[MS-EMMCSOM]:

Enterprise Managed Metadata Client-Side Object Model Protocol

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](http://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.
## Revision Summary

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision History</th>
<th>Revision Class</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>1/20/2012</td>
<td>0.1</td>
<td>New</td>
<td>Released new document.</td>
</tr>
<tr>
<td>4/11/2012</td>
<td>0.1</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>7/16/2012</td>
<td>0.1</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>9/12/2012</td>
<td>0.1</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>10/8/2012</td>
<td>1.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>2/11/2013</td>
<td>1.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>7/30/2013</td>
<td>1.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>11/18/2013</td>
<td>1.1</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>2/10/2014</td>
<td>1.1</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>4/30/2014</td>
<td>1.2</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>7/31/2014</td>
<td>1.2</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>10/30/2014</td>
<td>1.2</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>2/26/2016</td>
<td>2.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>7/15/2016</td>
<td>2.0</td>
<td>None</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>10/13/2016</td>
<td>3.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>6/5/2017</td>
<td>4.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>9/19/2017</td>
<td>4.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>12/12/2017</td>
<td>5.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>7/24/2018</td>
<td>6.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>10/1/2018</td>
<td>7.0</td>
<td>Major</td>
<td>Significantly changed the technical content.</td>
</tr>
<tr>
<td>12/11/2018</td>
<td>7.1</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
</tbody>
</table>
Table of Contents

1 Introduction .................................................................................................................. 14
  1.1 Glossary .................................................................................................................. 14
  1.2 References ............................................................................................................. 17
    1.2.1 Normative References ...................................................................................... 17
    1.2.2 Informative References .................................................................................... 17
  1.3 Overview ................................................................................................................. 17
  1.4 Relationship to Other Protocols ............................................................................ 17
  1.5 Prerequisites/Preconditions .................................................................................... 18
  1.6 Applicability Statement ......................................................................................... 18
  1.7 Versioning and Capability Negotiation ................................................................. 18
  1.8 Vendor-Extensible Fields ....................................................................................... 18
  1.9 Standards Assignments .......................................................................................... 18

2 Messages ..................................................................................................................... 19
  2.1 Transport ................................................................................................................ 19
  2.2 Message Syntax ...................................................................................................... 19

3 Protocol Details ......................................................................................................... 20
  3.1 Server Details ....................................................................................................... 20
    3.1.1 Abstract Data Model ....................................................................................... 20
    3.1.2 Timers ............................................................................................................. 20
    3.1.3 Initialization ..................................................................................................... 20
    3.1.4 Higher-Layer Triggered Events and Sequencing Rules ............................... 20
    3.1.5 Message Processing Events and Sequencing Rules ................................. 20
      3.1.5.1 Microsoft.SharePoint.Client.Taxonomy.ChangedGroup ................... 20
        3.1.5.1.1 Properties ......................................................................................... 21
          3.1.5.1.1.1 Scalar Properties ....................................................................... 21
            3.1.5.1.1.1.1 ChangedBy ........................................................................... 21
            3.1.5.1.1.1.2 ChangedTime ...................................................................... 21
            3.1.5.1.1.1.3 Id .......................................................................................... 21
            3.1.5.1.1.1.4 ItemType ............................................................................ 21
            3.1.5.1.1.1.5 Operation ............................................................................. 21
