[MS-OEXTXML]:

Office Shared Extensibility in Office Open XML

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation ("this documentation") for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.
- **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 can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.
- **No Trade Secrets**. Microsoft does not claim any trade secret rights in this documentation.
- **Patents**. Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft Open Specifications Promise or the Microsoft Community Promise. If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **License Programs**. To see all of the protocols in scope under a specific license program and the associated patents, visit the Patent Map.
- **Trademarks**. The names of companies and products contained in this documentation might 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.
- **Fictitious Names**. The example companies, organizations, products, domain names, email addresses, logos, people, places, and events that are 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 as specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications documentation does 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 documents are intended for use in conjunction with publicly available standards specifications and network programming art and, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

Support. For questions and support, please contact dochelp@microsoft.com.

Preliminary Documentation. This particular Open Specifications document provides documentation for past and current releases and/or for the pre-release version of this technology. This document provides final documentation for past and current releases and preliminary documentation, as applicable and specifically noted in this document, for the pre-release version. Microsoft will release final documentation in connection with the commercial release of the updated or new version of this technology. Because this documentation might change between the pre-release version and the final

Preliminary

version of this technology, there are risks in relying on this preliminary documentation. To the extent that you incur additional development obligations or any other costs as a result of relying on this preliminary documentation, you do so at your own risk.

Preliminary

Copyright © 2021 Microsoft Corporation

Revision Summary

Date	Revision History	Revision Class	Comments
12/7/2020	1.0	New	Released new document.
4/22/2021	2.0	Major	Significantly changed the technical content.

Preliminary

Copyright © 2021 Microsoft Corporation

Table of Contents

Preliminary

Copyright © 2021 Microsoft Corporation

1 Introduction

This document specifies complex types for representing extension lists in the context of Open XML file formats described in [ISO/IEC29500-1:2016]. The new types are presented using the extensibility mechanisms described in [ISO/IEC29500-3:2015].

Sections 1.7 and 2 of this specification are normative. All other sections and examples in this specification are informative.

1.1 Glossary

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

1.2 References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the Errata.

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. We will assist you in finding the relevant information.

[ISO/IEC29500-1:2016] ISO/IEC, "Information technology -- Document description and processing languages -- Office Open XML File Formats -- Part 1: Fundamentals and Markup Language Reference", ISO/IEC 29500-1:2016, https://www.iso.org/standard/71691.html

[ISO/IEC29500-2:2012] ISO/IEC, "Information technology -- Document description and processing languages -- Office Open XML File Formats -- Part 2: Open Packaging Conventions", ISO/IEC 29500-2:2012, http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=61796

[ISO/IEC29500-3:2015] ISO/IEC, "Information technology -- Document description and processing languages -- Office Open XML File Formats -- Part 3: Markup Compatibility and Extensibility", https://www.iso.org/standard/65533.html

[ISO/IEC29500-4:2016] ISO/IEC, "Information technology -- Document description and processing languages -- Office Open XML File Formats -- Part 4: Transitional Migration Features", https://www.iso.org/standard/71692.html

[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

[XMLSCHEMA1/2] Thompson, H., Beech, D., Maloney, M., and Mendelsohn, N., Eds., "XML Schema Part 1: Structures Second Edition", W3C Recommendation, October 2004, http://www.w3.org/TR/2004/REC-xmlschema-1-20041028/

[XMLSCHEMA2] Biron, P.V., Ed. and Malhotra, A., Ed., "XML Schema Part 2: Datatypes", W3C Recommendation, May 2001, http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/

1.2.2 Informative References

None.

Preliminary

Copyright © 2021 Microsoft Corporation

1.3 Overview

The complex types specified in this document represent extension lists. Extension lists provide a convention for extending a file format at wherever an extension list has been predefined.

The types can be used to extend the Office Open XML file formats as described in [ISO/IEC29500-1:2016] and this document. See section 3 for examples of using these types in markup specifications.

1.4 Relationship to Protocols and Other Structures

This specification is dependent on the structures and concepts defined in the following references:

- [ISO/IEC29500-1:2016] for baseline Open XML file formats.
- [ISO/IEC29500-2:2012] for open packaging conventions.
- [ISO/IEC29500-3:2015] for markup compatibility and extensibility.
- [ISO/IEC29500-4:2016] for backwards compatibility considerations.

1.5 Applicability Statement

This document specifies complex types for representing extension lists. These types can be used by extensions to the Office Open XML file formats ([ISO/IEC29500-1:2016]). The types specified in this document are not applicable as a stand-alone file format.

1.6 Versioning and Localization

None.

1.7 Vendor-Extensible Fields

None.

Preliminary

Copyright © 2021 Microsoft Corporation

2 Structures

2.1 http://schemas.microsoft.com/office/2019/extlst

2.1.1 Elements

None.

2.1.2 Attributes

None.

2.1.3 Complex Types

2.1.3.1 CT_Extension

Target namespace: http://schemas.microsoft.com/office/2019/extlst

Referenced by: CT_ExtensionList

A complex type that specifies an extension within an extension list.

