



OData Vocabularies Version 4.0

Committee Specification Draft 02

19 June 2024

This stage:

<https://docs.oasis-open.org/odata/odata-vocabularies/v4.0/csd02/odata-vocabularies-v4.0-csd02.md> (Authoritative)
<https://docs.oasis-open.org/odata/odata-vocabularies/v4.0/csd02/odata-vocabularies-v4.0-csd02.html>
<https://docs.oasis-open.org/odata/odata-vocabularies/v4.0/csd02/odata-vocabularies-v4.0-csd02.pdf>

Previous stage:

<https://docs.oasis-open.org/odata/odata-vocabularies/v4.0/csprd01/odata-vocabularies-v4.0-csprd01.docx>
(Authoritative)
<https://docs.oasis-open.org/odata/odata-vocabularies/v4.0/csprd01/odata-vocabularies-v4.0-csprd01.html>
<https://docs.oasis-open.org/odata/odata-vocabularies/v4.0/csprd01/odata-vocabularies-v4.0-csprd01.pdf>

Latest stage:

<https://docs.oasis-open.org/odata/odata-vocabularies/v4.0/odata-vocabularies-v4.0.docx> (Authoritative)
<https://docs.oasis-open.org/odata/odata-vocabularies/v4.0/odata-vocabularies-v4.0.html>
<https://docs.oasis-open.org/odata/odata-vocabularies/v4.0/odata-vocabularies-v4.0.pdf>

Technical Committee:

[OASIS Open Data Protocol \(OData\) TC](#)

Chairs:

Ralf Handl (ralf.handl@sap.com), [SAP SE](#)
Michael Pizzo (mikep@microsoft.com), [Microsoft](#)

Editors:

Ralf Handl (ralf.handl@sap.com), [SAP SE](#)
Michael Pizzo (mikep@microsoft.com), [Microsoft](#)
Heiko Theißen (heiko.theissen@sap.com), [SAP SE](#)

Related work:

This specification is related to:

- *OData Version 4.02*. Edited by Michael Pizzo, Ralf Handl, and Heiko Theißen. A multi-part Work Product that includes:
 - *OData Version 4.02 Part 1: Protocol*. Latest stage. <https://docs.oasis-open.org/odata/odata/v4.02/odata-v4.02-part1-protocol.html>
 - *OData Version 4.02 Part 2: URL Conventions*. Latest stage. <https://docs.oasis-open.org/odata/odata/v4.02/odata-v4.02-part2-url-conventions.html>

- *ABNF components: OData ABNF Construction Rules Version 4.02 and OData ABNF Test Cases.*
<https://docs.oasis-open.org/odata/odata/v4.02/csd01/abnf/>
- *OData Common Schema Definition Language (CSDL) JSON Representation Version 4.02.* Edited by Michael Pizzo, Ralf Handl, and Heiko Theißen. Latest stage: <https://docs.oasis-open.org/odata/odata-csdl-json/v4.02/odata-csdl-json-v4.02.html>
- *OData Common Schema Definition Language (CSDL) XML Representation Version 4.02.* Edited by Michael Pizzo, Ralf Handl, and Heiko Theißen. Latest stage: <https://docs.oasis-open.org/odata/odata-csdl-xml/v4.02/odata-csdl-xml-v4.02.html>
- *OData JSON Format Version 4.02.* Edited by Michael Pizzo, Ralf Handl, and Heiko Theißen. Latest stage: <https://docs.oasis-open.org/odata/odata-json-format/v4.02/odata-json-format-v4.02.html>
- *OData Data Aggregation Extension Version 4.0.* Edited by Ralf Handl, Hubert Heijkers, Gerald Krause, Michael Pizzo, Heiko Theißen, and Martin Zurmuehl. Latest stage: <https://docs.oasis-open.org/odata/odata-data-aggregation-ext/v4.0/odata-data-aggregation-ext-v4.0.html>
- *OData Extension for Temporal Data Version 4.0.* Edited by Ralf Handl, Hubert Heijkers, Gerald Krause, Michael Pizzo, Heiko Theißen, and Martin Zurmuehl. Latest stage: <https://docs.oasis-open.org/odata/odata-temporal-ext/v4.0/odata-temporal-ext-v4.0.html>

Abstract:

This prose specification describes a set of OData vocabularies maintained by the OASIS OData Technical Committee. These vocabulary components are continuously evolved. The latest work-in-progress revisions are available at the odata-vocabularies OASIS TC GitHub repository: <https://github.com/oasis-tcs/odata-vocabularies>.

Status:

This document was last revised or approved by the OASIS Open Data Protocol (OData) TC on the above date. The level of approval is also listed above. Check the “Latest stage” location noted above for possible later revisions of this document. Any other numbered Versions and other technical work produced by the Technical Committee (TC) are listed at <https://groups.oasis-open.org/communities/tc-community-home2?CommunityKey=e7cac2a9-2d18-4640-b94d-018dc7d3f0e2#technical>.

TC members should send comments on this specification to the TC’s email list. Any individual may submit comments to the TC by sending email to Technical-Committee-Comments@oasis-open.org. Please use a Subject line like “Comment on OData Vocabularies”. Note for TC Editors: we expect to have the permanent TC-specific comment lists properly available in the future, so this clause will change at that time.

This specification is provided under the [RF on RAND Terms Mode](#) of the [OASIS IPR Policy](#), the mode chosen when the Technical Committee was established. For information on whether any patents have been disclosed that may be essential to implementing this specification, and any offers of patent licensing terms, please refer to the Intellectual Property Rights section of the TC’s web page (<https://www.oasis-open.org/committees/odata/ipr.php>).

Note that any machine-readable content ([Computer Language Definitions](#)) declared Normative for this Work Product is provided in separate plain text files. In the event of a discrepancy between any such plain text file and display content in the Work Product’s prose narrative document(s), the content in the separate plain text file prevails.

Key words:

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “NOT RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in BCP 14 [\[RFC2119\]](#) and [\[RFC8174\]](#) when, and only when, they appear in all capitals, as shown here.

Citation format:

When referencing this specification the following citation format should be used:

[OData-Vocabularies-v4.0]

OData Vocabularies Version 4.0. Edited by Ralf Handl, Michael Pizzo, and Heiko Theißen. 19 June 2024. OASIS Committee Specification Draft 02. <https://docs.oasis-open.org/odata/odata-vocabularies/v4.0/csd02/odata-vocabularies-v4.0-csd02.html>. Latest stage: <https://docs.oasis-open.org/odata/odata-vocabularies/v4.0/odata-vocabularies-v4.0.html>.

Notices

Copyright © OASIS Open 2024. All Rights Reserved.

Distributed under the terms of the OASIS [IPR Policy](#).

The name “OASIS” is a trademark of [OASIS](#), the owner and developer of this specification, and should be used only to refer to the organization and its official outputs.

For complete copyright information please see the full Notices section in an Appendix below.

Table of Contents

1 Introduction	
2 Core Vocabulary	
3 Capabilities Vocabulary	
4 Measures Vocabulary	
5 Validation Vocabulary	
6 Aggregation Vocabulary	
7 Authorization Vocabulary	
8 JSON Vocabulary	
9 Repeatability Vocabulary	
10 Temporal Vocabulary	
A References	
A.1 Normative References	
A.2 Non-Normative References	
B Acknowledgments	
B.1 Special Thanks	
B.2 Participants	
C Revision History	
D Notices	

1 Introduction

OData provides the ability to decorate model elements of a schema and instances of data with descriptive elements called annotations. Annotations can be used to specify additional facts about an element, such as whether it is read-only, or to define a common concept, such as a person or a movie. Annotations consist of a term (the namespace-qualified name of the annotation being applied), a target (the element to which the term is applied), and a value. A set of related annotation terms with a common namespace comprises a Vocabulary. Shared vocabularies provide a powerful extensibility point for OData. Refer to the [\[OData-Protocol\]](#) and [\[OData-CSDL\]](#) specifications for more information on the use of vocabularies.

This prose specification describes a set of OData vocabularies maintained by the OASIS OData Technical Committee. It is one component of a Work Product that also includes the various vocabulary components. These vocabulary components are continuously evolved, the latest work-in-progress revisions are available at [\[OData-Vocabs\]](#).

Other OData vocabularies may be created, shared, and maintained outside of this work product.

2 Core Vocabulary

Namespace: [Org.OData.Core.V1](#)

Core terms needed to write vocabularies

Terms

Term	Type	Description
ODataVersions	String	A space-separated list of supported versions of the OData Protocol. Note that 4.0 is implied by 4.01 and does not need to be separately listed.
SchemaVersion	String	Service-defined value representing the version of the schema. Services MAY use semantic versioning, but clients MUST NOT assume this is the case.
Revisions	[RevisionType]	List of revisions of a model element
Description	String?	A brief description of a model element
LongDescription	String?	A long description of a model element
Links	[Link]	Link to related information
Example	ExampleValue	Example for an instance of the annotated model element (Example)
Messages	[MessageType]	Instance annotation for warning and info messages
ValueException	ValueExceptionType	The annotated value is problematic
ResourceException	ResourceExceptionType	The annotated instance within a success payload is problematic
DataModificationException	DataModificationExceptionType	A modification operation failed on the annotated instance or collection within a success payload
IsLanguageDependent	Tag	Properties and terms annotated with this term are language-dependent
RequiresType	String	Terms annotated with this term can only be applied to elements that have a type that is identical to or derived from the given type name

Term	Type	Description
AppliesViaContainer	Tag	The target path of an annotation with the tagged term MUST start with an entity container or the annotation MUST be embedded within an entity container, entity set or singleton Services MAY additionally annotate a container-independent model element (entity type, property, navigation property) if allowed by the <code>AppliesTo</code> property of the term and the annotation applies to all uses of that model element.
ResourcePath	URL	Resource path for entity container child, can be relative to <code>xml:base</code> and the request URL
DereferenceableIDs	Tag	Entity-ids are URLs that locate the identified entity
ConventionalIDs	Tag	Entity-ids follow OData URL conventions
Permissions	Permission	Permissions for accessing a resource
ContentID	String	A unique identifier for nested entities within a request.
DefaultNamespace	Tag	Functions, actions and types in this namespace can be referenced in URLs with or without namespace- or alias- qualification. Data Modelers should ensure uniqueness of schema children across all default namespaces, and should avoid naming bound functions, actions, or derived types with the same name as a structural or navigational property of the type.
Immutable	Tag	A value for this non-key property can be provided by the client on insert and remains unchanged on update
Computed	Tag	A value for this property is generated on both insert and update
ComputedDefaultValue	Tag	A value for this property can be provided by the client on insert and update. If no value is provided on insert, a non-static default value is generated
IsURL	Tag	Properties and terms annotated with this term MUST contain a valid URL
AcceptableMediaTypes	[MediaType]	Lists the MIME types acceptable for the annotated entity type marked with <code>HasStream="true"</code> or the annotated binary, stream, or string property or term The annotation of a <code>TypeDefinition</code> propagates to the model elements having this type

