define the mapping of Aeries data to Ed-Fi data



Status

Ready for certification

Document Version 1.5

Last Updated 25 Feb 2020

Links

Entity

<https://mappedu.ed-fi.org/#/entity/5987fca4-4ed2-e811-80d9-005056b36109/info?dataStandardId=db3697f4-4dd2-e811-80d9-005056b36109>

Scenarios

<https://techdocs.ed-fi.org/display/EDFICERT/Student+Enrollment+%3E+Graduation+Plan+Scenarios>

Schema

<https://schema.ed-fi.org/datahandbook-v22/Ed-Fi-UDM-Handbook-Index.html#/GraduationPlan562>

Root Table

REQ

SQL

Scheduled

REQ_Functions.GetEdFiDataTable_GraduationPlan

```
DECLARE @intYear smallint;
SET @intYear = CONVERT(smallint, (SELECT TOP 1 Year(DT) [YR] FROM [DAY] WHERE [DEL] = 0 ORDER BY [DT]
DESC));
WITH cteGradREQ AS (
SELECT LTRIM(RTRIM([REQ].[GRT])) [GRT], [REQ].[CD], [REQ].[SA], [REQ].[C1] AS Credits, (@intYear + 0)
GraduationYear, SUM(C1)
OVER (PARTITION BY [REQ].[GRT], [REQ].[SC]) TotalCredits, [REQ].[DEL], [REQ].[SC], [REQ].[DTS]
FROM [REQ]
WHERE [REQ].[SC] = @SC
UNION
SELECT LTRIM(RTRIM([REQ].[GRT])) [GRT], [REQ].[CD], [REQ].[SA], [REQ].[C2] AS Credits, (@intYear + 1)
GraduationYear, SUM(C2)
OVER (PARTITION BY [REQ].[GRT], [REQ].[SC]) TotalCredits, [REQ].[DEL], [REQ].[SC], [REQ].[DTS]
FROM [REQ]
WHERE [REQ].[SC] = @SC
UNION
SELECT LTRIM(RTRIM([REQ].[GRT])) [GRT], [REQ].[CD], [REQ].[SA], [REQ].[C3] AS Credits, (@intYear + 2)
GraduationYear, SUM(C3)
OVER (PARTITION BY [REQ].[GRT], [REQ].[SC]) TotalCredits, [REQ].[DEL], [REQ].[SC], [REQ].[DTS]
FROM [REQ]
WHERE [REQ].[SC] = @SC
UNION
SELECT LTRIM(RTRIM([REQ].[GRT])) [GRT], [REQ].[CD], [REQ].[SA], [REQ].[C4] AS Credits, (@intYear + 3)
GraduationYear, SUM(C4)
OVER (PARTITION BY [REQ].[GRT], [REQ].[SC]) TotalCredits, [REQ].[DEL], [REQ].[SC], [REQ].[DTS]
FROM [REQ]
WHERE [REQ].[SC] = @SC
)
SELECT [SC], [GRT], [CD], [SA], Credits, GraduationYear, TotalCredits
FROM cteGradREQ
WHERE [DEL] = 0
ORDER BY [SC], GraduationYear, [GRT], [CD]
```

Real-Time

REQ_Functions.GetEdFiDataTable_GraduationPlan

```

DECLARE @intYear smallint;
SET @intYear = CONVERT(smallint, (SELECT TOP 1 Year(DT) [YR] FROM [DAY] WHERE [DEL] = 0 ORDER BY [DT]
DESC));
WITH cteGradREQ AS (
    SELECT LTRIM(RTRIM([REQ].[GRT])) [GRT], [REQ].[CD], [REQ].[SA], [REQ].[C1] AS Credits, (@intYear + 0)
    GraduationYear, SUM(C1)
    OVER (PARTITION BY [REQ].[GRT], [REQ].[SC]) TotalCredits, [REQ].[DEL], [REQ].[SC], [REQ].[DTS]
    FROM [REQ]
    WHERE [REQ].[SC] = @SC
    UNION
    SELECT LTRIM(RTRIM([REQ].[GRT])) [GRT], [REQ].[CD], [REQ].[SA], [REQ].[C2] AS Credits, (@intYear + 1)
    GraduationYear, SUM(C2)
    OVER (PARTITION BY [REQ].[GRT], [REQ].[SC]) TotalCredits, [REQ].[DEL], [REQ].[SC], [REQ].[DTS]
    FROM [REQ]
    WHERE [REQ].[SC] = @SC AND [REQ].[DTS] >= @TimeSince
    UNION
    SELECT LTRIM(RTRIM([REQ].[GRT])) [GRT], [REQ].[CD], [REQ].[SA], [REQ].[C3] AS Credits, (@intYear + 2)
    GraduationYear, SUM(C3)
    OVER (PARTITION BY [REQ].[GRT], [REQ].[SC]) TotalCredits, [REQ].[DEL], [REQ].[SC], [REQ].[DTS]
    FROM [REQ]
    WHERE [REQ].[SC] = @SC AND [REQ].[DTS] >= @TimeSince
    UNION
    SELECT LTRIM(RTRIM([REQ].[GRT])) [GRT], [REQ].[CD], [REQ].[SA], [REQ].[C4] AS Credits, (@intYear + 3)
    GraduationYear, SUM(C4)
    OVER (PARTITION BY [REQ].[GRT], [REQ].[SC]) TotalCredits, [REQ].[DEL], [REQ].[SC], [REQ].[DTS]
    FROM [REQ]
    WHERE [REQ].[SC] = @SC AND [REQ].[DTS] >= @TimeSince
)
SELECT [SC], [GRT], [CD], [SA], Credits, GraduationYear, TotalCredits
FROM cteGradREQ
WHERE [DEL] = 0
ORDER BY [SC], GraduationYear, [GRT], [CD]