See CT_ExtensionList for more details.

Attributes:

uri: A xsd:token ([XMLSCHEMA2] section 3.3.2) attribute that specifies a unique identifier for the extension.

The following W3C XML Schema ([XMLSCHEMA1/2] section 2.1) fragment specifies the contents of this complex type.

```
<xsd:complexType name="CT_Extension">
  <xsd:sequence>
    <xsd:any processContents="lax"/>
  </xsd:sequence>
  <xsd:attribute name="uri" type="xsd:token"/>
</xsd:complexType>
```

See section 5.1 for the full W3C XML Schema ([XMLSCHEMA1/2] section 2.1).

2.1.3.2 CT_ExtensionList

Target namespace: http://schemas.microsoft.com/office/2019/extlst

A complex type that specifies an extension list.

An extension list denotes a predefined place in a markup specification that allows for future extensibility. Future extensibility is possible because the extensions within an extension list can reference future markup specifications.

Any number of extensions are allowed within an extension list and the extensions are allowed to appear in any order.

When an extension list is processed, a consumer might not understand all of the extensions. A consumer determines whether it can understand an extension based on the **uri** of the extension

Copyright © 2021 Microsoft Corporation

Preliminary

(CT_Extension). If a consumer does not understand an extension, it MUST NOT attempt to interpret the contents. Instead, it MUST preserve the extension unless some ancestor of the extension list has been discarded. The extension is preserved as-is, excepting any namespace mapping that is necessary to remain compliant with the XML standard.

Markup namespaces within extensions are not required to be listed in the **Ignorable** Compatibility-Rule attribute ([ISO/IEC29500-3:2015] section 7.2).

Child Elements:

ext: A CT_Extension element that specifies an extension in the extension list.

The following W3C XML Schema ([XMLSCHEMA1/2] section 2.1) fragment specifies the contents of this complex type.

```
<xsd:complexType name="CT_ExtensionList">
  <xsd:sequence>
    <xsd:element name="ext" type="CT_Extension" minOccurs="0" maxOccurs="unbounded"/>
  </xsd:sequence>
</xsd:complexType>
```

See section 5.1 for the full W3C XML Schema ([XMLSCHEMA1/2] section 2.1).

2.1.4 Simple Types

None.

Preliminary

3 Structure Examples

3.1 Using Extension List Types in a Markup Specification

This example shows how to use the extension list types of this document in a markup specification.

ContosoWidgets is a hypothetical markup specification that extends the Office Open XML file formats ([ISO/IEC29500-1:2016]) by allowing the definition of widgets within a document.

When ContosoWidgets was originally defined, it was known that each widget had a name and id. Thus, the CT_ContosoWidget type has attributes for both.

Additionally, it was known that there was a possibility that a widget might have more than just a name and id in the future. To account for this, CT_ContosoWidget is extensible via an extension list child element.

For that extension list element, CT_ContosoWidgets uses the CT_ExtensionList type defined in this document. The element conforms to the extension list behavior detailed in this document.

```
<xsd:schema xmlns="http://www.example.com/ContosoWidgets"
targetNamespace="http://www.example.com/ContosoWidgets" elementFormDefault="qualified"
attributeFormDefault="unqualified" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:oel="http://schemas.microsoft.com/office/2019/extlst">
  <xsd:import id="oel" namespace="http://schemas.microsoft.com/office/2019/extlst"
schemaLocation="officeextlst.xsd"/>
  <xsd:element name="Widgets" type="CT_ContosoWidgets"/>
  <xsd:complexType name="CT ContosoWidgets">
    <xsd:sequence>
      <xsd:element name="Widget" type="CT_ContosoWidget" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="CT_ContosoWidget">
    <xsd:sequence>
      <xsd:element name="extLst" minOccurs="0" maxOccurs="1" type="oel:CT_ExtensionList"/>
    </xsd:sequence>
    <xsd:attribute name="name" type="xsd:string" use="required"/>
    <xsd:attribute name="id" type="xsd:int" use="required"/>
  </xsd:complexType>
</xsd:schema>
```

3.2 Using a New Part and Extension List Types to Extend a Previously Defined Markup Specification

This example shows how to extend a previously defined markup specification by using a new part and make the new part extensible with the extension list types of this document.

ContosoGizmos is a hypothetical markup specification that extends the Office Open XML file formats ([ISO/IEC29500-1:2016]) by allowing the definition of gizmos within a document.

Below is the initial version of ContosoGizmos. Each gizmo has a name and a unique id.

```
<xsd:schema xmlns="http://www.example.com/ContosoGizmos"
targetNamespace="http://www.example.com/ContosoGizmos" elementFormDefault="qualified"
attributeFormDefault="unqualified" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:element name="Gizmos" type="CT_ContosoGizmos"/>
  <xsd:complexType name="CT ContosoGizmos">
    <xsd:sequence>
      <xsd:element name="Gizmo" type="CT_ContosoGizmo" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
```


Copyright © 2021 Microsoft Corporation

```
  </xsd:complexType>
  <xsd:complexType name="CT_ContosoGizmo">
    <xsd:attribute name="name" type="xsd:string" use="required"/>
    <xsd:attribute name="id" type="xsd:int" use="required"/>
  </xsd:complexType>
</xsd:schema>
```

Later, a need arises to specify the size of each gizmo.

