

Ed-Fi 2.x Staff Section Association 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/90cbf4aa-4ed2-e811-80d9-005056b36109/info?dataStandardId=db3697f4-4dd2-e811-80d9-005056b36109>

Scenarios

<https://techdocs.ed-fi.org/display/EDFICERT/StaffAssociation+%3E+StaffSectionAssociation+Scenarios>

Schema

<https://schema.ed-fi.org/datahandbook-v22/Ed-Fi-UDM-Handbook-Index.html#/StaffSectionAssociation604>

Root Table

TCH

Join Table

MST, TRM

SQL

Scheduled

TCH_Functions.GetEdFiData_StaffSectionAssociations

```
;with cteAllTheData AS
(
    SELECT [MST].[SC] AS [SC], [MST].[SE] AS [SE], [MST].[TN], [MST].[TN2], [MST].[TN3], [TCH].[ID], [TCH].[ID2], [TCH].[ID3]
    , [MST].[SM] AS [SM], [MST].[RM] AS [RM], [MST].[PD] AS [PD], [MST].[BL] AS [BL], [MST].[DY] AS [DY], [MST].[CN] AS [CN], [TRM].[D1] AS [D1]
    FROM MST
    INNER JOIN [TRM] ON [MST].[SC] = [TRM].[SC] AND [MST].[SM] = [TRM].[TM]
    INNER JOIN [TCH] ON [TCH].[SC] = [MST].[SC] AND [TCH].[TN] = [MST].[TN]
    WHERE [MST].[DEL] = 0 AND [TCH].[DEL] = 0 AND [TCH].[TN] > 0 AND [MST].[SC] = @SC
)
, cteMST_TN AS
(
    SELECT [SC], [SE], [ID], [SM], [RM], [PD], [BL], [DY], [CN], [D1], 'Teacher of Record' classroomPosition
    FROM cteAllTheData
)
, cteMST_TN2 AS
(
    SELECT c.[SC], c.[SE], t.[ID], c.[SM], c.[RM], c.[PD], c.[BL], c.[DY], c.[CN], c.[D1], 'Assistant Teacher' classroomPosition
    FROM cteAllTheData c
    INNER JOIN [MST] m on c.[SC] = m.[SC] and c.[SE] = m.[SE] and c.[TN2] <> 0 and m.[SC] = @sc
    INNER JOIN [TCH] t on t.[SC] = m.[SC] and t.[TN] = m.[TN2]
)
, cteMST_TN3 AS
(
    SELECT c.[SC], c.[SE], t.[ID], c.[SM], c.[RM], c.[PD], c.[BL], c.[DY], c.[CN], c.[D1], 'Assistant Teacher' classroomPosition
    FROM cteAllTheData c
    INNER JOIN [MST] m on c.[SC] = m.[SC] and c.[SE] = m.[SE] and c.[TN3] <> 0 and m.[SC] = @sc
    INNER JOIN [TCH] t on t.[SC] = m.[SC] and t.[TN] = m.[TN3]
)
, cteTCH_ID2 AS
(
    SELECT c.[SC], c.[SE], c.[ID2] [ID], c.[SM], c.[RM], c.[PD], c.[BL], c.[DY], c.[CN], c.[D1], 'Support Teacher' classroomPosition
    FROM cteAllTheData c
    INNER JOIN [MST] m on c.[SC] = m.[SC] and c.[SE] = m.[SE] and c.[ID2] <> 0 and m.[SC] = @sc
)
, cteTCH_ID3 AS
(
    SELECT c.[SC], c.[SE], c.[ID3] [ID], c.[SM], c.[RM], c.[PD], c.[BL], c.[DY], c.[CN], c.[D1], 'Support Teacher' classroomPosition
    FROM cteAllTheData c
    INNER JOIN [MST] m on c.[SC] = m.[SC] and c.[SE] = m.[SE] and c.[ID3] <> 0 and m.[SC] = @sc
)
SELECT * FROM
(
    SELECT * FROM cteMST_TN
    UNION
    SELECT * FROM cteMST_TN2
    UNION
    SELECT * FROM cteMST_TN3
    UNION
    SELECT * FROM cteTCH_ID2
    UNION
    SELECT * FROM cteTCH_ID3
) x
ORDER BY x.[SC], x.[SE]
```

