# Schema

Tables included in this schema:

* Appointment
* AppointmentParticipant
* Person
* EmailAddress
* PhoneNumber
* PhysicalAddress
* EmailParticipant
* Email
* EmailExtendedProperties

## Appointment

**Appointment**, which should accurately represent the properties found on an Event object in Office 365 (Event\_v1 in Microsoft Graph data connect)

* + This should include meetings both scheduled and attended by any users in the user scope
  + Omissions from the Graph schema: categories (String collection), location (location object), locations (collection of locations), recurrence (patternedRecurrence object), responseStatus (responseStatus object)
    - These are all omitted as they would require another table to represent
  + Missing the following from Customer Engagement Appointment schema: **ActualDurationMinutes**, AttachmentErrors, **Category**, ImportSequenceNumber, IsBilled, **IsDraft**, IsMapiPrivate, IsWorkflowCreated, LastOnHoldTime, Location, **OptionalAttendees**, OverridenCreatedOn, OwnerId, OwnerIdType, ProcessId, RegardingObjectId, RegardingObjectTypeCode, ScheduledDurationMinutes, ServiceId, SLAId, SortDate, StageId, StateCode, Subcategory, SubscriptionId, TranscationCurrencyId

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| **Body** | **String** | **Represents properties of the body of an item, such as a message, event or group post. Text format only** |
| **createdDateTime** | **DateTimeOffset** | **The Timestamp type represents date and time information using ISO 8601 format and is always in UTC time. For example, midnight UTC on Jan 1, 2014 would look like this: '2014-01-01T00:00:00Z'** |
| **End** | **DateTimeOffset** | **The Timestamp type represents date and time information using ISO 8601 format and is always in UTC time. For example, midnight UTC on Jan 1, 2014 would look like this: '2014-01-01T00:00:00Z'** |
| **hasAttachments** | **Boolean** | **Set to true if the event has attachments.** |
| **iCalUId** | **String** | **A unique identifier that is shared by all instances of an event across different calendars.** |
| **AppointmentId** | **String** | **Unique identifier for each meeting (including recurring meetings); primary key** |
| **Importance** | **String<enum>** | **The importance of the event. The possible values are: low, normal, high.** |
| **isAllDay** | **Boolean** | **Set to true if the event lasts all day.** |
| **isCancelled** | **Boolean** | **Set to true if the event has been canceled.** |
| **isReminderOn** | **Boolean** | **Set to true if an alert is set to remind the user of the event.** |
| **lastModifiedDateTime** | **DateTimeOffset** | **The Timestamp type represents date and time information using ISO 8601 format and is always in UTC time. For example, midnight UTC on Jan 1, 2014 would look like this: '2014-01-01T00:00:00Z'** |
| **onlineMeetingUrl** | **String** | **A URL for an online meeting. The property is set only when an organizer specifies an event as an online meeting such as a Skype meeting.** |
| **originalEndTimeZone** | **String** | **The end time zone that was set when the event was created. A value of tzone://Microsoft/Custom indicates that a legacy custom time zone was set in desktop Outlook.** |
| **originalStart** | **DateTimeOffset** | **The Timestamp type represents date and time information using ISO 8601 format and is always in UTC time. For example, midnight UTC on Jan 1, 2014 would look like this: '2014-01-01T00:00:00Z'** |
| **originalStartTimeZone** | **DateTimeOffset** | **The Timestamp type represents date and time information using ISO 8601 format and is always in UTC time. For example, midnight UTC on Jan 1, 2014 would look like this: '2014-01-01T00:00:00Z'** |
| **reminderMinutesBeforeStart** | **Int32** | **The number of minutes before the event start time that the reminder alert occurs.** |
| **responseRequested** | **Boolean** | **Set to true if the sender would like a response when the event is accepted or declined.** |
| **Sensitivity** | **String<enum>** | **The possible values are: normal, personal, private, confidential.** |
| **seriesMasterId** | **String** | **The ID for the recurring series master item, if this event is part of a recurring series.** |
| **showAs** | **String<enum>** | **The status to show. The possible values are: free, tentative, busy, oof, workingElsewhere, unknown.** |
| **Start** | **DateTimeOffset** | **The Timestamp type represents date and time information using ISO 8601 format and is always in UTC time. For example, midnight UTC on Jan 1, 2014 would look like this: '2014-01-01T00:00:00Z'** |
| **SourceOfOrigin** | **String** | **Can be Office365/CRM** |
| **Subject** | **String** | **The text of the event's subject line.** |
| **Type** | **String** | **The event type. The possible values are: singleInstance, occurrence, exception, seriesMaster.** |

## AppointmnetParticipant

**AppointmentParticipant**, which links meetings to the users invited/attended

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| **AppointmentId** | **String** | **Unique identifier for each meeting (including recurring meetings); foreign key matching Id in Meetings table** |
| **PersonId** | **String** | **Unique value for a participant any time an HR attribute changes; foreign key matching Id in Person table** |
| **IsOrganizer** | **Boolean** | **True if this participant organized the meeting** |
| **Response** | **String<enum>** | **Invitee's response to the meeting: declined, tentative, accepted, or no response** |

## Person

**Person**, which should accurately represent the AAD user object (combination of User\_v1 and Manager\_v0 data set in Microsoft Graph data connect).