Term	Type	Description
MediaType	MediaType?	The media type of the media stream of the annotated entity type marked with HasStream="true" or the annotated binary, stream, or string property or term The annotation of a TypeDefinition propagates to the model elements having this type
IsMediaType	Tag	Properties and terms annotated with this term MUST contain a valid MIME type
ContentDisposition	ContentDispositionType	The content disposition of the media stream of the annotated entity type marked with HasStream="true" or the annotated binary, stream, or string property or term
OptimisticConcurrency	[PropertyPath]	Data modification requires the use of ETags. A non-empty collection contains the set of properties that are used to compute the ETag. An empty collection means that the service won't tell how it computes the ETag
AdditionalProperties	Tag	Instances of this type may contain properties in addition to those declared in \$metadata If specified as false clients can assume that instances will not contain dynamic properties, irrespective of the value of the OpenType attribute.
AutoExpand	Tag	The service will automatically expand this stream property, navigation property, or the media stream of this media entity type even if not requested with \$expand
AutoExpandReferences	Tag	The service will automatically expand this navigation property as entity references even if not requested with <i>expand</i> = ... /ref
MayImplement	[QualifiedTypeName]	A collection of qualified type names outside of the type hierarchy that instances of this type might be addressable as by using a type-cast segment.
Ordered	Tag	Collection has a stable order. Ordered collections of primitive or complex types can be indexed by ordinal.
PositionalInsert	Tag	Items can be inserted at a given ordinal index.
AlternateKeys	[AlternateKey]	Communicates available alternate keys
OptionalParameter	OptionalParameterType	Supplying a value for the action or function parameter is optional. All parameters marked as optional must come after any parameters not marked as optional. The binding parameter must not be marked as optional.

Term	Type	Description
OperationAvailable	Boolean?	Action or function is available The annotation value will usually be an expression, e.g. using properties of the binding parameter type for instance-dependent availability, or using properties of a singleton for global availability. The static value <code>null</code> means that availability cannot be determined upfront and is instead expressed as an operation advertisement.
RequiresExplicitBinding	Tag?	This bound action or function is only available on model elements annotated with the <code>ExplicitOperationBindings</code> term.
ExplicitOperationBindings	[QualifiedBoundOperationName]	The qualified names of explicitly bound operations that are supported on the target model element. These operations are in addition to any operations not annotated with <code>RequiresExplicitBinding</code> that are bound to the type of the target model element.
SymbolicName	SimpleIdentifier	A symbolic name for a model element
GeometryFeature	GeometryFeatureType?	A Feature Object represents a spatially bounded thing
AnyStructure	Tag	Instances of a type are annotated with this tag if they have no common structure in a given response payload The select-list of a context URL MUST be <code>(@Core.AnyStructure)</code> if it would otherwise be empty, but this instance annotation SHOULD be omitted from the response value.

[RevisionType](#)

Property	Type	Description
Version	String?	The schema version with which this revision was first published
Kind	RevisionKind	The kind of revision
Description	String	Text describing the reason for the revision

[RevisionKind](#)

Member	Value	Description
Added	0	Model element was added
Modified	1	Model element was modified
Deprecated	2	Model element was deprecated

Link

The Link term is inspired by the `atom:link` element, see [RFC4287](#), and the `Link` HTTP header, see [RFC5988](#)

Property	Type	Description
rel	String	Link relation type, see IANA Link Relations
href	URL	URL of related information

ExampleValue

Derived Types:

- [PrimitiveExampleValue](#)
- [ComplexExampleValue](#)
- [EntityExampleValue](#)
- [ExternalExampleValue](#)

Property	Type	Description
Description	String?	Description of the example value

PrimitiveExampleValue: ExampleValue

Property	Type	Description
Description	String?	Description of the example value
Value	PrimitiveType	Example value for the custom parameter

ComplexExampleValue: ExampleValue

Property	Type	Description
Description	String?	Description of the example value
Value	ComplexType	Example value for the custom parameter

EntityExampleValue: ExampleValue

Property	Type	Description
Description	String?	Description of the example value
Value	EntityType	Example value for the custom parameter

ExternalExampleValue: ExampleValue

Property	Type	Description
Description	String?	Description of the example value

Property	Type	Description
ExternalValue	URL	Url reference to the value in its literal format

[MessageType](#)

Property	Type	Description
code	String	Machine-readable, language-independent message code
message	String	Human-readable, language-dependent message text
severity	MessageSeverity	Severity of the message
target	String?	A path to the target of the message detail, relative to the annotated instance
details	[MessageType]	List of detail messages

[MessageSeverity](#)

Type: String

Allowed Value	Description
success	Positive feedback - no action required
info	Additional information - no action required
warning	Warning - action may be required
error	Error - action is required

[ExceptionType](#)

Derived Types:

- [ValueExceptionType](#)
- [ResourceExceptionType](#)
- [DataModificationExceptionType](#)

Property	Type	Description
info	MessageType?	Information about the exception

[ValueExceptionType](#): [ExceptionType](#)

Property	Type	Description
info	MessageType?	Information about the exception
value	String?	String representation of the exact value

ResourceExceptionType: ExceptionType

Property	Type	Description
info	MessageType?	Information about the exception
retryLink	URL?	A GET request to this URL retries retrieving the problematic instance

DataModificationExceptionType: ExceptionType

Property	Type	Description
info	MessageType?	Information about the exception
failedOperation	DataModificationOperationKind	The kind of modification operation that failed
responseCode	Int16?	Response code of the failed operation, e.g. 424 for a failed dependency

DataModificationOperationKind

Member	Value	Description
insert	0	Insert new instance
update	1	Update existing instance
upsert	2	Insert new instance or update it if it already exists
delete	3	Delete existing instance
invoke	4	Invoke action or function
link	5	Add link between entities
unlink	6	Remove link between entities

Tag

Type: Boolean

This is the type to use for all tagging terms

Permission

Flag Member	Value	Description
None	0	No permissions
Read	1	Read permission
Write	2	Write permission
ReadWrite	3	Read and write permission
Invoke	4	Permission to invoke actions

[ContentDispositionType](#)

Property	Type	Description
Type	String	The disposition type of the binary or stream value, see RFC 6266, Disposition Type
Filename	String?	The proposed filename for downloading the binary or stream value, see RFC 6266, Disposition Parameter: 'Filename'

[QualifiedTermName](#)

Type: String

The qualified name of a term in scope.

[QualifiedTypeName](#)

Type: String

The qualified name of a type in scope.

[QualifiedActionName](#)

Type: String

The qualified name of an action in scope.

[QualifiedBoundOperationName](#)

Type: String

The qualified name of a bound action or function in scope.

Either

- the qualified name of an action, to indicate the single bound overload with the specified binding parameter type,
- the qualified name of a function, to indicate all bound overloads with the specified binding parameter type, or
- the qualified name of a function followed by parentheses containing a comma-separated list of parameter types, in the order of their definition, to identify a single function overload with the first (binding) parameter matching the specified parameter type.

[AlternateKey](#)

Property	Type	Description
Key	[PropertyRef]	The set of properties that make up this key

[PropertyRef](#)

Property	Type	Description
Name	PropertyPath	A path expression resolving to a primitive property of the entity type itself or to a primitive property of a complex or navigation property (recursively) of the entity type. The names of the properties in the path are joined together by forward slashes.

Property	Type	Description
Alias	String	A SimpleIdentifier that MUST be unique within the set of aliases, structural and navigation properties of the containing entity type that MUST be used in the key predicate of URLs

Dictionary

A dictionary of name-value pairs. Names must be valid property names, values may be restricted to a list of types via an annotation with term `Validation.OpenPropertyTypeConstraint`.

Property	Type
Any simple identifier	Any type listed in <code>Validation.OpenPropertyTypeConstraint</code> , or any type if there is no constraint

OptionalParameterType

Property	Type	Description
DefaultValue	String?	Default value for an optional parameter of primitive or enumeration type, using the same rules as the <code>cast</code> function in URLs. If no explicit <code>DefaultValue</code> is specified, the service is free on how to interpret omitting the parameter from the request. For example, a service might interpret an omitted optional parameter <code>KeyDate</code> as having the current date.

LocalDateTime

Type: String

A string representing a Local Date-Time value with no offset.

SimpleIdentifier

Type: String

A [simple identifier](#)

GeometryFeatureType

A [Feature Object](#) represents a spatially bounded thing

Property	Type	Description
geometry	Geometry?	Location of the Feature
properties	Dictionary?	Properties of the Feature
id	String?	Commonly used identifier for a Feature

3 Capabilities Vocabulary

Namespace: [Org.OData.Capabilities.V1](#)

Terms describing capabilities of a service

There are some capabilities which are strongly recommended for services to support even though they are optional. Support for \$top and \$skip is a good example as supporting these query options helps with performance of a service and are essential. Such capabilities are assumed to be default capabilities of an OData service even in the case that a capabilities annotation doesn't exist. Capabilities annotations are mainly expected to be used to explicitly specify that a service doesn't support such capabilities. Capabilities annotations can as well be used to declaratively specify the support of such capabilities.

On the other hand, there are some capabilities that a service may choose to support or not support and in varying degrees. \$filter and \$orderby are such good examples. This vocabulary aims to define terms to specify support or no support for such capabilities.