          3.1.5.1.1.2 ObjectPath Properties ............................................................... 21
        3.1.5.1.2 Methods ............................................................................................... 21
          3.1.5.1.2.1 Scalar Methods ....................................................................... 21
          3.1.5.1.2.2 ObjectPath Methods ............................................................... 22
      3.1.5.2 Microsoft.SharePoint.Client.Taxonomy.ChangedItem .................... 22
        3.1.5.2.1 Properties ......................................................................................... 22
          3.1.5.2.1.1 Scalar Properties ....................................................................... 22
            3.1.5.2.1.1.1 ChangedBy ........................................................................... 22
            3.1.5.2.1.1.2 ChangedTime ...................................................................... 22
            3.1.5.2.1.1.3 Id .......................................................................................... 22
            3.1.5.2.1.1.4 ItemType ............................................................................ 22
            3.1.5.2.1.1.5 Operation ............................................................................. 22
          3.1.5.2.1.2 ObjectPath Properties ............................................................... 23
        3.1.5.2.2 Methods ............................................................................................... 23
          3.1.5.2.2.1 Scalar Methods ....................................................................... 23
          3.1.5.2.2.2 ObjectPath Methods ............................................................... 23
      3.1.5.3 Microsoft.SharePoint.Client.Taxonomy.ChangedItemCollection .......... 23
        3.1.5.3.1 Properties ......................................................................................... 23
          3.1.5.3.1.1 Scalar Properties ....................................................................... 23
          3.1.5.3.1.2 ObjectPath Properties ............................................................... 23
        3.1.5.3.2 Methods ............................................................................................... 23
          3.1.5.3.2.1 Scalar Methods ....................................................................... 23
          3.1.5.3.2.2 ObjectPath Methods ............................................................... 23
3.1.5.4 Microsoft.SharePoint.Client.Taxonomy.ChangedItemType .......................... 23
  3.1.5.4.1 Field Values .................................................................................. 23
    3.1.5.4.1.1 Unknown ................................................................................. 24
    3.1.5.4.1.2 Term ....................................................................................... 24
    3.1.5.4.1.3 TermSet .................................................................................. 24
    3.1.5.4.1.4 Group ..................................................................................... 24
    3.1.5.4.1.5 TermStore ............................................................................... 24
    3.1.5.4.1.6 Site ........................................................................................ 24
  3.1.5.5 Microsoft.SharePoint.Client.Taxonomy.ChangedOperationType .............. 24
    3.1.5.5.1 Field Values ............................................................................... 24
      3.1.5.5.1.1 Unknown .............................................................................. 24
      3.1.5.5.1.2 Add ...................................................................................... 24
      3.1.5.5.1.3 Edit ....................................................................................... 25
      3.1.5.5.1.4 DeleteObject ....................................................................... 25
      3.1.5.5.1.5 Move ..................................................................................... 25
      3.1.5.5.1.6 Copy ...................................................................................... 25
      3.1.5.5.1.7 PathChange ......................................................................... 25
      3.1.5.5.1.8 Merge .................................................................................... 25
      3.1.5.5.1.9 ImportObject ....................................................................... 25
      3.1.5.5.1.10 Restore .............................................................................. 25
  3.1.5.6 Microsoft.SharePoint.Client.Taxonomy.ChangedSite ............................. 25
    3.1.5.6.1 Properties ............................................................................... 26
      3.1.5.6.1.1 Scalar Properties ................................................................. 26
        3.1.5.6.1.1.1 ChangedBy ................................................................. 26
        3.1.5.6.1.1.2 ChangedTime ............................................................ 26
        3.1.5.6.1.1.3 Id ............................................................................... 26
        3.1.5.6.1.1.4 ItemType ................................................................... 26
        3.1.5.6.1.1.5 Operation .................................................................. 26
        3.1.5.6.1.1.6 SiteId ........................................................................ 26