This table contains all information about a user except for their email addresses, phone numbers and physical addresses, which are provided as separate tables as one user can have more than one of each.

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| **Birthday** | **DateTimeOffset** | **The birthday of the user. The Timestamp type represents date and time information using ISO 8601 format and is always in UTC time. For example, midnight UTC on Jan 1, 2014 would look like this: '2014-01-01T00:00:00Z'** |
| **companyName** | **String** | **The company name which the user is associated. This property can be useful for describing the company that an external user comes from. Can be null** |
| **createdDateTime** | **DateTimeOffset** | **The created date of the user object.** |
| **Department** | **String** | **The name for the department in which the user works.** |
| **displayName** | **String** | **The name displayed in the address book for the user. This is usually the combination of the user's first name, middle initial and last name. This property is required when a user is created and it cannot be cleared during updates** |
| **givenName** | **String** | **The given name (first name) of the user.** |
| **hireDate** | **DateTimeOffset** | **The hire date of the user. The Timestamp type represents date and time information using ISO 8601 format and is always in UTC time. For example, midnight UTC on Jan 1, 2014 would look like this: '2014-01-01T00:00:00Z'** |
| **PersonId** | **String** | **Unique value for every person for each HR change; primary key. Maps to the AAD User ID if the user is internal, the mail attribute if the user is external** |
| **jobTitle** | **String** | **The user’s job title.** |
| **ManagerId** | **String** | **Unique ID for each person’s manager. Null if the user doesn’t have a manager** |
| **preferredLanguage** | **String** | **The preferred language for the user. Should follow ISO 639-1 Code; for example "en-US".** |
| **preferredName** | **String** | **The preferred name for the user.** |
| **Surname** | **String** | **The user's surname (family name or last name).** |
| **usageLocation** | **String** | **A two letter country code (ISO standard 3166). Required for users that will be assigned licenses due to legal requirement to check for availability of services in countries. Examples include: "US", "JP", and "GB"** |
| **userPrincipalName** | **String** | **The user principal name (UPN) of the user. The UPN is an Internet-style login name for the user based on the Internet standard RFC 822. By convention, this should map to the user's email name. The general format is alias@domain, where domain must be present in the tenant’s collection of verified domains. This property is required when a user is created. The verified domains for the tenant can be accessed from the verifiedDomains property of** [**organization**](https://docs.microsoft.com/en-us/graph/api/resources/organization?view=graph-rest-1.0)**.** |
| **userType** | **String** | **A string value that can be used to classify user types in your directory, such as “Member” and “Guest”.** |

## EmailAddress

**EmailAddress**, which contains the email addresses for a Person from AAD or from the email or contact record, if the user is external to the Office 365 organization

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| PersonId | String | Unique identifer for every person; foreign key matching Id in Person table primary key |
| EmailAddress | String | The SMTP address for the user, for example, "jeff@contoso.onmicrosoft.com". Can be the user’s UPN or an alias. |

## PhoneNumber

**PhoneNumber**, which contains the phone numbers of a Person from AAD (if the are in the Office 365 organization) or from an Appointment or contact record (if the user is external to the organization)

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| PersonId | String | Unique identifier for every person; foreign key matching Id in Person table primary key |
| PhoneNumber | String | A phone number of the user. |
| PhoneNumberType | String<enum> | Can be “Business” (from User or Contact table), “Mobile” (from User or Contact), or “Home” (from Contact) |

## PhysicalAddress

**PhysicalAddress,** which contains the physical address information for a Person. This is populated either from AAD or from the contact records.