A service is assumed to support by default the following capabilities even though an annotation doesn't exist:

- Countability (\$count)
- Client pageability (\$top, \$skip)
- Expandability (\$expand)
- Indexability by key
- Batch support (\$batch)
- Navigability of navigation properties

A service is expected to support the following capabilities. If not supported, the service is expected to call out the restrictions using annotations:

- Filterability (\$filter)
- Sortability (\$orderby)
- Queryability of top level entity sets
- Query functions

A client cannot assume that a service supports certain capabilities. A client can try, but it needs to be prepared to handle an error in case the following capabilities are not supported:

- Insertability
- Updatability
- Deletability

Terms

Term	Type	Description
ConformanceLevel	ConformanceLevelType	The conformance level achieved by this service
SupportedFormats	[MediaType]	Media types of supported formats, including format parameters
SupportedMetadataFormats	[MediaType]	Media types of supported formats for \$metadata, including format parameters

Term	Type	Description
AcceptableEncodings	[String]	List of acceptable compression methods for (\$batch) requests, e.g. gzip
AsynchronousRequestsSupported	Tag	Service supports the asynchronous request preference
BatchContinueOnErrorSupported	Tag	Service supports the continue on error preference. Supports \$batch requests. Services that apply the BatchContinueOnErrorSupported term should also specify the ContinueOnErrorSupported property from the BatchSupport term.
IsolationSupported	IsolationLevel	Supported odata.isolation levels
CrossJoinSupported	Tag	Supports cross joins for the entity sets in this container
CallbackSupported	CallbackType	Supports callbacks for the specified protocols
ChangeTracking	ChangeTrackingType	Change tracking capabilities of this service or entity set
CountRestrictions	CountRestrictionsType	Restrictions on /\$count path suffix and \$count=true system query option
NavigationRestrictions	NavigationRestrictionsType	Restrictions on navigating properties according to OData URL conventions Restrictions specified on an entity set are valid whether the request is directly to the entity set or through a navigation property bound to that entity set. Services can specify a different set of restrictions specific to a path, in which case the more specific restrictions take precedence.
IndexableByKey	Tag	Supports key values according to OData URL conventions
TopSupported	Tag	Supports \$top
SkipSupported	Tag	Supports \$skip
ComputeSupported	Tag	Supports \$compute
SelectSupport	SelectSupportType	Support for \$select and nested query options within \$select

Term	Type	Description
BatchSupported	Tag	Supports \$batch requests. Services that apply the BatchSupported term should also apply the more comprehensive BatchSupport term.
BatchSupport	BatchSupportType	Batch Support for the service
FilterFunctions	[String]	List of functions and operators supported in filter expressions If not specified, null, or empty, all functions and operators may be attempted.
FilterRestrictions	FilterRestrictionsType	Restrictions on filter expressions
SortRestrictions	SortRestrictionsType	Restrictions on orderby expressions
ExpandRestrictions	ExpandRestrictionsType	Restrictions on expand expressions
SearchRestrictions	SearchRestrictionsType	Restrictions on search expressions
KeyAsSegmentSupported	Tag	Supports key-as-segment convention for addressing entities within a collection
QuerySegmentSupported	Tag	Supports passing query options in the request body
InsertRestrictions	InsertRestrictionsType	Restrictions on insert operations
DeepInsertSupport	DeepInsertSupportType?	Deep Insert Support of the annotated resource (the whole service, an entity set, or a collection-valued resource)
UpdateRestrictions	UpdateRestrictionsType	Restrictions on update operations
DeepUpdateSupport	DeepUpdateSupportType	Deep Update Support of the annotated resource (the whole service, an entity set, or a collection-valued resource)
DeleteRestrictions	DeleteRestrictionsType	Restrictions on delete operations
CollectionPropertyRestrictions	[CollectionPropertyRestrictionsType]	Describes restrictions on operations applied to collection-valued structural properties
OperationRestrictions	OperationRestrictionsType	Restrictions for function or action operation
AnnotationValuesInQuerySupported	Tag	Supports annotation values within system query options

Term	Type	Description
ModificationQueryOptions	ModificationQueryOptionsType	Support for query options with modification requests (insert, update, action invocation)
ReadRestrictions	ReadRestrictionsType	Restrictions for retrieving a collection of entities, retrieving a singleton instance.
CustomHeaders	[CustomParameter]	Custom headers that are supported/required for the annotated resource (Example)
CustomQueryOptions	[CustomParameter]	Custom query options that are supported/required for the annotated resource (Example) If the entity container is annotated, the query option is supported/required by all resources in that container.
MediaLocationUpdateSupported	Tag	Stream property or media stream supports update of its media edit URL and/or media read URL
DefaultCapabilities	DefaultCapabilitiesType	<p>Default capability settings for all collection-valued resources in the container</p> <p>Annotating a specific capability term, which is included as property in <code>DefaultCapabilitiesType</code>, for a specific collection-valued resource overrides the default capability with the specified properties using PATCH semantics:</p> <ul style="list-style-type: none"> Primitive or collection-valued properties specified in the specific capability term replace the corresponding properties specified in <code>DefaultCapabilities</code> Complex-valued properties specified in the specific capability term override the corresponding properties specified in <code>DefaultCapabilities</code> using PATCH semantics recursively Properties specified neither in the specific term nor in <code>DefaultCapabilities</code> have their default value

ConformanceLevelType

Member	Value	Description
Minimal	0	Minimal conformance level
Intermediate	1	Intermediate conformance level
Advanced	2	Advanced conformance level

IsolationLevel

Flag Member	Value	Description
Snapshot	1	All data returned for a request, including multiple requests within a batch or results retrieved across multiple pages, will be consistent as of a single point in time

CallbackType

A non-empty collection lists the full set of supported protocols. A empty collection means 'only HTTP is supported'

Property	Type	Description
CallbackProtocols	[CallbackProtocol]	List of supported callback protocols, e.g. <code>http</code> or <code>wss</code>

CallbackProtocol

Property	Type	Description
Id	String?	Protocol Identifier
UrlTemplate	String?	URL Template including parameters. Parameters are enclosed in curly braces {} as defined in RFC6570
DocumentationUrl	URL?	Human readable description of the meaning of the URL Template parameters

ChangeTrackingBase

Derived Types:

- [ChangeTrackingType](#)

Property	Type	Description
Supported	Boolean	odata.track-changes preference is supported

ChangeTrackingType: ChangeTrackingBase

Property	Type	Description
Supported	Boolean	odata.track-changes preference is supported

Property	Type	Description
FilterableProperties	[PropertyPath]	Change tracking supports filters on these properties If no properties are specified or FilterableProperties is omitted, clients cannot assume support for filtering on any properties in combination with change tracking.
ExpandableProperties	[NavigationPropertyPath]	Change tracking supports these properties expanded If no properties are specified or ExpandableProperties is omitted, clients cannot assume support for expanding any properties in combination with change tracking.

[CountRestrictionsBase](#)

Derived Types:

- [CountRestrictionsType](#)

Property	Type	Description
Countable	Boolean	Instances can be counted in requests targeting a collection

[CountRestrictionsType: CountRestrictionsBase](#)

Property	Type	Description
Countable	Boolean	Instances can be counted in requests targeting a collection
NonCountableProperties	[PropertyPath]	Members of these collection properties cannot be counted
NonCountableNavigationProperties	[NavigationPropertyPath]	Members of these navigation properties cannot be counted

[NavigationRestrictionsType](#)

Property	Type	Description
Navigability	NavigationType?	Default navigability for all navigation properties of the annotation target. Individual navigation properties can override this value via RestrictedProperties/Navigability .
RestrictedProperties	[NavigationPropertyRestriction]	List of navigation properties with restrictions

[NavigationPropertyRestriction](#)

Using a property of [NavigationPropertyRestriction](#) in a [NavigationRestrictions](#) annotation is discouraged in favor of using an annotation with the corresponding term from this vocabulary and a target path starting with a container and ending in the [NavigationProperty](#), unless the favored alternative is impossible because a dynamic expression requires an instance path whose evaluation starts at the target of the [NavigationRestrictions](#) annotation. See [this example](#).

Property	Type	Description
NavigationProperty	NavigationPropertyPath	Navigation properties can be navigated The target path of a NavigationRestrictions annotation followed by this navigation property path addresses the resource to which the other properties of NavigationPropertyRestriction apply. Instance paths that occur in dynamic expressions are evaluated starting at the boundary between both paths, which must therefore be chosen accordingly.
Navigability	NavigationType?	Supported navigability of this navigation property
FilterFunctions	[String]	List of functions and operators supported in filter expressions If not specified, null, or empty, all functions and operators may be attempted.
FilterRestrictions	FilterRestrictionsType?	Restrictions on filter expressions
SearchRestrictions	SearchRestrictionsType?	Restrictions on search expressions
SortRestrictions	SortRestrictionsType?	Restrictions on orderby expressions
TopSupported	Boolean	Supports \$top
SkipSupported	Boolean	Supports \$skip
SelectSupport	SelectSupportType?	Support for \$select
IndexableByKey	Boolean	Supports key values according to OData URL conventions
InsertRestrictions	InsertRestrictionsType?	Restrictions on insert operations
DeepInsertSupport	DeepInsertSupportType?	Deep Insert Support of the annotated resource (the whole service, an entity set, or a collection-valued resource)
UpdateRestrictions	UpdateRestrictionsType?	Restrictions on update operations
DeepUpdateSupport	DeepUpdateSupportType?	Deep Update Support of the annotated resource (the whole service, an entity set, or a collection-valued resource)
DeleteRestrictions	DeleteRestrictionsType?	Restrictions on delete operations
OptimisticConcurrencyControl	Boolean	Data modification (including insert) along this navigation property requires the use of ETags
ReadRestrictions	ReadRestrictionsType?	Restrictions for retrieving entities

[NavigationType](#)

Member	Value	Description
Recursive	0	Navigation properties can be recursively navigated
Single	1	Navigation properties can be navigated to a single level
None	2	Navigation properties are not navigable

[SelectSupportType](#)

Property	Type	Description
Supported	Boolean	Supports \$select
InstanceAnnotationsSupported	Boolean	Supports instance annotations in \$select list
Expandable	Boolean	\$expand within \$select is supported
Filterable	Boolean	\$filter within \$select is supported
Searchable	Boolean	\$search within \$select is supported
TopSupported	Boolean	\$top within \$select is supported
SkipSupported	Boolean	\$skip within \$select is supported
ComputeSupported	Boolean	\$compute within \$select is supported
Countable	Boolean	\$count within \$select is supported
Sortable	Boolean	\$orderby within \$select is supported

[BatchSupportType](#)

Property	Type	Description
Supported	Boolean	Service supports requests to \$batch
ContinueOnErrorSupported	Boolean	Service supports the continue on error preference
ReferencesInRequestBodiesSupported	Boolean	Service supports Content-ID referencing in request bodies
ReferencesAcrossChangeSetsSupported	Boolean	Service supports Content-ID referencing across change sets
EtagReferencesSupported	Boolean	Service supports referencing Etags from previous requests
RequestDependencyConditionsSupported	Boolean	Service supports the if member in JSON batch requests

Property	Type	Description
SupportedFormats	[MediaType]	Media types of supported formats for \$batch Allowed values: multipart/mixed Multipart Batch Format application/json JSON Batch Format

Applicable Annotation Terms:

- [Description](#)
- [LongDescription](#)

[FilterRestrictionsBase](#)**Derived Types:**

- [FilterRestrictionsType](#)

Property	Type	Description
Filterable	Boolean	\$filter is supported
RequiresFilter	Boolean	\$filter is required
MaxLevels	Int32	The maximum number of levels (including recursion) that can be traversed in a filter expression. A value of -1 indicates there is no restriction.