        3.1.5.6.1.1.7 TermId ........................................................................ 26
        3.1.5.6.1.1.8 TermSetId ................................................................... 27
      3.1.5.6.1.2 ObjectPath Properties ......................................................... 27
    3.1.5.6.2 Methods ................................................................................... 27
      3.1.5.6.2.1 Scalar Methods ................................................................. 27
      3.1.5.6.2.2 ObjectPath Methods .......................................................... 27
  3.1.5.7 Microsoft.SharePoint.Client.Taxonomy.ChangedTerm .......................... 27
    3.1.5.7.1 Properties ............................................................................... 27
      3.1.5.7.1.1 Scalar Properties ................................................................. 27
        3.1.5.7.1.1.1 ChangedBy ................................................................. 27
        3.1.5.7.1.1.2 ChangedTime ............................................................ 27
        3.1.5.7.1.1.3 GroupId ....................................................................... 28
        3.1.5.7.1.1.4 Id ............................................................................... 28
        3.1.5.7.1.1.5 ItemType ................................................................... 28
        3.1.5.7.1.1.6 Operation .................................................................. 28
        3.1.5.7.1.1.7 TermSetId ................................................................... 28
        3.1.5.7.1.1.8 ChangedCustomProperties ............................................. 28
        3.1.5.7.1.1.9 ChangedLocalCustomProperties .................................. 28
        3.1.5.7.1.1.10 LcidsForChangedDescriptions ...................................... 28
        3.1.5.7.1.1.11 LcidsForChangedLabels ................................................. 29
      3.1.5.7.1.2 ObjectPath Properties ......................................................... 29
    3.1.5.7.2 Methods ................................................................................... 29
      3.1.5.7.2.1 Scalar Methods ................................................................. 29
      3.1.5.7.2.2 ObjectPath Methods .......................................................... 29
  3.1.5.8 Microsoft.SharePoint.Client.Taxonomy.ChangedTermSet ..................... 29
    3.1.5.8.1 Properties ............................................................................... 29
      3.1.5.8.1.1 Scalar Properties ................................................................. 29
        3.1.5.8.1.1.1 ChangedBy ................................................................. 29
        3.1.5.8.1.1.2 ChangedTime ............................................................ 29

3.1.5.8.1.1.3 FromGroupId .................................................. 30
3.1.5.8.1.1.4 GroupId .................................................... 30
3.1.5.8.1.1.5 Id .......................................................... 30
3.1.5.8.1.1.6 ItemType .................................................. 30
3.1.5.8.1.1.7 Operation .................................................. 30
3.1.5.8.1.2 ObjectPath Properties ...................................... 30
3.1.5.8.2 Methods .......................................................... 30
3.1.5.8.2.1 Scalar Methods .............................................. 30
3.1.5.8.2.2 ObjectPath Methods .................................... 30
3.1.5.9 Microsoft.SharePoint.Client.Taxonomy.ChangedTermStore ........................................ 31
3.1.5.9.1 Properties ....................................................... 31
3.1.5.9.1.1 Scalar Properties ......................................... 31
3.1.5.9.1.1.1 ChangedBy ............................................... 31
3.1.5.9.1.1.2 ChangedLanguage ....................................... 31
3.1.5.9.1.1.3 ChangedTime ............................................ 31
3.1.5.9.1.1.4 Id .......................................................... 31
3.1.5.9.1.1.5 IsDefaultLanguageChanged .......................... 31
3.1.5.9.1.1.6 IsFullFarmRestore ..................................... 31
3.1.5.9.1.1.7 ItemType .................................................. 32
3.1.5.9.1.1.8 Operation ................................................ 32
3.1.5.9.1.2 ObjectPath Properties .................................... 32
3.1.5.9.2 Methods .......................................................... 32
3.1.5.9.2.1 Scalar Methods .............................................. 32
3.1.5.9.2.2 ObjectPath Methods .................................... 32
3.1.5.10 Microsoft.SharePoint.Client.Taxonomy.Label .................................................. 32
3.1.5.10.1 Properties ....................................................... 32
3.1.5.10.1.1 Scalar Properties ......................................... 32
3.1.5.10.1.1.1 IsDefaultForLanguage ................................ 32
3.1.5.10.1.1.2 Language ................................................ 33
3.1.5.10.1.1.3 Value ..................................................... 33
3.1.5.10.1.2 ObjectPath Properties .................................... 34
3.1.5.10.1.2.1 Term ...................................................... 34
3.1.5.10.2 Methods .......................................................... 34
3.1.5.10.2.1 Scalar Methods .............................................. 34
3.1.5.10.2.1.1 DeleteObject ............................................ 34
