

Ed-Fi 3.x Student School Attendance Event Entity Mapping

Overview

This document defines the mapping of Aeries data to the Ed-Fi data standard.



Document Version 1.0

Last Updated 07 Nov 2021

Status

Ready for v3 certification.

Description

This event entity represents the recording of whether a student is in attendance for a school day.

Student School Attendance Event Links

Mapping v3.2.0-c

<https://mappingedu.ed-fi.org/#!/entity/b0076848-e7ba-ea11-80df-005056b36109/info?dataStandardId=7ec6133d-e7ba-ea11-80df-005056b36109>

Schema v3.2.0-c

<https://schema.ed-fi.org/datahandbook-v320c/#!/StudentSchoolAttendanceEvent593>

API v3.4.0

<https://api.ed-fi.org/v3.4.0/docs/index.html#/studentSchoolAttendanceEvents>

Scenarios v3

<https://techdocs.ed-fi.org/display/EDFICERT/v3+Student+Attendance+%3E+StudentSchoolAttendanceEvent+Scenarios>

Natural Key

- AttendanceEventCategoryDescriptor
- EventDate
- SchoolId
- SchoolYear
- SessionName
- StudentUniqueId

Used By

- None

Entity Mapping

We get a list of these items from a single row from the ATT table - We create one item for each of the AL, A0-A9 column values, when any are found.

Parent	Property	Type	Required	Table	Field	Other	Code / Notes
\	eventDate	Date	Identity	ATT	DT		GetNullableDate(rowATT!DT)
\	schoolReference	SchoolReference	Identity				
schoolReference	schoolId	Integer	Identity	ATT	SC		GetNullableInteger(s("SC"))
\	sessionReference	SessionReference	Identity				
sessionReference	schoolId	Integer	Identity	ATT	SC		GetNullableInteger(s("SC"))
sessionReference	schoolYear	Integer	Identity	ATT	SchoolYear		GetNullableInteger(s("SchoolYear")) GetCurrentYear() + 1 (see below)

sessionReference	sessionName	string	Identity	TRM	SC, DT		termName = GetTermDescriptorForEvent (rowATT!SC, rowATT!DT) GetSessionName(rowATT! SchoolYear, termName)
\	studentReference	StudentReference	Identity				
studentReference	studentUniqueld	string	Identity	ATT	SC, SN		STU_Functions. ConvertSCSNtoID(rowATT!SC, rowATT!SN)
\	attendanceEvent CategoryDescriptor	Descriptor	Identity	ATT	AL		GetEdFiCodeValue(AttendanceEvent CategoryDescriptor, "ATT", "AL", rowATT!AL)
\	educationalEnvironment Descriptor	Descriptor	Optional				
\	attendanceEvent Reason	String	Required	ATN	CO		GetString(rowATT!AttDesc))
\	eventDuration	number	Optional				
\	arrivalTime	String	Optional				
\	departureTime	String	Optional				
\	schoolAttendanceDuration	number	Optional				

Code

```

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

    Dim CurrentYear As Integer = 0

    Using drDAY As SqlDataReader =
        ConnectDB.GetDataReader("SELECT TOP 1 CASE WHEN SC = @SC THEN
'1' ELSE '2' END as [SORT], DT, DY
        FROM DAY WHERE DEL = 0 ORDER BY SORT, DY", ConnectionString,
New SqlParameter() {MakeParam("@SC", SqlDbType.Int, 0, SC)})

        If drDAY IsNot Nothing AndAlso drDAY.HasRows Then
            drDAY.Read()
            CurrentYear = Year(drDAY("DT"))
        End If
    End Using

    If CurrentYear = 0 Then
        CurrentYear = ConnectDB.GetSchoolYearFromConnection
(ConnectionString)
    End If

    Return CurrentYear

End Function

Public Shared Function GetTermDescriptorForEvent(ByVal schoolcode As
String, ByVal eventDate As String, ByVal connectionString As String) As

```

String

```
Dim termDescriptor As String = String.Empty
Dim eventDateValue As Date

If Date.TryParse(eventDate, eventDateValue) Then
    Dim term As String = TRM_Functions.GetTerm(schoolcode,
eventDateValue, String.Empty, connectionString)
    termDescriptor = EdFiv2Functions.GetTermDescriptor
(schoolcode, term, connectionString)
End If

Return termDescriptor

End Function

Public Shared Function GetTerm(ByVal SC As Integer, ByVal eventDate As
Date, ByVal USR As String, ByVal ConnectionString As String) As String

    Dim ix1 As Integer = 1
    Dim rcdTRM As TRM_Record

    GetTerm = ""

    If Trimester(SC, ConnectionString) Then
        For ix1 = 1 To 4
            rcdTRM = New TRM_Record(USR, SC, ix1.ToString(),
ConnectionString)

            If Not rcdTRM.NewRecord Then
                Dim endDateTrimester As Date
                If Date.TryParse(rcdTRM!D2,
endDateTrimester) Then
                    If eventDate <=
endDateTrimester Then
                        GetTerm = rcdTRM.TM
                        GoTo CurrentTerm_exit
                    End If
                End If
            End If
        Next
    Else
        rcdTRM = New TRM_Record(USR, SC, "F", ConnectionString)
        If Not rcdTRM.NewRecord Then
            Dim endDateFall As Date
            If Date.TryParse(rcdTRM!D2, endDateFall) Then
                If eventDate <= endDateFall Then
                    GetTerm = rcdTRM.TM
                    GoTo CurrentTerm_exit
                End If
            End If
        End If
    End If
```

```

        End If
        rcdTRM = New TRM_Record(USR, SC, "S", ConnectionString)
        If Not rcdTRM.NewRecord Then
            Dim endDateSpring As Date
            If Date.TryParse(rcdTRM!D2, endDateSpring) Then
                If eventDate <= endDateSpring Then
                    GetTerm = rcdTRM.TM
                    GoTo CurrentTerm_exit
                End If
            End If
        End If
    End If

    rcdTRM = New TRM_Record(USR, SC, "Y", ConnectionString)
    If Not rcdTRM.NewRecord Then
        GetTerm = rcdTRM.TM
        GoTo CurrentTerm_exit
    End If

```

```

CurrentTerm_exit:
    rcdTRM = Nothing
    Exit Function

```

```

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