Applicable Annotation Terms:

- [Description](#)

[FilterRestrictionsType](#): [FilterRestrictionsBase](#)

Property	Type	Description
Filterable	Boolean	\$filter is supported
RequiresFilter	Boolean	\$filter is required
MaxLevels	Int32	The maximum number of levels (including recursion) that can be traversed in a filter expression. A value of -1 indicates there is no restriction.
RequiredProperties	[PropertyPath]	These properties must be specified in the \$filter clause (properties of derived types are not allowed here)
NonFilterableProperties	[PropertyPath]	These structural properties cannot be used in filter expressions

Property	Type	Description
FilterExpressionRestrictions	[FilterExpressionRestrictionType]	These properties only allow a subset of filter expressions. A valid filter expression for a single property can be enclosed in parentheses and combined by and with valid expressions for other properties.

Applicable Annotation Terms:

- [Description](#)

[FilterExpressionRestrictionType](#)

Property	Type	Description
Property	PropertyPath?	Path to the restricted property
AllowedExpressions	FilterExpressionType?	Allowed subset of expressions

[FilterExpressionType](#)**Type:** String

Allowed Value	Description
SingleValue	Property can be used in a single eq clause
MultiValue	Property can be used in multiple eq and in clauses, combined by or (which is logically equivalent to a single in clause)
SingleRange	Property can be used in at most one ge and/or one le clause, separated by and
MultiRange	Property can be compared to a union of one or more closed, half-open, or open intervals The filter expression for this property consists of one or more interval expressions combined by or . A single interval expression is either a single comparison of the property and a literal value with eq , le , lt , ge , or gt , or pair of boundaries combined by and and enclosed in parentheses. The lower boundary is either ge or gt , the upper boundary either le or lt .
SearchExpression	String property can be used as first operand in startswith , endswith , and contains clauses
MultiRangeOrSearchExpression	Property can be compared to a union of zero or more closed, half-open, or open intervals plus zero or more simple string patterns The filter expression for this property consists of one or more interval expressions or string comparison functions combined by or . See MultiRange for a definition of an interval expression. See SearchExpression for the allowed string comparison functions.

[SortRestrictionsBase](#)**Derived Types:**

- [SortRestrictionsType](#)

Property	Type	Description
Sortable	Boolean	\$orderby is supported

Applicable Annotation Terms:

- [Description](#)

[SortRestrictionsType](#): [SortRestrictionsBase](#)

Property	Type	Description
Sortable	Boolean	\$orderby is supported
AscendingOnlyProperties	[PropertyPath]	These properties can only be used for sorting in Ascending order
DescendingOnlyProperties	[PropertyPath]	These properties can only be used for sorting in Descending order
NonSortableProperties	[PropertyPath]	These structural properties cannot be used in orderby expressions

Applicable Annotation Terms:

- [Description](#)

[ExpandRestrictionsBase](#)

Derived Types:

- [ExpandRestrictionsType](#)

Property	Type	Description
Expandable	Boolean	\$expand is supported
StreamsExpandable	Boolean	\$expand is supported for stream properties and media streams
MaxLevels	Int32	The maximum number of levels that can be expanded in a expand expression. A value of -1 indicates there is no restriction.

Applicable Annotation Terms:

- [Description](#)

[ExpandRestrictionsType](#): [ExpandRestrictionsBase](#)

Property	Type	Description
Expandable	Boolean	\$expand is supported
StreamsExpandable	Boolean	\$expand is supported for stream properties and media streams

Property	Type	Description
MaxLevels	Int32	The maximum number of levels that can be expanded in a expand expression. A value of -1 indicates there is no restriction.
NonExpandableProperties	[NavigationPropertyPath]	These properties cannot be used in expand expressions
NonExpandableStreamProperties	[PropertyPath]	These stream properties cannot be used in expand expressions

Applicable Annotation Terms:

- [Description](#)

[SearchRestrictionsType](#)

Property	Type	Description
Searchable	Boolean	\$search is supported
UnsupportedExpressions	SearchExpressions	Expressions not supported in \$search

Applicable Annotation Terms:

- [Description](#)

[SearchExpressions](#)

Flag Member	Value	Description
none	0	Single search term
AND	1	Multiple search terms, optionally separated by AND
OR	2	Multiple search terms separated by OR
NOT	4	Search terms preceded by NOT
phrase	8	Search phrases enclosed in double quotes
group	16	Precedence grouping of search expressions with parentheses

[InsertRestrictionsBase](#)**Derived Types:**

- [InsertRestrictionsType](#)

Property	Type	Description
Insertable	Boolean	Entities can be inserted

Property	Type	Description
MaxLevels	Int32	The maximum number of navigation properties that can be traversed when addressing the collection to insert into. A value of -1 indicates there is no restriction.
TypecastSegmentSupported	Boolean	Entities of a specific derived type can be created by specifying a type-cast segment
QueryOptions	ModificationQueryOptionsType?	Support for query options with insert requests
CustomHeaders	[CustomParameter]	Supported or required custom headers
CustomQueryOptions	[CustomParameter]	Supported or required custom query options
Description	String?	A brief description of the request
LongDescription	String?	A long description of the request
ErrorResponses	[HttpResponse]	Possible error responses returned by the request.

[InsertRestrictionsType](#): [InsertRestrictionsBase](#)

Property	Type	Description
Insertable	Boolean	Entities can be inserted
MaxLevels	Int32	The maximum number of navigation properties that can be traversed when addressing the collection to insert into. A value of -1 indicates there is no restriction.
TypecastSegmentSupported	Boolean	Entities of a specific derived type can be created by specifying a type-cast segment
QueryOptions	ModificationQueryOptionsType?	Support for query options with insert requests
CustomHeaders	[CustomParameter]	Supported or required custom headers
CustomQueryOptions	[CustomParameter]	Supported or required custom query options
Description	String?	A brief description of the request
LongDescription	String?	A long description of the request
ErrorResponses	[HttpResponse]	Possible error responses returned by the request.
NonInsertableProperties	[PropertyPath]	These structural properties cannot be specified on insert

Property	Type	Description
NonInsertableNavigationProperties	[NavigationPropertyPath]	These navigation properties do not allow deep inserts
RequiredProperties	[PropertyPath]	These structural properties must be specified on insert
Permissions	[PermissionType?]	Required permissions. One of the specified sets of scopes is required to perform the insert.

[PermissionType](#)

Property	Type	Description
SchemeName	SchemeName	Authorization flow scheme name
Scopes	[ScopeType]	List of scopes that can provide access to the resource

[ScopeType](#)

Property	Type	Description
Scope	String	Name of the scope.
RestrictedProperties	String?	Comma-separated string value of all properties that will be included or excluded when using the scope. Possible string value identifiers when specifying properties are *, <i>PropertyName</i> , - <i>PropertyName</i> . * denotes all properties are accessible. - <i>PropertyName</i> excludes that specific property. <i>PropertyName</i> explicitly provides access to the specific property. The absence of <i>RestrictedProperties</i> denotes all properties are accessible using that scope.

[DeepInsertSupportType](#)

Property	Type	Description
Supported	Boolean	Annotation target supports deep inserts
ContentIDSupported	Boolean	Annotation target supports accepting and returning nested entities annotated with the <code>Core.ContentID</code> instance annotation.

[UpdateRestrictionsBase](#)

Derived Types:

- [UpdateRestrictionsType](#)

Property	Type	Description
Updatable	Boolean	Entities can be updated

Property	Type	Description
Upsertable	Boolean	Entities can be upserted
DeltaUpdateSupported	Boolean	Entities can be inserted, updated, and deleted via a PATCH request with a delta payload
UpdateMethod	HttpMethod?	Supported HTTP Methods (PUT or PATCH) for updating an entity. If null, PATCH SHOULD be supported and PUT MAY be supported.
FilterSegmentSupported	Boolean	Members of collections can be updated via a PATCH request with a <code>/\$filter(...)/\$each</code> segment
TypecastSegmentSupported	Boolean	Members of collections can be updated via a PATCH request with a type-cast segment and a <code>/\$each</code> segment
MaxLevels	Int32	The maximum number of navigation properties that can be traversed when addressing the collection or entity to update. A value of -1 indicates there is no restriction.
Permissions	[PermissionType?]	Required permissions. One of the specified sets of scopes is required to perform the update.
QueryOptions	ModificationQueryOptionsType?	Support for query options with update requests
CustomHeaders	[CustomParameter]	Supported or required custom headers
CustomQueryOptions	[CustomParameter]	Supported or required custom query options
Description	String?	A brief description of the request
LongDescription	String?	A long description of the request
ErrorResponses	[HttpResponse]	Possible error responses returned by the request.

[UpdateRestrictionsType: UpdateRestrictionsBase](#)

Property	Type	Description
Updatable	Boolean	Entities can be updated
Upsertable	Boolean	Entities can be upserted
DeltaUpdateSupported	Boolean	Entities can be inserted, updated, and deleted via a PATCH request with a delta payload
UpdateMethod	HttpMethod?	Supported HTTP Methods (PUT or PATCH) for updating an entity. If null, PATCH SHOULD be supported and PUT MAY be supported.

Property	Type	Description
<i>FilterSegmentSupported</i>	Boolean	Members of collections can be updated via a PATCH request with a /\$filter(...)/\$each segment
<i>TypecastSegmentSupported</i>	Boolean	Members of collections can be updated via a PATCH request with a type-cast segment and a /\$each segment
<i>MaxLevels</i>	Int32	The maximum number of navigation properties that can be traversed when addressing the collection or entity to update. A value of -1 indicates there is no restriction.
<i>Permissions</i>	[<i>PermissionType?</i>]	Required permissions. One of the specified sets of scopes is required to perform the update.
<i>QueryOptions</i>	<i>ModificationQueryOptionsType?</i>	Support for query options with update requests
<i>CustomHeaders</i>	[<i>CustomParameter</i>]	Supported or required custom headers
<i>CustomQueryOptions</i>	[<i>CustomParameter</i>]	Supported or required custom query options
<i>Description</i>	String?	A brief description of the request
<i>LongDescription</i>	String?	A long description of the request
<i>ErrorResponses</i>	[<i>HttpResponse</i>]	Possible error responses returned by the request.
<i>NonUpdatableProperties</i>	[PropertyPath]	These structural properties cannot be specified on update
<i>NonUpdatableNavigationProperties</i>	[NavigationPropertyPath]	These navigation properties do not allow rebinding
<i>RequiredProperties</i>	[PropertyPath]	These structural properties must be specified on update

HttpMethod

Flag Member	Value	Description
<i>GET</i>	1	The HTTP GET Method
<i>PATCH</i>	2	The HTTP PATCH Method
<i>PUT</i>	4	The HTTP PUT Method
<i>POST</i>	8	The HTTP POST Method

Flag Member	Value	Description
DELETE	16	The HTTP DELETE Method
OPTIONS	32	The HTTP OPTIONS Method
HEAD	64	The HTTP HEAD Method

[DeepUpdateSupportType](#)

Property	Type	Description
Supported	Boolean	Annotation target supports deep updates
ContentIDSupported	Boolean	Annotation target supports accepting and returning nested entities annotated with the <code>Core.ContentID</code> instance annotation.