Real-Time

TCH_Functions.GetEdFiData_StaffSectionAssociations

```
;with cteAllTheData AS
(
    SELECT [MST].[SC] AS [SC], [MST].[SE] AS [SE], [MST].[TN], [MST].[TN2], [MST].[TN3], [TCH].[ID], [TCH].[ID2], [TCH].[ID3]
    , [MST].[SM] AS [SM], [MST].[RM] AS [RM], [MST].[PD] AS [PD], [MST].[BL] AS [BL], [MST].[DY] AS [DY], [MST].[CN] AS [CN], [TRM].[D1] AS [D1]
    FROM MST
    INNER JOIN [TRM] ON [MST].[SC] = [TRM].[SC] AND [MST].[SM] = [TRM].[TM]
    INNER JOIN [TCH] ON [TCH].[SC] = [MST].[SC] AND [TCH].[TN] = [MST].[TN]
    WHERE [MST].[DEL] = 0 AND [TCH].[DEL] = 0 AND [TCH].[TN] > 0 AND [MST].[SC] = @SC
    AND ([MST].[DTS] >= @TimeSince OR [TRM].[DTS] >= @TimeSince)
)
, cteMST_TN AS
(
    SELECT [SC], [SE], [ID], [SM], [RM], [PD], [BL], [DY], [CN], [D1], 'Teacher of Record' classroomPosition
    FROM cteAllTheData
)
, cteMST_TN2 AS
(
    SELECT c.[SC], c.[SE], t.[ID], c.[SM], c.[RM], c.[PD], c.[BL], c.[DY], c.[CN], c.[D1], 'Assistant Teacher' classroomPosition
    FROM cteAllTheData c
    INNER JOIN [MST] m on c.[SC] = m.[SC] and c.[SE] = m.[SE] and c.[TN2] <> 0 and m.[SC] = @sc
    INNER JOIN [TCH] t on t.[SC] = m.[SC] and t.[TN] = m.[TN2]
)
, cteMST_TN3 AS
(
    SELECT c.[SC], c.[SE], t.[ID], c.[SM], c.[RM], c.[PD], c.[BL], c.[DY], c.[CN], c.[D1], 'Assistant Teacher' classroomPosition
    FROM cteAllTheData c
    INNER JOIN [MST] m on c.[SC] = m.[SC] and c.[SE] = m.[SE] and c.[TN3] <> 0 and m.[SC] = @sc
    INNER JOIN [TCH] t on t.[SC] = m.[SC] and t.[TN] = m.[TN3]
)
, cteTCH_ID2 AS
(
    SELECT c.[SC], c.[SE], c.[ID2] [ID], c.[SM], c.[RM], c.[PD], c.[BL], c.[DY], c.[CN], c.[D1], 'Support Teacher' classroomPosition
    FROM cteAllTheData c
    INNER JOIN [MST] m on c.[SC] = m.[SC] and c.[SE] = m.[SE] and c.[ID2] <> 0 and m.[SC] = @sc
)
, cteTCH_ID3 AS
(
    SELECT c.[SC], c.[SE], c.[ID3] [ID], c.[SM], c.[RM], c.[PD], c.[BL], c.[DY], c.[CN], c.[D1], 'Support Teacher' classroomPosition
    FROM cteAllTheData c
    INNER JOIN [MST] m on c.[SC] = m.[SC] and c.[SE] = m.[SE] and c.[ID3] <> 0 and m.[SC] = @sc
)
SELECT * FROM
(
    SELECT * FROM cteMST_TN
    UNION
    SELECT * FROM cteMST_TN2
    UNION
    SELECT * FROM cteMST_TN3
    UNION
    SELECT * FROM cteTCH_ID2
    UNION
    SELECT * FROM cteTCH_ID3
) x
ORDER BY x.[SC], x.[SE]
```