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| PersonId | String | Unique identifer for every person; foreign key matching Id in Person table primary key |
| **City** | **String** | **The city in which the user is located.** |
| **Country** | **String** | **The country/region in which the user is located; for example, “US” or “UK”.** |
| **officeLocation** | **String** | **The office location in the user's place of business.** |
| **postalCode** | **String** | **The postal code for the user's postal address. The postal code is specific to the user's country/region. In the United States of America, this attribute contains the ZIP code.** |
| **State** | **String** | **The state or province in the user's address.** |
| **streetAddress** | **String** | **The street address of the user's place of business.** |

## EmailParticipant

**EmailParticipant**, which links messages to the users who sent/received them

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| EmailId | String | Unique identifier for every email sent; foreign key matching Id in Mails table |
| PersonId | String | Unique identifer for every person; foreign key matching Id in Person table primary key |
| IsSender | Boolean | True if this person sent the email |
| recipientType | String<enum> | Type of recipient. Possible values are To, CC, or BCC. Null if the user is the sender. |

## Email

**Email**, which should accurately represent the properties found on a Message in Office 365

* This should include messages both sent to and from any user in the user scope
* Omits bccRecipients (recipient collection), ccRecipients (recipient collection), flag (followupFlag object), from (recipient object), internetMessageHeaders (internetMessageHeader collection), replyTo (recipient collection), sender (recipient object), toRecipients (recipient collection),
  + Mainly due to the requirement for another table or it’s in the EmailParticipant table (cc,bcc)
* Missing properties from the Email Customer Engagement schema: ActualDurationMinutes, ActualEnd, ActualStart, AttachmentOpenCount, BaseConversationIndex, **Category**, DelayedEmailSendTime, DeliveryAttempts, DeliveryPriorityCode, EmailReminderExpiryTime, EmailReminderType, ImportSequenceNumber, IsBilled, IsWorkflowCreated, LastOnHoldTime, LastOpenedTime, LinksClickedCount, MimeType, Notifications, OpenCount, OverridentCreatedOn, OwnerId, OwnerIdType, ParentActivityId, PriorityCode, ProcessId, RegardingObjectId, RegardingObjectTypeCode, ReminderActionCardId, ScheduleEnd, ScheduledStart, ServiceId, SLAId, SortDate, StageId, StateCode, StatusCode, Subcategory

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| **Body** | **String** | **Represents properties of the body of an item, such as a message, event or group post. Text format only** |
| **conversationId** | **String** | **The ID of the conversation the email belongs to.** |
| **createdDateTime** | **DateTimeOffset** | **The Timestamp type represents date and time information using ISO 8601 format and is always in UTC time. For example, midnight UTC on Jan 1, 2014 would look like this: '2014-01-01T00:00:00Z'** |
| **hasAttachments** | **Boolean** | **Set to true if the event has attachments.** |
| **EmailId** | **String** | **Unique identifier for each mail (including recurring meetings); primary key** |
| **Importance** | **String<enum>** | **The importance of the event. The possible values are: low, normal, high.** |
| **inferenceClassification** | **String<enum>** | **The classification of the message for the user, based on inferre d relevance or importance, or on an explicit override. The possible values are: focused or other.** |
| **internetMessageId** | **String** | **The message ID in the format specified by** [**RFC2822**](https://www.ietf.org/rfc/rfc2822.txt)**.** |
| **isDeliveryReceiptRequested** | **Boolean** | **Indicates whether a read receipt is requested for the message.** |
| **isDraft** | **Boolean** | **Indicates whether the message is a draft. A message is a draft if it hasn't been sent yet.** |
| **isRead** | **Boolean** | **Indicates whether the message has been read.** |
| **isReadReceiptRequested** | **Boolean** | **Indicates whether a read receipt is requested for the message.** |
| **lastModifiedDateTime** | **DateTimeOffset** | **The Timestamp type represents date and time information using ISO 8601 format and is always in UTC time. For example, midnight UTC on Jan 1, 2014 would look like this: '2014-01-01T00:00:00Z'** |
| **parentFolderId** | **String** | **The unique identifier for the message’s parent mailFolder.** |
| **receivedDateTime** | **DateTimeOffset** | **The date and time the message was received.** |
| **sentDateTime** | **DateTimeOffset** | **The date and time the message was sent.** |
| **Subject** | **String** | **The subject of the message.** |
| **SourceOfOrigin** | **String** | **Can be Office365/CRM/** |
| **webLink** | **String** | **The URL to open the message in Outlook Web App.  You can append an ispopout argument to the end of the URL to change how the message is displayed. If ispopout is not present or if it is set to 1, then the message is shown in a popout window. If ispopout is set to 0, then the browser will show the message in the Outlook Web App review pane.  The message will open in the browser if you are logged in to your mailbox via Outlook Web App. You will be prompted to login if you are not already logged in with the browser.  This URL can be accessed from within an iFrame.** |

## EmailExtendedProperties

CDM entity with SingleValueExtended properties extracted as part of emails. A collection of single-value extended properties. This is a navigation property.

|  |  |  |
| --- | --- | --- |
| Property | Type | Description |
| **EmailId** | **String** | **Unique identifier for each mail (including recurring meetings); primary key** |
| **PropertyId** | **String** | **Id of the SingleValueProperty** |
| **Type** | **String** | **Type of the SVP** |
| **Value** | **String** | **Value of the SVP** |