[DeleteRestrictionsBase](#)

Derived Types:

- [DeleteRestrictionsType](#)

Property	Type	Description
Deletable	Boolean	Entities can be deleted
MaxLevels	Int32	The maximum number of navigation properties that can be traversed when addressing the collection to delete from or the entity to delete. A value of -1 indicates there is no restriction.
FilterSegmentSupported	Boolean	Members of collections can be deleted via a DELETE request with a <code>/\$filter(...)/\$each</code> segment
TypecastSegmentSupported	Boolean	Members of collections can be deleted via a DELETE request with a type-cast segment and a <code>/\$each</code> segment
Permissions	[PermissionType?]	Required permissions. One of the specified sets of scopes is required to perform the delete.
CustomHeaders	[CustomParameter]	Supported or required custom headers
CustomQueryOptions	[CustomParameter]	Supported or required custom query options
Description	String?	A brief description of the request
LongDescription	String?	A long description of the request
ErrorResponses	[HttpResponse]	Possible error responses returned by the request.

[DeleteRestrictionsType](#): [DeleteRestrictionsBase](#)

Property	Type	Description
Deletable	Boolean	Entities can be deleted

Property	Type	Description
<u>MaxLevels</u>	Int32	The maximum number of navigation properties that can be traversed when addressing the collection to delete from or the entity to delete. A value of -1 indicates there is no restriction.
<u>FilterSegmentSupported</u>	Boolean	Members of collections can be deleted via a DELETE request with a <code>/\$filter(...)/\$each</code> segment
<u>TypecastSegmentSupported</u>	Boolean	Members of collections can be deleted via a DELETE request with a type-cast segment and a <code>/\$each</code> segment
<u>Permissions</u>	[<u>PermissionType?</u>]	Required permissions. One of the specified sets of scopes is required to perform the delete.
<u>CustomHeaders</u>	[<u>CustomParameter</u>]	Supported or required custom headers
<u>CustomQueryOptions</u>	[<u>CustomParameter</u>]	Supported or required custom query options
<u>Description</u>	String?	A brief description of the request
<u>LongDescription</u>	String?	A long description of the request
<u>ErrorResponses</u>	[<u>HttpResponse</u>]	Possible error responses returned by the request.
<u>NonDeletableNavigationProperties</u>	[NavigationPropertyPath]	These navigation properties do not allow DeleteLink requests

[CollectionPropertyRestrictionsType](#)

Property	Type	Description
<u>CollectionProperty</u>	PropertyPath?	Restricted Collection-valued property
<u>FilterFunctions</u>	[String]	List of functions and operators supported in filter expressions If not specified, null, or empty, all functions and operators may be attempted.
<u>FilterRestrictions</u>	<u>FilterRestrictionsType?</u>	Restrictions on filter expressions
<u>SearchRestrictions</u>	<u>SearchRestrictionsType?</u>	Restrictions on search expressions
<u>SortRestrictions</u>	<u>SortRestrictionsType?</u>	Restrictions on orderby expressions
<u>TopSupported</u>	Boolean	Supports \$top
<u>SkipSupported</u>	Boolean	Supports \$skip
<u>SelectSupport</u>	<u>SelectSupportType?</u>	Support for \$select
<u>Insertable</u>	Boolean	Members can be inserted into this collection If additionally annotated with <u>Core.PositionalInsert</u> , members can be inserted at a specific position

Property	Type	Description
Updatable	Boolean	Members of this ordered collection can be updated by ordinal
Deletable	Boolean	Members of this ordered collection can be deleted by ordinal

[OperationRestrictionsType](#)

Property	Type	Description
FilterSegmentSupported	Boolean	Bound action or function can be invoked on a collection-valued binding parameter path with a <code>/\$filter(...)</code> segment
Permissions	[PermissionType?]	Required permissions. One of the specified sets of scopes is required to invoke an action or function
CustomHeaders	[CustomParameter]	Supported or required custom headers
CustomQueryOptions	[CustomParameter]	Supported or required custom query options
ErrorResponses	[HttpResponse]	Possible error responses returned by the request.

[ModificationQueryOptionsType](#)

Property	Type	Description
ExpandSupported	Boolean	Supports \$expand with modification requests
SelectSupported	Boolean	Supports \$select with modification requests
ComputeSupported	Boolean	Supports \$compute with modification requests
FilterSupported	Boolean	Supports \$filter with modification requests
SearchSupported	Boolean	Supports \$search with modification requests
SortSupported	Boolean	Supports \$orderby with modification requests

[ReadRestrictionsBase](#)

Derived Types:

- [ReadByKeyRestrictionsType](#)
- [ReadRestrictionsType](#)

Property	Type	Description
Readable	Boolean	Entities can be retrieved
Permissions	[PermissionType?]	Required permissions. One of the specified sets of scopes is required to read.
CustomHeaders	[CustomParameter]	Supported or required custom headers
CustomQueryOptions	[CustomParameter]	Supported or required custom query options

Property	Type	Description
Description	String?	A brief description of the request
LongDescription	String?	A long description of the request
ErrorResponses	[HttpResponse]	Possible error responses returned by the request.

ReadByKeyRestrictionsType: ReadRestrictionsBase

Restrictions for retrieving an entity by key

Property	Type	Description
Readable	Boolean	Entities can be retrieved
Permissions	[PermissionType?]	Required permissions. One of the specified sets of scopes is required to read.
CustomHeaders	[CustomParameter]	Supported or required custom headers
CustomQueryOptions	[CustomParameter]	Supported or required custom query options
Description	String?	A brief description of the request
LongDescription	String?	A long description of the request
ErrorResponses	[HttpResponse]	Possible error responses returned by the request.

ReadRestrictionsType: ReadRestrictionsBase

Property	Type	Description
Readable	Boolean	Entities can be retrieved
Permissions	[PermissionType?]	Required permissions. One of the specified sets of scopes is required to read.
CustomHeaders	[CustomParameter]	Supported or required custom headers
CustomQueryOptions	[CustomParameter]	Supported or required custom query options
Description	String?	A brief description of the request
LongDescription	String?	A long description of the request
ErrorResponses	[HttpResponse]	Possible error responses returned by the request.
TypecastSegmentSupported	Boolean	Entities of a specific derived type can be read by specifying a type-cast segment
ReadByKeyRestrictions	ReadByKeyRestrictionsType?	Restrictions for retrieving an entity by key Only valid when applied to a collection. If a property of ReadByKeyRestrictions is not specified, the corresponding property value of ReadRestrictions applies.

CustomParameter

A custom parameter is either a header or a query option

The type of a custom parameter is always a string. Restrictions on the parameter values can be expressed by annotating the record expression describing the parameter with terms from the Validation vocabulary, e.g. Validation.Pattern or Validation.AllowedValues.

Property	Type	Description
Name	String	Name of the custom parameter
Description	String?	Description of the custom parameter
DocumentationURL	URL?	URL of related documentation
Required	Boolean	true: parameter is required, false or not specified: parameter is optional
ExampleValues	[PrimitiveExampleValue]	Example values for the custom parameter

DefaultCapabilitiesType

Property	Type	Description
ChangeTracking	ChangeTrackingBase?	Change tracking capabilities
CountRestrictions	CountRestrictionsBase?	Restrictions on /\$count path suffix and \$count=true system query option
IndexableByKey	Tag?	Supports key values according to OData URL conventions
TopSupported	Tag?	Supports \$top
SkipSupported	Tag?	Supports \$skip
ComputeSupported	Tag?	Supports \$compute
SelectSupport	SelectSupportType?	Support for \$select and nested query options within \$select
FilterRestrictions	FilterRestrictionsBase?	Restrictions on filter expressions
SortRestrictions	SortRestrictionsBase?	Restrictions on orderby expressions
ExpandRestrictions	ExpandRestrictionsBase?	Restrictions on expand expressions
SearchRestrictions	SearchRestrictionsType?	Restrictions on search expressions
InsertRestrictions	InsertRestrictionsBase?	Restrictions on insert operations
UpdateRestrictions	UpdateRestrictionsBase?	Restrictions on update operations
DeleteRestrictions	DeleteRestrictionsBase?	Restrictions on delete operations
OperationRestrictions	OperationRestrictionsType?	Restrictions for function or action operations

Property	Type	Description
ReadRestrictions	ReadRestrictionsType?	Restrictions for retrieving a collection of entities, retrieving a singleton instance

[HttpResponse](#)

Property	Type	Description
StatusCode	String	HTTP response status code, for example 400, 403, 501
Description	String	Human-readable description of the response

4 Measures Vocabulary

Namespace: [Org.OData.Measures.V1](#)

Terms describing monetary amounts and measured quantities

Terms

Term	Type	Description
ISOCurrency	String	The currency for this monetary amount as an ISO 4217 currency code
Scale	Byte	The number of significant decimal places in the scale part (less than or equal to the number declared in the Scale facet)
Unit	String	The unit of measure for this measured quantity, e.g. cm for centimeters or % for percentages
UNECEUnit	String	The unit of measure for this measured quantity, according to the UN/CEFACT Recommendation 20
DurationGranularity	DurationGranularityType	The minimum granularity of duration values. Absence of this annotation means a granularity of seconds with sub-seconds according to the Precision facet.

[DurationGranularityType](#)

Type: String

Allowed Value	Description
days	Duration in days, e.g. P1D
hours	Duration in days and hours, e.g. P1DT23H
minutes	Duration in days, hours, and minutes, e.g. P1DT23H59M

5 Validation Vocabulary

Namespace: [Org.OData.Validation.V1](#)

Terms describing validation rules

Terms

Term	Type	Description
Pattern	String	The pattern that a string property, parameter, or term must match. This SHOULD be a valid regular expression, according to the ECMA 262 regular expression dialect.
Minimum	PrimitiveType	Minimum value that a property, parameter, or term can have. Can be annotated with: <ul style="list-style-type: none"> • Exclusive
Maximum	PrimitiveType	Maximum value that a property, parameter, or term can have. Can be annotated with: <ul style="list-style-type: none"> • Exclusive
Exclusive	Tag	Tags a Minimum or Maximum as exclusive, i.e. an open interval boundary.
AllowedValues	[AllowedValue]	A collection of valid values for the annotated property, parameter, or type definition
MultipleOf	Decimal	The value of the annotated property, parameter, or term must be an integer multiple of this positive value. For temporal types, the value is measured in seconds.
Constraint	ConstraintType	Condition that the annotation target has to fulfill
ItemsOf	[ItemsOfType]	A list of constraints describing that entities related via one navigation property MUST also be related via another, collection-valued navigation property. The same path value MUST NOT occur more than once. Example: entity type <code>Customer</code> has navigation properties <code>AllOrders</code> , <code>OpenOrders</code> , and <code>ClosedOrders</code> . The term allows to express that items of <code>OpenOrders</code> and <code>ClosedOrders</code> are also items of the <code>AllOrders</code> navigation property, even though they are defined in an <code>Orders</code> entity set.
OpenPropertyTypeConstraint	[SingleOrCollectionType]	Dynamic properties added to the annotated open structured type are restricted to the listed types.