Mapping

Parent	Property	Type	Required	Table	Field	Other	Code / Notes
\	beginDate	Date	Required?				Scenario says it's required but we're not passing anything and it's not failing so maybe it's not actually required
	classroomPositionDescriptor	ClassroomPositionMapType	Required				classroomPosition
\	endDate	Date	Optional				
\	highlyQualifiedTeacher	Boolean	Optional				
\	percentageContribution	Double	Optional				
\	sectionReference	SectionReference	Identity				
sectionReference	classPeriodName	String	Identity	MST	PD, BL, DY		GetClassPeriodName(rowTCH!PD, rowTCH!BL, rowTCH!DY) see function below
	classroomIdentificationCode	String	Identity	MST	RM		GetClassroomIdentificationCode(s("RM")) see function below
sectionReference	link	Link	Optional				
sectionReference	localCourseCode	String	Identity	MST	CN		GetString(s("CN"))
sectionReference	schoolId	Integer	Identity	MST	SC		GetNullableInteger(s("SC"))
sectionReference	schoolYear	Integer	Identity	TRM	D1		GetSchoolYear(s("D1")) see function below
sectionReference	sequenceOfCourse	Integer	Identity			hard code to 1	GetNullableInteger(1)
sectionReference	termDescriptor	TermMapType	Identity	MST	SC, SM		GetTermDescriptor(rowTCH!SC, rowTCH!SM)
sectionReference	uniqueSectionCode	String	Identity	MST	SC, SE		GetUniqueSectionCode(s("SC"), s("SE")) see function below
\	staffReference	StaffReference	Optional				
staffReference	link	Link	Optional				
staffReference	staffUniqueld	String	Identity	TCH	ID		GetString(s("ID"))
\	teacherStudentDataLinkExclusion	Boolean	Optional				

Code

```

Public Shared Function GetClassPeriodName(ByVal period As Integer, ByVal block As Integer, ByVal day As String) As String

    Return period & (IIf(block > 1, "-" & (period + block - 1), "")) & " " & day 'Examples: "1 MTWTF", "2-3 MTWTF", "5 MW", "6 A-DAYS"

End Function

Public Shared Function GetClassroomIdentificationCode(ByVal room As String) As String

    Return IIf(String.IsNullOrEmpty(room), "Unassigned", room)

End Function

Public Shared Function GetSchoolYear(ByVal SchoolDate As String) As Integer?

    Dim schoolYear As Integer? = Nothing
    Dim schoolDateValue As Date

    If Date.TryParse(SchoolDate, schoolDateValue) Then
        schoolYear = EdFiV2Functions.GetSchoolYearFromDate(schoolDateValue)
    End If

    Return schoolYear

Public Shared Function GetSchoolYearFromDate(ByVal SchoolDate As Date) As Integer

```

```

    ' Ed-Fi sometimes uses a single year as 'school year'. It's actually, just the end year of a full school
    year in the form YYYY-YYYY
    Return GetEndYear(SchoolDate)

End Function

Public Shared Function GetEndYear(ByVal startDate As Date) As Integer

    Dim endYear As Integer = Nothing
    Dim startMonth As Integer = Month(startDate)

    If startMonth >= 7 Then          ' July-Dec = "this year" (i.e. 9/1/2012 = 2012-2013, so 2013)
        endYear = Year(startDate) + 1
    Else ' startMonth < 7          ' Jan-June = "last year" (i.e. 3/1/2012 = 2011-2012, so 2012)
        endYear = Year(startDate)
    End If

    Return endYear

End Function

Public Shared Function GetTermDescriptor(ByVal SC As Integer, ByVal TM As String, ByVal ConnectionString As
String) As String

    Select Case TM
        Case "Y"
            Return "Year Round"
        Case "S"
            Return "Spring Semester"
        Case "F"
            Return "Fall Semester"
    End Select

    If Trimester(SC, ConnectionString) Then
        Select Case TM
            Case "1"
                Return "First Trimester"
            Case "2"
                Return "Second Trimester"
            Case "3"
                Return "Third Trimester"
        End Select
    Else
        Select Case TM
            Case "1"
                Return "First Quarter"
            Case "2"
                Return "Second Quarter"
            Case "3"
                Return "Third Quarter"
            Case "4"
                Return "Fourth Quarter"
        End Select
    End If

    Return TM

End Function

Public Shared Function GetUniqueSectionCode(ByVal school As String, ByVal section As String) As String

    Return school & "-" & section

End Function

```