```

Entity Mapping

Parent	Property	Type	Required	Table	Field	Other	Code / Notes
\	creditsByCourses	List(Of GraduationPlan CreditsByCourse)	Optional				
creditsByCourses	courses	List(Of GraduationPlan CreditsByCourseCourse)	Optional				
creditsByCourses\courses	courseReference	CourseReference	Optional				
creditsByCourses\courses\courseReference	code	String	Optional				
creditsByCourses\courses\courseReference	educationOrganizationId	Integer	Optional				
creditsByCourses\courses\courseReference	link	Link	Optional				
creditsByCourses	courseSetName	String	Optional				
creditsByCourses	creditConversion	Double	Optional				
creditsByCourses	credits	Double	Optional				
creditsByCourses	creditType	String	Optional				
creditsByCourses	whenTakenGradeLevel Descriptor	String	Optional				
\	creditsBySubjects	List(Of GraduationPlan CreditsBySubject)	Optional				
creditsBySubjects	academicSubject Descriptor	AcademicSubject MapType	Optional	REQ	CD	CRS.S1 Code Values	GetEdFiCodeValue(AcademicSubjectDescriptor "CRS", "S1", row!CD)
creditsBySubjects	creditConversion	Double	Optional				
creditsBySubjects	credits	Double	Optional	REQ	Credits	C1-C4	GetNullableDouble(row!Credits)

creditsBySubjects	creditType	String	Optional				
\	educationOrganizationReference	EducationOrganizationReference	Identity				
educationOrganizationReference	educationOrganizationId	Integer	Identity	REQ	SC		GetNullableInteger(s("SC"))
educationOrganizationReference	link	Link	Optional				
\	graduationSchoolYearTypeReference	SchoolYearTypeReference	Identity				
graduationSchoolYearTypeReference	link	Link	Optional				
graduationSchoolYearTypeReference	schoolYear	Integer	Identity	REQ	GraduationYear		GetNullableDouble(s("GraduationYear"))
\	individualPlan	Boolean	Optional			Hard coded to False	GetNullableBoolean(False)
\	requiredAssessments	List(Of GraduationPlanRequiredAssessment)	Optional				
requiredAssessments	assessmentPerformanceLevel	GraduationPlanRequiredAssessmentAssessmentPerformanceLevel	Optional				
requiredAssessments\assessmentPerformanceLevel	assessmentReportingMethodType	String	Optional				
requiredAssessments\assessmentPerformanceLevel	maximumScore	String	Optional				
requiredAssessments\assessmentPerformanceLevel	minimumScore	String	Optional				
requiredAssessments\assessmentPerformanceLevel	performanceLevelDescriptor	String	Optional				
requiredAssessments\assessmentPerformanceLevel	resultDatatypeType	String	Optional				
requiredAssessments	assessmentReference	AssessmentReference	Optional				
requiredAssessments\assessmentReference	identifier	String	Optional				
requiredAssessments\assessmentReference	link	Link	Optional				
requiredAssessments\assessmentReference	namespace	String	Optional				
requiredAssessments	scores	List(Of GraduationPlanRequiredAssessmentScore)	Optional				
requiredAssessments\scores	assessmentReportingMethodType	String	Optional				
requiredAssessments\scores	maximumScore	String	Optional				
requiredAssessments\scores	minimumScore	String	Optional				
requiredAssessments\scores	resultDatatypeType	String	Optional				
\	totalRequiredCreditConversion	Double	Optional				
\	totalRequiredCredits	Double	Optional	REQ	TotalCredits		GetNullableDouble(s("TotalCredits"))
\	totalRequiredCreditType	Double	Optional				
\	typeDescriptor	GraduationPlanTypeDescriptor GraduationPlanTypeMapType	Identity	REQ	GRT		GetEdFiCodeValue(GraduationPlanTypeDescriptor, "REQ", "GRT", rowREQ!GRT)