Term	Type	Description
DerivedTypeConstraint	[SingleOrCollectionType]	<p>Values are restricted to types that are both identical to or derived from the declared type and a type listed in this collection.</p> <p>This allows restricting values to certain sub-trees of an inheritance hierarchy, including hierarchies starting at the Built-In Abstract Types. Types listed in this collection are ignored if they are not derived from the declared type of the annotated model element or would not be allowed as declared type of the annotated model element.</p> <p>When applied to a collection-valued element, this annotation specifies the types allowed for members of the collection without mentioning the <code>Collection()</code> wrapper. The <code>SingleOrCollectionType</code> may only include the <code>Collection()</code> wrapper if the annotation is applied to an element with declared type <code>Edm.Untyped</code>.</p>
AllowedTerms	[QualifiedTermName]	<p>Annotate a term of type <code>Edm.AnnotationPath</code>, or a property of type <code>Edm.AnnotationPath</code> that is used within a structured term, to restrict the terms that can be targeted by the path.</p> <p>The annotation path expression is intended to end in a path segment with one of the listed terms. For forward compatibility, clients should be prepared for the annotation to reference terms besides those listed.</p>
ApplicableTerms	[QualifiedTermName]	Names of specific terms that are applicable and may be applied in the current context. This annotation does not restrict the use of other terms.
MaxItems	Int64	The annotated collection must have at most the specified number of items.
MinItems	Int64	The annotated collection must have at least the specified number of items.

AllowedValue

Property	Type	Description
Value	PrimitiveType?	An allowed value for the property, parameter, or type definition

Applicable Annotation Terms:

- [SymbolicName](#)

ConstraintType

Property	Type	Description
FailureMessage	String?	Human-readable message that can be shown to end users if the constraint is not fulfilled

Property	Type	Description
Condition	Boolean	Value MUST be a dynamic expression that evaluates to true if and only if the constraint is fulfilled

[ItemsOfType](#)

Entities related via the single- or collection-valued navigation property identified by `path` are also related via the collection-valued navigation property identified by `target`.

Property	Type	Description
path	NavigationPropertyPath	A path to a single- or collection-valued navigation property
target	NavigationPropertyPath	A path to a collection-valued navigation property

[SingleOrCollectionType](#)

Type: String

The qualified name of a type in scope, optionally wrapped in `Collection()` to denote a collection of instances of the type

6 Aggregation Vocabulary

Namespace: [Org.OData.Aggregation.V1](#)

Terms to describe which data in a given entity model can be aggregated, and how.

Related to the specification document [OData-Data-Agg-v4.0](#).

Terms

Term	Type	Description
ApplySupported	ApplySupportedType	This entity set or collection supports the \$apply system query option
ApplySupportedDefaults	ApplySupportedBase	<p>Default support of the \$apply system query option for all collection-valued resources in the container</p> <p>Annotating term ApplySupported for a specific collection-valued resource overrides the default support with the specified properties using PATCH semantics:</p> <ul style="list-style-type: none">• Primitive or collection-valued properties specified in ApplySupported replace the corresponding properties specified in ApplySupportedDefaults• Complex-valued properties specified in ApplySupported override the corresponding properties specified in ApplySupportedDefaults using PATCH semantics recursively• Properties specified neither in ApplySupported nor in ApplySupportedDefaults have their default value
Groupable (Deprecated)	Tag	Deprecated in favor of ApplySupported/GroupableProperties
Aggregatable (Deprecated)	Tag	Deprecated in favor of ApplySupported/AggregatableProperties
CustomAggregate	String	<p>Dynamic property that can be used in the aggregate transformation</p> <p>This term MUST be applied with a Qualifier, the Qualifier value is the name of the dynamic property. The value of the annotation MUST be the qualified name of a primitive type. The aggregated value will be of that type.</p>

Term	Type	Description
ContextDefiningProperties	[PropertyPath]	The annotated property or custom aggregate is only well-defined in the context of these properties The context-defining properties need either be part of the result entities, or be restricted to a single value by a pre-filter operation. Examples are postal codes within a country, or monetary amounts whose context is the unit of currency.
LeveledHierarchy	[PropertyPath]	Defines a leveled hierarchy (OData-Data-Agg-v4.0, section 5.5.1)
RecursiveHierarchy	RecursiveHierarchyType	Defines a recursive hierarchy (OData-Data-Agg-v4.0, section 5.5.2)
UpPath	[String]	The string values of the node identifiers in a path from the annotated node to a start node in a traversal of a recursive hierarchy This instance annotation occurs in the result set after a traverse transformation (OData-Data-Agg-v4.0, section 6.2.2.2). A use case for this is traversal with multiple parents, when this annotation takes as value one parent node identifier followed by one grandparent node identifier and so on.
AvailableOnAggregates	AvailableOnAggregatesType	This function is available on aggregated entities if the RequiredProperties are still defined

Functions

[isnode](#)

Is the entity a node of the hierarchy specified by the [parameter pair](#) ([HierarchyNodes](#), [HierarchyQualifier](#))? (See OData-Data-Agg-v4.0, section 5.5.2.1)

If a node identifier passed to the function is null, the function returns null.

Parameter	Type	Description
HierarchyNodes	[EntityType]	A collection, given through a \$root expression
HierarchyQualifier	HierarchyQualifier	
Node	PrimitiveType?	Node identifier of the putative node
→	Boolean?	

[isroot](#)

Is the entity a root node of the hierarchy specified by the [parameter pair](#) ([HierarchyNodes](#), [HierarchyQualifier](#))?

If a node identifier passed to the function is null, the function returns null.

Parameter	Type	Description
HierarchyNodes	[EntityType]	A collection, given through a <code>\$root</code> expression
HierarchyQualifier	HierarchyQualifier	
Node	PrimitiveType?	Node identifier of the putative root
\rightarrow	Boolean?	

[isdescendant](#)

Is the entity a descendant node of the ancestor node in the hierarchy specified by the [parameter pair](#) ([HierarchyNodes](#), [HierarchyQualifier](#)) with at most the specified distance? (See OData-Data-Agg-v4.0, section 5.5.2.1)

If a node identifier passed to the function is null, the function returns null.

Parameter	Type	Description
HierarchyNodes	[EntityType]	A collection, given through a <code>\$root</code> expression
HierarchyQualifier	HierarchyQualifier	
Node	PrimitiveType?	Node identifier of the putative descendant
Ancestor	PrimitiveType?	Node identifier of the ancestor node
MaxDistance	Int16	<i>Optional parameter</i>
IncludeSelf	Boolean	<i>Optional parameter: Whether to include the node itself in the result</i>
\rightarrow	Boolean?	

[isancestor](#)

Is the entity an ancestor node of the descendant node in the hierarchy specified by the [parameter pair](#) ([HierarchyNodes](#), [HierarchyQualifier](#)) with at most the specified distance? (See OData-Data-Agg-v4.0, section 5.5.2.1)

If a node identifier passed to the function is null, the function returns null.

Parameter	Type	Description
HierarchyNodes	[EntityType]	A collection, given through a <code>\$root</code> expression
HierarchyQualifier	HierarchyQualifier	
Node	PrimitiveType?	Node identifier of the putative ancestor
Descendant	PrimitiveType?	Node identifier of the descendant node
MaxDistance	Int16	<i>Optional parameter</i>
IncludeSelf	Boolean	<i>Optional parameter: Whether to include the node itself in the result</i>

Parameter	Type	Description
→	Boolean?	

[issibling](#)

Is the entity a sibling node of the other node in the hierarchy specified by the [parameter pair](#) ([HierarchyNodes](#), [HierarchyQualifier](#))? (See OData-Data-Agg-v4.0, section 5.5.2.1)

A node is not a sibling of itself. If a node identifier passed to the function is null, the function returns null.

Parameter	Type	Description
HierarchyNodes	[EntityType]	A collection, given through a <code>\$root</code> expression
HierarchyQualifier	HierarchyQualifier	
Node	PrimitiveType?	Node identifier of the putative sibling
Other	PrimitiveType?	Node identifier of the other node
→	Boolean?	

[isleaf](#)

Is the entity a leaf node in the hierarchy specified by the [parameter pair](#) ([HierarchyNodes](#), [HierarchyQualifier](#))? (See OData-Data-Agg-v4.0, section 5.5.2.1)

If a node identifier passed to the function is null, the function returns null.

Parameter	Type	Description
HierarchyNodes	[EntityType]	A collection, given through a <code>\$root</code> expression
HierarchyQualifier	HierarchyQualifier	
Node	PrimitiveType?	Node identifier of the putative leaf
→	Boolean?	

[rollupnode](#)

During `rolluprecursive` for a hierarchy node, this function returns the node

This function may only occur in the second parameter of a `groupby` transformation whose first parameter contains `rolluprecursive(...)`. It is evaluated as part of the transformation $R(x)$ in the “`rolluprecursive` algorithm” (OData-Data-Agg-v4.0, section 6.3). Its behavior is undefined outside of this algorithm.

```
Sales?$apply=groupby((rolluprecursive(...)), filter(SalesOrganization eq
Aggregation.rollupnode())/aggregate(...))
```

constructs a rollup that contains aggregates per hierarchy node while excluding descendants from the aggregation.

Parameter	Type	Description
Position	Int16	<i>Optional parameter:</i> Position N among the <code>rolluprecursive</code> operators in the first argument of <code>groupby</code> Every instance in the output set of a <code>groupby</code> transformation with M <code>rolluprecursive</code> operators has M relationships to M nodes in M recursive hierarchies. This function returns the node x with path r to the root in relationship number N. If several such <code>groupby</code> transformations are nested, this function refers to the innermost one.
→	EntityType	

ApplySupportedBase

Services that do not fully implement a certain aggregation-related functionality may document this by annotating the [ApplySupported](#) or [ApplySupportedDefaults](#) annotation with a description.

