

# [MS-ODATAJSON]: OData Protocol JSON Format Standards Support Document

---

## Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft [Open Specification Promise](#) or the [Community Promise](#). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting [iplg@microsoft.com](mailto:iplg@microsoft.com).
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit [www.microsoft.com/trademarks](http://www.microsoft.com/trademarks).
- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

**Reservation of Rights.** All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

**Tools.** The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

## Revision Summary

| Date       | Revision History | Revision Class | Comments   |
|------------|------------------|----------------|--|
| 06/11/2013 | 1.0              | New            | Released new document.   |
| 08/08/2013 | 1.0              | No change      | No changes to the meaning, language, or formatting of the technical content. |
| 12/05/2013 | 1.0              | No change      | No changes to the meaning, language, or formatting of the technical content. |
| 02/11/2014 | 1.0              | No change      | No changes to the meaning, language, or formatting of the technical content. |

# Contents

|   |           |
|---|-----------|
| <b>1 Introduction</b>   | <b>4</b>  |
| 1.1 Glossary  | 4         |
| 1.2 References  | 4         |
| 1.2.1 Normative References  | 4         |
| 1.2.2 Informative References  | 4         |
| 1.3 Microsoft Implementations   | 4         |
| 1.4 Standards Support Requirements  | 4         |
| 1.5 Notation  | 5         |
| <b>2 Standards Support Statements</b>   | <b>6</b>  |
| 2.1 Normative Variations  | 6         |
| 2.1.1 Section 3.1.1 odata.metadata=minimal                                      | 6         |
| 2.1.2 Section 3.1.2 odata.metadata=full   | 6         |
| 2.1.3 Section 3.1.3 odata.metadata=none   | 6         |
| 2.1.4 Section 4.1 Header Content-Type   | 6         |
| 2.1.5 Section 4.2 Message Body  | 6         |
| 2.1.6 Section 4.3 Relative URLs   | 7         |
| 2.1.7 Section 4.5.1 odata.metadata  | 7         |
| 2.1.8 Section 4.5.3 Annotation odata.type                                       | 7         |
| 2.1.9 Section 4.5.6 Annotation odata.deltaLink                                  | 8         |
| 2.1.10 Section 4.5.8 Annotation odata.editLink and odata.readLink               | 8         |
| 2.1.11 Section 4.5.9 Annotation odata.kind                                      | 8         |
| 2.1.12 Section 4.5.11 Annotation odata.navigationLink and odata.associationLink | 8         |
| 2.1.13 Section 5 Service Document   | 9         |
| 2.1.14 Section 7.1 Primitive Value  | 9         |
| 2.1.15 Section 13 Resource Reference  | 9         |
| 2.1.16 Section 14 Delta Response  | 10        |
| 2.1.17 Section 17 Action Parameters   | 10        |
| 2.1.18 Section 19 Error Response  | 10        |
| 2.2 Clarifications  | 10        |
| 2.3 Error Handling  | 10        |
| 2.4 Security  | 10        |
| <b>3 Change Tracking</b>  | <b>11</b> |
| <b>4 Index</b>  | <b>12</b> |

# 1 Introduction

OData Protocol JSON Format Standards Support Document provides a statement of standards support for the Open Data (OData) protocol. This document describes the variations in the **JavaScript Object Notation (JSON)** format between the specified Microsoft implementations and the OASIS OData JSON format standards document.

The OData protocol comprises a set of specifications for representing and interacting with structured content. The core specifications for the OData protocol are specified in "Open Data Protocol (OData)" [[MS-ODATA](#)]. The OASIS "OData JSON Format Version 4.0" [[ODataJSON4.0](#)] standards document is an extension of the OData core protocol and defines representations for the OData requests and responses by using a JSON format.

## 1.1 Glossary

The following terms are defined in [[MS-ODATA](#)]:

**JavaScript Object Notation (JSON)**  
**property**  
**resource**

The following terms are specific to this document:

**MAY, SHOULD, MUST, SHOULD NOT, MUST NOT:** These terms (in all caps) are used as described in [[RFC2119](#)]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

## 1.2 References

### 1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com). We will assist you in finding the relevant information.

[ODataJSON4.0] OASIS, "OData JSON Format Version 4.0", OASIS Committee Specification Draft 01 / Public Review Draft 01, April 2013, <http://docs.oasis-open.org/odata/odata-json-format/v4.0/csprd01/odata-json-format-v4.0-csprd01.doc>

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

### 1.2.2 Informative References

[MS-ODATA] Microsoft Corporation, "[Open Data Protocol \(OData\)](#)".