One option for addressing this need is to update Gizmos. However, existing readers would be unable to fully understand the new version since the elements did not exist at the time of their implementation. Depending on how exactly Gizmos was updated, those readers would either err while processing the new version or fail to preserve the new content.

An option that avoids those issues is to create a new XML part with the size data, mapping it to a gizmo by id. Many existing readers, such as Microsoft Office, preserve unknown parts that are related to known parts (see [ISO/IEC29500-1:2016] section 9.1). Thus, these existing readers will preserve an added part that is unknown to them but related to a known part.

In this example scenario, there has already been a need to specify more data for gizmos. To handle the possibility of specifying even more data in the future, one option is to make the new part extensible with extension lists. Any future new data can be specified in an extension.

Below is an example schema for the new part ContosoGizmosExtended. It is a list of the additional data for each gizmo, mapped by id. The additional data includes the size and extension list.

```
<xsd:schema xmlns="http://www.example.com/ContosoGizmosExtended"
targetNamespace="http://www.example.com/ContosoGizmosExtended" elementFormDefault="qualified"
attributeFormDefault="unqualified" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:oel="http://schemas.microsoft.com/office/2019/extlst">
  <xsd:import id="oel" namespace="http://schemas.microsoft.com/office/2019/extlst"
schemaLocation="officeextlst.xsd"/>
  <xsd:element name="GizmosExtended" type="CT_ContosoGizmosExtended"/>
  <xsd:complexType name="CT_ContosoGizmosExtended">
    <xsd:sequence>
      <xsd:element name="GizmoExtended" type="CT_ContosoGizmoExtended" minOccurs="0"
maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:complexType name="CT_ContosoGizmoExtended">
    <xsd:sequence>
      <xsd:element name="Size" type="CT_ContosoSize" minOccurs="0" maxOccurs="1"/>
      <xsd:element name="extLst" type="oel:CT_ExtensionList" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute name="id" type="xsd:int" use="required"/>
  </xsd:complexType>
  <xsd:complexType name="CT ContosoSize">
    <xsd:attribute name="width" type="xsd:int" use="required"/>
    <xsd:attribute name="height" type="xsd:int" use="required"/>
  </xsd:complexType>
</xsd:schema>
```


Copyright © 2021 Microsoft Corporation

4 Security

4.1 Security Considerations for Implementers

None.

4.2 Index of Security Fields

None.

Preliminary

Copyright © 2021 Microsoft Corporation

5 Appendix A: Full XML Schemas

Schema name	Prefix	Section
http://schemas.microsoft.com/office/2019/extlst Schema	None.	5.1

5.1 http://schemas.microsoft.com/office/2019/extlst Schema

```
<xsd:schema xmlns="http://schemas.microsoft.com/office/2019/extlst"
targetNamespace="http://schemas.microsoft.com/office/2019/extlst"
elementFormDefault="qualified" attributeFormDefault="unqualified"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <xsd:complexType name="CT Extension">
    <xsd:sequence>
      <xsd:any processContents="lax"/>
    </xsd:sequence>
    <xsd:attribute name="uri" type="xsd:token"/>
  </xsd:complexType>
  <xsd:complexType name="CT ExtensionList">
    <xsd:sequence>
      <xsd:element name="ext" type="CT_Extension" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:schema>
```

Prelim

Copyright © 2021 Microsoft Corporation

6 Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include updates to those products.

- Microsoft Office 2019
- Microsoft Office 2021

Exceptions, if any, are noted in this section. If an update version, service pack or Knowledge Base (KB) number appears with a product name, the behavior changed in that update. The new behavior also applies to subsequent updates unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms "SHOULD" or "SHOULD NOT" implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term "MAY" implies that the product does not follow the prescription.

Preliminary

Copyright © 2021 Microsoft Corporation

7 Change Tracking

This section identifies changes that were made to this document since the last release. Changes are classified as Major, Minor, or None.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements.
- A document revision that captures changes to protocol functionality.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **None** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the relevant technical content is identical to the last released version.

The changes made to this document are listed in the following table. For more information, please contact dochelp@microsoft.com.

Section	Description	Revision class
3.1 Using Extension List Types in a Markup Specification	Updated the property name in the specification.	Minor
3.2 Using a New Part and Extension List Types to Extend a Previously Defined Markup Specification	Updated the property name in the specification.	Minor
6 Appendix B: Product Behavior	Added the supported product.	Minor
6 Appendix B: Product Behavior	Updated list of supported products.	major

Preliminary

Copyright © 2021 Microsoft Corporation

8 Index

Preliminary

Copyright © 2021 Microsoft Corporation