Derived Types:

- [ApplySupportedType](#)

Property	Type	Description
Transformations	[Transformation]	Transformations that can be used in <code>\$apply</code>
CustomAggregationMethods	[String]	Qualified names of custom aggregation methods that can be used in <code>aggregate(...with...)</code>
Rollup	RollupType	The service supports rollup hierarchies in a <code>groupby</code> transformation
From	Boolean	The service supports the <code>from</code> keyword in an <code>aggregate</code> transformation

ApplySupportedType: ApplySupportedBase

Property	Type	Description
Transformations	[Transformation]	Transformations that can be used in <code>\$apply</code>
CustomAggregationMethods	[String]	Qualified names of custom aggregation methods that can be used in <code>aggregate(...with...)</code>
Rollup	RollupType	The service supports rollup hierarchies in a <code>groupby</code> transformation
From	Boolean	The service supports the <code>from</code> keyword in an <code>aggregate</code> transformation
PropertyRestrictions (Deprecated)	Boolean	Deprecated since Groupable and Aggregatable are deprecated
GroupableProperties	[AnyPropertyPath]	A non-empty collection indicates that only the listed properties of the annotated target are supported by the <code>groupby</code> transformation

Property	Type	Description
AggregatableProperties	[AggregatablePropertyType]	A non-empty collection indicates that only the listed properties of the annotated target can be used in the <code>aggregate</code> transformation, optionally restricted to the specified aggregation methods

[AggregatablePropertyType](#)

Property	Type	Description
Property	PropertyPath	Aggregatable property
SupportedAggregationMethods	[AggregationMethod]	Standard and custom aggregation methods that can be applied to the property. If omitted, all aggregation methods can be applied
RecommendedAggregationMethod	AggregationMethod?	Recommended method for aggregating values of the property

[Transformation](#)

Type: String

A transformation that can be used in `$apply`

Allowed Value	Description
aggregate	OData-Data-Agg-v4.0, section 3.2.1
groupby	OData-Data-Agg-v4.0, section 3.2.3
concat	OData-Data-Agg-v4.0, section 3.2.2
identity	OData-Data-Agg-v4.0, section 3.4.1
filter	OData-Data-Agg-v4.0, section 3.3.2
search	OData-Data-Agg-v4.0, section 3.3.4
nest	OData-Data-Agg-v4.0, section 3.5.2
adnested	OData-Data-Agg-v4.0, section 3.4.3
join	OData-Data-Agg-v4.0, section 3.5.1
outerjoin	OData-Data-Agg-v4.0, section 3.5.1
compute	OData-Data-Agg-v4.0, section 3.4.2
bottomcount	OData-Data-Agg-v4.0, section 3.3.1.1
bottomsum	OData-Data-Agg-v4.0, section 3.3.1.3
bottompercent	OData-Data-Agg-v4.0, section 3.3.1.2

Allowed Value	Description
topcount	OData-Data-Agg-v4.0, section 3.3.1.1
topsum	OData-Data-Agg-v4.0, section 3.3.1.3
toppercent	OData-Data-Agg-v4.0, section 3.3.1.2
orderby	OData-Data-Agg-v4.0, section 3.3.3
top	OData-Data-Agg-v4.0, section 3.3.6
skip	OData-Data-Agg-v4.0, section 3.3.5
ancestors	OData-Data-Agg-v4.0, section 6.2.1
descendants	OData-Data-Agg-v4.0, section 6.2.1
traverse	OData-Data-Agg-v4.0, section 6.2.2

AggregationMethod

Type: String

Standard or custom aggregation method

Custom aggregation methods **MUST** use a namespace-qualified name, that is contain at least one dot.

Allowed Value	Description
sum	Can be applied to numeric values to return the sum of the non-null values, or null if there are no non-null values or the input set is empty
min	Can be applied to values with a totally ordered domain to return the smallest of the non-null values, or null if there are no non-null values or the input set is empty
max	Can be applied to values with a totally ordered domain to return the largest of the non-null values, or null if there are no non-null values or the input set is empty
average	Can be applied to numeric values to return the sum of the non-null values divided by the count of the non-null values, or null if there are no non-null values or the input set is empty
countdistinct	Counts the distinct values, omitting any null values For navigation properties, it counts the distinct entities in the union of all entities related to entities in the input set. For collection-valued primitive properties, it counts the distinct items in the union of all collection values in the input set.

RollupType

The number of `rollup` or `rolluprecursive` operators allowed in a `groupby` transformation

Member	Value	Description
None	0	No support for <code>rollup</code> or <code>rolluprecursive</code>

Member	Value	Description
SingleHierarchy	1	Only one rollup or rolluprecursive operator per groupby
MultipleHierarchies	2	Full support for rollup and rolluprecursive

[RecursiveHierarchyType](#)

Property	Type	Description
NodeProperty	PropertyPath	Primitive property holding the node identifier
ParentNavigationProperty	NavigationPropertyPath	Property for navigating to the parent node(s). Its type MUST be the entity type annotated with this term, and it MUST be collection-valued or nullable single-valued.

[HierarchyQualifier](#)

Type: String

Qualifier of a [RecursiveHierarchy](#) annotation

Every recursive hierarchy function defined in this vocabulary has

- a parameter `HierarchyQualifier` of this type and
- a parameter `HierarchyNodes` that is a collection of entities of a common type without multiple occurrences of the same entity.

`HierarchyQualifier` is the qualifier of a `RecursiveHierarchy` annotation on the entity type of the collection given by the `HierarchyNodes` parameter. This specifies a recursive hierarchy that is evaluated by the function.

[AvailableOnAggregatesType](#)

Property	Type	Description
RequiredProperties	[PropertyPath]	Properties required to apply this function

[NavigationPropertyAggregationCapabilities](#): [NavigationPropertyRestriction](#) (*Deprecated*)

[Capabilities.NavigationRestrictions](#) that make use of the additional properties in this subtype are deprecated in favor of [ApplySupported](#) and [CustomAggregate](#)

[CustomAggregateType](#) (*Deprecated*)

Deprecated since [NavigationPropertyAggregationCapabilities](#) is also deprecated

7 Authorization Vocabulary

Namespace: [Org.OData.Authorization.V1](#)

The Authorization Vocabulary provides terms for describing authorization requirements of the service

Terms

Term	Type	Description
SecuritySchemes	[SecurityScheme]	At least one of the specified security schemes are required to make a request against the service
Authorizations	[Authorization]	Lists the methods supported by the service to authorize access

[SecurityScheme](#)

Property	Type	Description
Authorization	SchemeName	The name of a required authorization scheme
RequiredScopes	[String]	The names of scopes required from this authorization scheme

[Authorization](#)

Base type for all Authorization types

Derived Types:

- [OpenIDConnect](#)
- [Http](#)
- [OAuthAuthorization](#)
 - [OAuth2ClientCredentials](#)
 - [OAuth2Implicit](#)
 - [OAuth2Password](#)
 - [OAuth2AuthCode](#)
- [ApiKey](#)

Property	Type	Description
Name	String	Name that can be used to reference the authorization scheme
Description	String?	Description of the authorization scheme

[OpenIDConnect: Authorization](#)

Property	Type	Description
Name	String	Name that can be used to reference the authorization scheme
Description	String?	Description of the authorization scheme

Property	Type	Description
IssuerUrl	URL	Issuer location for the OpenID Provider. Configuration information can be obtained by appending /.well-known/openid-configuration to this Url.

Http: Authorization

Property	Type	Description
Name	String	Name that can be used to reference the authorization scheme
Description	String?	Description of the authorization scheme
Scheme	String	HTTP Authorization scheme to be used in the Authorization header, as per RFC7235
BearerFormat	String?	Format of the bearer token

OAuthAuthorization: Authorization

Derived Types:

- [OAuth2ClientCredentials](#)
- [OAuth2Implicit](#)
- [OAuth2Password](#)
- [OAuth2AuthCode](#)

Property	Type	Description
Name	String	Name that can be used to reference the authorization scheme
Description	String?	Description of the authorization scheme
Scopes	[AuthorizationScope]	Available scopes
RefreshUrl	URL?	Refresh Url

OAuth2ClientCredentials: OAuthAuthorization

Property	Type	Description
Name	String	Name that can be used to reference the authorization scheme
Description	String?	Description of the authorization scheme
Scopes	[AuthorizationScope]	Available scopes
RefreshUrl	URL?	Refresh Url
TokenUrl	URL	Token Url

OAuth2Implicit: OAuthAuthorization

Security note: OAuth2 implicit grant is considered to be not secure and should not be used by clients, see [OAuth 2.0 Security Best Current Practice](#).

Property	Type	Description
<i>Name</i>	String	Name that can be used to reference the authorization scheme
<i>Description</i>	String?	Description of the authorization scheme
<i>Scopes</i>	[AuthorizationScope]	Available scopes
<i>RefreshUrl</i>	URL?	Refresh Url
<i>AuthorizationUrl</i>	URL	Authorization URL

OAuth2Password: OAuthAuthorization

Property	Type	Description
<i>Name</i>	String	Name that can be used to reference the authorization scheme
<i>Description</i>	String?	Description of the authorization scheme
<i>Scopes</i>	[AuthorizationScope]	Available scopes
<i>RefreshUrl</i>	URL?	Refresh Url
<i>TokenUrl</i>	URL	Token Url

OAuth2AuthCode: OAuthAuthorization

Property	Type	Description
<i>Name</i>	String	Name that can be used to reference the authorization scheme
<i>Description</i>	String?	Description of the authorization scheme
<i>Scopes</i>	[AuthorizationScope]	Available scopes
<i>RefreshUrl</i>	URL?	Refresh Url
<i>AuthorizationUrl</i>	URL	Authorization URL
<i>TokenUrl</i>	URL	Token Url

AuthorizationScope

Property	Type	Description
<i>Scope</i>	String	Scope name
<i>Grant</i>	String?	Identity that has access to the scope or can grant access to the scope.
<i>Description</i>	String	Description of the scope

ApiKey: Authorization

Property	Type	Description
Name	String	Name that can be used to reference the authorization scheme
Description	String?	Description of the authorization scheme
KeyName	String	The name of the header or query parameter
Location	KeyLocation	Whether the API Key is passed in the header or as a query option

KeyLocation

Member	Value	Description
Header	0	API Key is passed in the header
QueryOption	1	API Key is passed as a query option
Cookie	2	API Key is passed as a cookie

SchemeName

Type: String

The name of the authorization scheme.