## 1.3 Microsoft Implementations

Windows Server 2012 R2 operating system

## 1.4 Standards Support Requirements

An OData implementation that is fully compliant with the OData standard implements all mandatory features and optionally implements any optional features.

[[ODataJSON4.0](#)] defines conformance for the OData protocol. This conformance is defined as support for all features in the document except the following:

- Features that are called out as optional.

The following table lists the sections of [[ODataJSON4.0](#)] that are considered normative and that are considered informative.

| Section            | Normative/Informative |
|--------------------|-----------------------|
| 1 and 2            | Informative           |
| 3 through 21       | Normative             |
| Appendices A and B | Informative           |

## 1.5 Notation

The following notations are used to identify clarifications in section 2.2.

| Notation | Explanation  |
|----------|--|
| C####    | This notation identifies a clarification of ambiguity in the target specification. This includes imprecise statements, omitted information, discrepancies, and errata. This does not include data formatting clarifications. |
| V####    | This notation identifies an intended point of variability in the target specification, such as the use of MAY, SHOULD, or RECOMMENDED. This does not include extensibility points.   |
| E####    | Because the use of extensibility points (such as optional implementation-specific data) could impair interoperability, this notation identifies such points in the target specification.                                     |

## 2 Standards Support Statements

### 2.1 Normative Variations

The following subsections detail the normative variations from [\[ODataJSON4.0\]](#).

#### 2.1.1 Section 3.1.1 odata.metadata=minimal

The specification states the following:

```
The client MAY specify odata.metadata=minimal to indicate that the server SHOULD remove
computable control information from the payload wherever possible.
```

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. The parameter value is `odata=minimalmetadata`.

#### 2.1.2 Section 3.1.2 odata.metadata=full

The specification states the following:

```
The client MAY specify odata.metadata=full to include all control information explicitly
in the payload. The service MUST return all metadata in this case.
```

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. The parameter value is `odata=fullmetadata`.

#### 2.1.3 Section 3.1.3 odata.metadata=none

The specification states the following:

```
The client MAY specify odata.metadata=none in order to request that the service omit control
information.
```

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. The parameter value is `odata=nometadata`.

#### 2.1.4 Section 4.1 Header Content-Type

The specification states the following:

```
Responses MUST add the odata.metadata parameter with the same value that was specified
in the Accept header of the request. If no value was specified in the Accept header,
odata.metadata=minimal MUST be used.
```

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. The default value is `odata=minimalmetadata`.

#### 2.1.5 Section 4.2 Message Body

The specification states the following:

Each message body MUST be represented as a single JSON object. This object is either the representation of an entity, an entity reference or a complex type instance, or it contains a name/value pair ... .

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. Entity reference is not supported.

### 2.1.6 Section 4.3 Relative URLs

The specification states the following:

URLs present in a payload (whether request or response) MAY be represented as relative URLs to the metadata URL.

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. The annotation is `odata.navigationLinkUrl`. Example:

```
{
  "odata.metadata": "http://host/service/$metadata#Customers/@Element",
  ...
  "odata.editLink": "Customers('ALFKI')",
  ...
  "Orders@odata.navigationLinkUrl": "Customers('ALFKI')/Orders",
  ...
}
```

### 2.1.7 Section 4.5.1 `odata.metadata`

The specification states the following:

The `odata.metadata` annotation MUST also be included for entities whose entity set cannot be determined from the metadata URL of the collection. This URL MAY be absolute or relative to the metadata URL of the collection.

The `odata.metadata` annotation MUST also be applied to navigation links for navigation properties not described in the metadata of the containing type. In this case the metadata URL MAY be relative to the metadata URL describing the parent entity and becomes the root metadata URL for the related entity or collection.

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. The `odata.metadata` annotation is not applied to individual entities within a collection or to navigation links.

### 2.1.8 Section 4.5.3 Annotation `odata.type`

The specification states the following:

If the `odata.type` annotation is present, its value MUST be the namespace- or alias-qualified name of the instance's type, in which case the type MUST be defined by the root of the current metadata URL, otherwise it MUST be a full URL to a metadata document with the namespace- or alias-qualified name of the instance's type appended as a URL fragment.

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. The `odata.type` annotation is never a full URL to a metadata document and always references a type defined by the root of the metadata URL.

### **2.1.9 Section 4.5.6 Annotation `odata.deltaLink`**

The specification states the following:

```
The odata.deltaLink annotation contains a URL that can be used to retrieve changes to
the current set of results.
```

OData in the specified Microsoft implementations ([section 1.3](#)) does not support this feature.