3.1.5.10.2.1.2 SetAsDefaultForLanguage .......................... 34
3.1.5.10.2.2 ObjectPath Methods .................................... 35
3.1.5.11 Microsoft.SharePoint.Client.Taxonomy.LabelCollection ........................................ 35
3.1.5.11.1 Properties ....................................................... 35
3.1.5.11.1.1 Scalar Properties ......................................... 35
3.1.5.11.1.2 ObjectPath Properties .................................... 35
3.1.5.11.2 Methods .......................................................... 35
3.1.5.11.2.1 Scalar Methods .............................................. 35
3.1.5.11.2.2 ObjectPath Methods .................................... 35
3.1.5.11.2.2.1 GetByValue ............................................. 35
3.1.5.12 Microsoft.SharePoint.Client.Taxonomy.StringMatchOption ........................................ 35
3.1.5.12.1 Field Values .................................................. 35
3.1.5.12.1.1 StartsWith ................................................ 35
3.1.5.12.1.2 ExactMatch ............................................... 36
3.1.5.13 Microsoft.SharePoint.Client.Taxonomy.TaxonomyFieldValue .................................... 36
3.1.5.13.1 Properties ....................................................... 36
3.1.5.13.1.1 Scalar Properties ......................................... 36
3.1.5.13.1.1.1 Label ....................................................... 36
3.1.5.13.1.1.2 TermGuid ................................................ 36
3.1.5.13.1.1.3 WssId ..................................................... 36
3.1.5.13.1.2 ObjectPath Properties .................................... 37
3.1.5.14 Microsoft.SharePoint.Client.Taxonomy.TaxonomyItem ........................................ 37
3.1.5.14.1 Properties ....................................................... 37
3.1.5.14.1.1 Scalar Properties ................................................................. 37
  3.1.5.14.1.1.1 CreatedDate ............................................................... 37
  3.1.5.14.1.1.2 Id ........................................................................ 37
  3.1.5.14.1.1.3 LastModifiedDate ....................................................... 37
  3.1.5.14.1.1.4 Name ...................................................................... 37
3.1.5.14.1.2 ObjectPath Properties ....................................................... 38
  3.1.5.14.1.2.1 TermStore ................................................................. 38
3.1.5.14.2 Methods ........................................................................ 38
  3.1.5.14.2.1 Scalar Methods ............................................................. 38
    3.1.5.14.2.1.1 DeleteObject ............................................................. 38
    3.1.5.14.2.1.2 NormalizeName ....................................................... 38
  3.1.5.14.2.2 ObjectPath Methods ....................................................... 38
3.1.5.15 Microsoft.SharePoint.Client.Taxonomy.TaxonomySession .................. 38
  3.1.5.15.1 Properties ..................................................................... 38
    3.1.5.15.1.1 Scalar Properties ......................................................... 38
      3.1.5.15.1.1.1 OfflineTermStoreNames ............................................ 39
    3.1.5.15.1.2 ObjectPath Properties .................................................. 39
      3.1.5.15.1.2.1 TermStores ........................................................... 39
  3.1.5.15.2 Methods ...................................................................... 39
    3.1.5.15.2.1 Scalar Methods .......................................................... 39
      3.1.5.15.2.1.1 UpdateCache ......................................................... 39
    3.1.5.15.2.2 ObjectPath Methods .................................................... 39
      3.1.5.15.2.2.1 GetTerm ................................................................. 39
      3.1.5.15.2.2.2 GetTerms .............................................................. 40
      3.1.5.15.2.2.3 GetTermsWithCustomProperty ................................. 41
      3.1.5.15.2.2.4 GetTermsInDefaultLanguage ................................... 43
      3.1.5.15.2.2.5 GetTermsInWorkingLocale ....................................... 44
      3.1.5.15.2.2.6 GetDefaultKeywordsTermStore ................................ 44
      3.1.5.15.2.2.7 GetDefaultSiteCollectionTermStore .......................... 44
      3.1.5.15.2.2.8 GetTaxonomySession ............................................. 45
      3.1.5.15.2.2.9 GetTermsById ....................................................... 45
      3.1.5.15.2.2.10 GetTermSetsByName ............................................. 45
      3.1.5.15.2.2.11 GetTermSetsByTermLabel ...................................... 46
3.1.5.16 Microsoft.SharePoint.Client.Taxonomy.Term .................................. 46
  3.1.5.16.1 Properties ................................................................... 46
    3.1.5.16.1.1 Scalar Properties ........................................................ 46
      3.1.5.16.1.1.1 CreatedDate .......................................................... 46
      3.1.5.16.1.1.2 CustomProperties .................................................. 46
      3.1.5.16.1.1.3 CustomSortOrder ................................................... 47
      3.1.5.16.1.1.4 Id ...................................................................... 47