8 JSON Vocabulary

Namespace: [Org.OData.JSON.V1](#)

Terms for JSON properties

Terms

Term	Type	Description
Schema	JSON	The JSON Schema for JSON values of the annotated media entity type, property, parameter, return type, term, or type definition The schema can be a schema reference, i.e. <code>{"\$ref": "url/of/schemafile#/path/to/schema/within/schemafile"}</code>

[JSON](#)

Type: Stream

Textual data of media type `application/json`

9 Repeatability Vocabulary

Namespace: [Org.OData.Repeatability.V1](#)

Terms describing repeatable requests

Terms

Term	Type	Description
Supported	Tag	Repeatable requests are supported for the annotated service, entity set, or action Annotations on entity set or action import level override an annotation on entity container level, and an annotation on action level override an annotation on action import level.
DeleteWithClientIDSupported	Tag	Deletion of remembered requests by client ID is supported Clients that specify a <code>RepeatabilityClientID</code> header can delete all remembered requests for that client ID by sending a <code>DELETE \$RepeatabilityRequestsWithClientID/{Repeatability-Client-ID}</code> request to the service root.
DeleteWithRequestIDSupported	Tag	Deletion of remembered requests by request ID is supported Clients can delete a single remembered request by sending a <code>DELETE \$RepeatabilityRequestWithRequestID/{Repeatability-Request-ID}</code> request to the service root.

10 Temporal Vocabulary

Namespace: [Org.OData.Temporal.V1](#)

Terms for describing time-dependent data

Terms

Term	Type	Description
ApplicationTimeSupport	ApplicationTimeSupportType	This collection supports temporal requests

Actions

[Update](#)

Updates existing time slices with values from delta time slices whose object keys match and whose periods overlap

The update behavior for a given object key is known from the [SQL statement](#) UPDATE ... FOR PORTION OF BUSINESS_TIME ... WHERE

deltaTimeslices with non-matching object keys and non-overlapping sub-periods of deltaTimeslices are disregarded.

Parameter	Type	Description
timeslices	[EntityType]	Binding parameter: Time slices to modify
deltaTimeslices	[TimesliceWithPeriod]	New time slices whose property values are used to update the <code>timeslices</code> collection The delta time slices need not contain all properties, but at least the boundary values of the period to change. An absent object key property matches any key property value. New time slices are processed in the order of the collection, which especially matters if some of the specified change periods overlap.
→	[TimesliceWithPeriod]	Modified time slices

[Upsert](#)

Like [Update](#), but additionally inserts those (sub-periods of) deltaTimeslices that Update disregards

Parameter	Type	Description
timeslices	[EntityType]	Binding parameter: Time slices to modify
deltaTimeslices	[TimesliceWithPeriod]	New time slices to be merged into the <code>timeslices</code> collection The delta time slices must contain all properties that are needed for insertion. New time slices are processed in the order of the collection, which especially matters if some of the specified change periods overlap.
→	[TimesliceWithPeriod]	Modified time slices

Delete

Deletes (sub-periods of) existing time slices whose object keys match and whose periods overlap `deltaTimeslices`

The deletion behavior for a given object key is known from the [SQL statement](#) `DELETE ... FOR PORTION OF BUSINESS_TIME ... WHERE ...`. The sub-period of an existing time slice that lies outside a given instance of `deltaTimeslices` is kept, effectively shortening the time slice.

Parameter	Type	Description
timeslices	[EntityType]	Binding parameter: Time slices to modify
deltaTimeslices	[TimesliceWithPeriod]	Time slices to be deleted from the <code>timeslices</code> collection The delta time slices contain only the boundary values of the period to delete and (parts of) the object key. An absent object key property matches any key property value.
→	[TimesliceWithPeriod]	Deleted (sub-periods of) time slices

ApplicationTimeSupportType

Property	Type	Description
UnitOfTime	UnitOfTime	Unit of time and other properties of a time period
Timeline	Timeline	Describes how the history and future of the data are represented
SupportedActions	[QualifiedActionName]	List of supported temporal actions

UnitOfTime

Unit of time and other properties of a time period

Derived Types:

- [UnitOfTimeDateTimeOffset](#)
- [UnitOfTimeDate](#)

[UnitOfTimeDateTimeOffset](#): [UnitOfTime](#)

Period start and end are of type `Edm.DateTimeOffset`

Property	Type	Description
Precision	Byte	Precision of <code>Edm.DateTimeOffset</code> values for period start and end

[UnitOfTimeDate](#): [UnitOfTime](#)

Period start and end are of type `Edm.Date`

The period is a contiguous set of days and does not consider the time of the day.

Property	Type	Description
ClosedClosedPeriods	Boolean	If <code>true</code> , the period end is the last day in the period; if <code>false</code> , the period end is the first day after the period

[Timeline](#)

Describes how the history and future of the data are represented

Derived Types:

- [TimelineSnapshot](#)
- [TimelineVisible](#)

[TimelineSnapshot: Timeline](#)

Each OData entity maps each point in application time to an instance of the entity type

To address an entity in a resource path or path to related resources, a point in application time must be specified as described in [OData-Temporal, section 4.2.1]. The addressed entity is then a snapshot of the data at the given point in time. When an action defined in this vocabulary is applied to a collection of this entity type, the entity key plays the role of object key.

[TimelineVisible: Timeline](#)

Each OData entity represents data during a period of application time

The temporal collection MUST NOT contain two entities with the same object key as defined by their `ObjectKey` properties and with overlapping application-time periods as defined by their `PeriodStart` and `PeriodEnd` properties. The temporal collection always contains all entities (with consecutive time periods) for a given object key.

Property	Type	Description
PeriodStart	PropertyPath	Property containing lower boundary of a period
PeriodEnd	PropertyPath	Property containing upper boundary of a period If an upper boundary property does not specify a default value, a default value of <code>max</code> is assumed.
ObjectKey	[PropertyPath]	The set of primitive properties that identify a temporal object A temporal object is a set of facts whose changes over application time are tracked by the service. The entities in the annotated collection belong to potentially multiple temporal objects, and each temporal object is uniquely identified by the values of the specified object key properties. Object key properties follow the same rules as entity key properties. If no object key is specified, only a single temporal object belongs to the annotated collection.

[TimesliceWithPeriod](#)

Delta time slices with validity period

The properties `PeriodStart` and `PeriodEnd` MUST NOT be present if the entity type of the `Timeslice` already contains period start and end, that is, if the collection on which the action is invoked has visible timeline. If present, they MUST have the same type, either `Edm.Date` or `Edm.DateTimeOffset`, and they are interpreted according to the [ApplicationTimeSupport/UnitOfTime](#) of the collection. In particular, `ApplicationTimeSupport/UnitOfTime/ClosedClosedPeriods` governs whether a `PeriodEnd` of type `Edm.Date` is the

last day in the period or the first day after the period. If `PeriodStart` is present and `PeriodEnd` is absent, a default value of `max` is assumed for `PeriodEnd`.

Property	Type	Description
PeriodStart	PrimitiveType?	Lower boundary of the time slice
PeriodEnd	PrimitiveType?	Upper boundary of the time slice
Timeslice	EntityType	A time slice with the same entity type as the binding parameter of the action When it appears in the return type of an action in this vocabulary, the time slice has the same entity set as the binding parameter value.

Appendix A. References

This appendix contains the normative and informative references that are used in this document.

While any hyperlinks included in this appendix were valid at the time of publication, OASIS cannot guarantee their long-term validity.

A.1 Normative References

The following documents are referenced in such a way that some or all of their content constitutes requirements of this document.

[OData-Protocol]

OData Version 4.02. Part 1: Protocol.

See link in "[Related work](#)" section on cover page.

[OData-CSDL]

OData Common Schema Definition Language (CSDL) JSON Representation Version 4.02.

See link in "[Related work](#)" section on cover page.

OData Common Schema Definition Language (CSDL) XML Representation Version 4.02.

See link in "[Related work](#)" section on cover page.

[RFC2119]

Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997. <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174]

Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017. <https://www.rfc-editor.org/info/rfc8174>.

A.2 Non-Normative References

[OData-Vocabs]

odata-vocabularies OASIS TC GitHub repository. <https://github.com/oasis-tcs/odata-vocabularies>.

Appendix B. Acknowledgments

B.1 Special Thanks

The contributions of the OASIS OData Technical Committee members, enumerated in [\[OData-Protocol\]](#), are gratefully acknowledged.

B.2 Participants

OData TC Members:

First Name	Last Name	Company
George	Ericson	Dell
Hubert	Heijkers	IBM
Ling	Jin	IBM
Stefan	Hagen	Individual
Michael	Pizzo	Microsoft
Christof	Sprenger	Microsoft
Ralf	Handl	SAP SE
Gerald	Krause	SAP SE
Heiko	Theißen	SAP SE
Martin	Zurmuehl	SAP SE

Appendix C. Revision History

Revision	Date	Editor	Changes Made
Committee Specification Draft 01	2016-12-08	Michael Pizzo Ralf Handl Ram Jeyaraman	Documentation automatically extracted from XML source files
Committee Specification Draft 02	2024-06-19	Michael Pizzo Ralf Handl Heiko Theißen	Documentation automatically extracted from XML source files

Appendix D. Notices

Copyright © OASIS Open 2024. All Rights Reserved.

All capitalized terms in the following text have the meanings assigned to them in the OASIS Intellectual Property Rights Policy (the “OASIS IPR Policy”). The full [Policy](#) may be found at the OASIS website.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published, and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this section are included on all such copies and derivative works. However, this document itself may not be modified in any way, including by removing the copyright notice or references to OASIS, except as needed for the purpose of developing any document or deliverable produced by an OASIS Technical Committee (in which case the rules applicable to copyrights, as set forth in the OASIS IPR Policy, must be followed) or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by OASIS or its successors or assigns.

This document and the information contained herein is provided on an “AS IS” basis and OASIS DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY OWNERSHIP RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

As stated in the OASIS IPR Policy, the following three paragraphs in brackets apply to OASIS Standards Final Deliverable documents (Committee Specification, Candidate OASIS Standard, OASIS Standard, or Approved Errata).

[OASIS requests that any OASIS Party or any other party that believes it has patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable, to notify OASIS TC Administrator and provide an indication of its willingness to grant patent licenses to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this deliverable.]

[OASIS invites any party to contact the OASIS TC Administrator if it is aware of a claim of ownership of any patent claims that would necessarily be infringed by implementations of this OASIS Standards Final Deliverable by a patent holder that is not willing to provide a license to such patent claims in a manner consistent with the IPR Mode of the OASIS Technical Committee that produced this OASIS Standards Final Deliverable. OASIS may include such claims on its website, but disclaims any obligation to do so.]

[OASIS takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this OASIS Standards Final Deliverable or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on OASIS' procedures with respect to rights in any document or deliverable produced by an OASIS Technical Committee can be found on the OASIS website. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this OASIS Standards Final Deliverable, can be obtained from the OASIS TC Administrator. OASIS makes no representation that any information or list of intellectual property rights will at any time be complete, or that any claims in such list are, in fact, Essential Claims.]

The name “OASIS” is a trademark of [OASIS](#), the owner and developer of this specification, and should be used only to refer to the organization and its official outputs. OASIS welcomes reference to, and implementation and use of, specifications, while reserving the right to enforce its marks against misleading uses. Please see <https://www.oasis-open.org/policies-guidelines/trademark/> for above guidance.