### **2.1.10 Section 4.5.8 Annotation `odata.editLink` and `odata.readLink`**

The specification states the following:

```
The odata.editLink annotation contains a URL that can be used to read, update, or delete
the entity. It MUST appear for updatable entities if odata.metadata=full is requested or
if its value differs from the value of the odata.id.
```

```
The odata.readLink annotation contains a URL that can be used to read the entity. It MUST
appear if odata.metadata=full is requested and its value differs from the value of the
odata.id and the odata.editlink is not present.
```

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. The `editlink` is always written in full metadata, and the `readlink` is never written.

### **2.1.11 Section 4.5.9 Annotation `odata.kind`**

The specification states the following:

```
The odata.kind annotation is used to differentiate the kind of entry represented by
the JSON object ... .
```

OData in the specified Microsoft implementations ([section 1.3](#)) does not support this feature.

### **2.1.12 Section 4.5.11 Annotation `odata.navigationLink` and `odata.associationLink`**

The specification states the following:

```
The odata.navigationLink annotation contains a URL that can be used to retrieve an entity or
collection of entities related to the current entity via a navigation property.
```

```
The odata.associationLink annotation contains a URL that can be used to retrieve a reference
to an entity or a collection of references to entities related to the current entity via a
navigation property.
```

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. The annotations are `odata.navigationLinkUrl` and `odata.associationLinkUrl`.



### 2.1.13 Section 5 Service Document

The specification states the following:

JSON objects representing an entity set MAY contain an additional name/value pair with name kind and a value of EntitySet.

JSON objects representing a function or action import MUST contain this name/value pair with a value of FunctionImport or ActionImport, respectively.

JSON objects representing a named entity MUST contain this name/value pair with a value of Entity.

JSON objects representing a related service document MUST contain this name/value pair with a value of ServiceDocument.

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. Only JSON objects representing an entity set are supported, and the kind **property** is not supported on objects within the representation of a Service Document.

### 2.1.14 Section 7.1 Primitive Value

The specification states the following:

Values of type Edm.Binary, Edm.Date, Edm.DateTimeOffset, Edm.Duration, Edm.Guid, and Edm.TimeOfDay as well as enumeration values are represented as JSON strings whose content satisfies the rules binaryValue, dateValue, dateTimeOffsetValue, durationValue, guidValue, timeOfDayValue, and enumValue, respectively, in [OData-ABNF].

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. The Date, Duration, and TimeOfDay primitive types, as well as enumeration values, are not supported. The following additional primitive types, which are not defined in the specification, are supported:

- Time
- Float
- DateTime

### 2.1.15 Section 13 Resource Reference

The specification states the following:

A resource reference is a reference to an entity or a property of an entity. Resource references referring to an entity are called entity references.

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. **Resource** reference is not supported. Links are represented by using a name/value pair named "url", as shown in the following example:

```
{
  "url": http://host/service/Orders(10643)
}
```

### **2.1.16 Section 14 Delta Response**

The specification states the following:

`Responses from a delta request are returned as a JSON object.`

OData in the specified Microsoft implementations ([section 1.3](#)) does not support the representation of Delta Responses.

### **2.1.17 Section 17 Action Parameters**

The specification states the following:

`Action parameter values MUST be encoded in a single JSON object in the request body.`

OData in the specified Microsoft implementations ([section 1.3](#)) does not support this feature.

### **2.1.18 Section 19 Error Response**

The specification states the following:

`This object MUST contain name/value pairs with the names code and message, and it MAY contain name/value pairs with the names target, details and innererror.`

OData in the specified Microsoft implementations ([section 1.3](#)) partially supports this feature. Name/value pairs for target and details are not supported.

## **2.2 Clarifications**

None.

## **2.3 Error Handling**

None.

## **2.4 Security**

None

### 3 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

## 4 Index

### C

[Change tracking](#) 11  
[Clarifications](#) 10

### E

[Error handling](#) 10

### G

[Glossary](#) 4

### I

[Implementations](#) 4  
[Informative references](#) 4  
[Introduction](#) 4

### M

[Microsoft implementations](#) 4

### N

[Normative references](#) 4  
[Normative variations](#) 6  
[Notation](#) 5

### R

References  
[informative](#) 4  
[normative](#) 4

### S

[Security](#) 10  
[Standards support requirements](#) 4

### T

[Tracking changes](#) 11

### V

Variations  
[normative](#) 6