      3.1.5.16.1.1.5 IsAvailableForTagging .......................................... 47
      3.1.5.16.1.1.6 IsDeprecated ........................................................... 48
      3.1.5.16.1.1.7 IsKeyword ............................................................... 48
      3.1.5.16.1.1.8 IsReused ................................................................. 48
      3.1.5.16.1.1.9 IsRoot ................................................................. 48
      3.1.5.16.1.1.10 IsSourceTerm ........................................................ 48
      3.1.5.16.1.1.11 LastModifiedDate ............................................... 49
      3.1.5.16.1.1.12 Name .................................................................. 49
      3.1.5.16.1.1.13 Owner .................................................................. 49
      3.1.5.16.1.1.14 TermsCount .......................................................... 50
      3.1.5.16.1.1.15 IsPinned ............................................................... 50
      3.1.5.16.1.1.16 LocalCustomProperties ........................................ 50
      3.1.5.16.1.1.17 MergedTermIds ..................................................... 50
      3.1.5.16.1.1.18 Description ........................................................... 50
      3.1.5.16.1.1.19 PathOfTerm .......................................................... 51
      3.1.5.16.1.1.20 IsPinnedRoot ........................................................ 51
  3.1.5.16.2 ObjectPath Properties ......................................................... 51
    3.1.5.16.2.1 Labels .................................................................... 51
3.1.5.16.1.22 Parent.......................................................... 51
3.1.5.16.1.23 ReusedTerms............................................. 51
3.1.5.16.1.24 SourceTerm............................................... 52
3.1.5.16.1.25 Terms...................................................... 52
3.1.5.16.1.26 TermSet................................................... 52
3.1.5.16.1.27 TermSets.................................................. 53
3.1.5.16.1.28 TermStore................................................ 53
3.1.5.16.1.29 PinSourceTermSet........................................ 53
3.1.5.16.2 Methods...................................................... 53
3.1.5.16.2.1 Scalar Methods........................................... 53
3.1.5.16.2.1.1 DeleteCustomProperty.............................. 53
3.1.5.16.2.1.2 DeleteObject......................................... 54
3.1.5.16.2.1.3 Deprecate.............................................. 54
3.1.5.16.2.1.4 GetDefaultLabel..................................... 55
3.1.5.16.2.1.5 GetDescription....................................... 55
3.1.5.16.2.1.6 GetPath............................................... 56
3.1.5.16.2.1.7 Move.................................................. 56
3.1.5.16.2.1.8 ReassignSourceTerm............................... 57
3.1.5.16.2.1.9 SetCustomProperty................................. 57
3.1.5.16.2.1.10 SetDescription..................................... 58
3.1.5.16.2.1.11 DeleteAllCustomProperties...................... 59
3.1.5.16.2.1.12 DeleteAllLocalCustomProperties.............. 59
3.1.5.16.2.1.13 DeleteLocalCustomProperty.................. 59
3.1.5.16.2.1.14 GetIsDescendantOf............................... 59
3.1.5.16.2.1.15 NormalizeName.................................... 60
3.1.5.16.2.1.16 SetLocalCustomProperty....................... 60
3.1.5.16.2.2 ObjectPath Methods................................. 60
3.1.5.16.2.2.1 Copy................................................. 60
3.1.5.16.2.2.2 CreateLabel........................................ 61
3.1.5.16.2.2.3 CreateTerm......................................... 61
3.1.5.16.2.2.4 GetAllLabels...................................... 63
3.1.5.16.2.2.5 GetTerms........................................... 63
3.1.5.16.2.2.6 Merge............................................... 64
3.1.5.16.2.2.7 ReuseTerm.......................................... 65
3.1.5.16.2.2.8 ReuseTermWithPinning............................ 66
3.1.5.17 Microsoft.SharePoint.Client.Taxonomy.TermCollection................................. 66
3.1.5.17.1 Properties................................................. 66
3.1.5.17.1.1 Scalar Properties.................................... 66
3.1.5.17.1.2 ObjectPath Properties............................. 66
3.1.5.17.2 Methods.................................................. 66
3.1.5.17.2.1 Scalar Methods...................................... 67
3.1.5.17.2.2 ObjectPath Methods............................... 67
3.1.5.17.2.2.1 GetById............................................. 67
3.1.5.17.2.2.2 GetByName......................................... 67
3.1.5.18 Microsoft.SharePoint.Client.Taxonomy.TermGroup................................. 67
3.1.5.18.1 Properties................................................. 67
3.1.5.18.1.1 Scalar Properties.................................... 67
3.1.5.18.1.1.1 CreatedDate....................................... 67
3.1.5.18.1.1.2 Description........................................ 67
3.1.5.18.1.1.3 Id.................................................. 68
3.1.5.18.1.1.4 IsSiteCollectionGroup......................... 68
3.1.5.18.1.1.5 IsSystemGroup..................................... 68
3.1.5.18.1.1.6 LastModifiedDate................................. 68
3.1.5.18.1.1.7 Name................................................ 68
3.1.5.18.1.1.8 ContributorPrincipalNames.................. 68
3.1.5.18.1.1.9 GroupManagerPrincipalNames.................. 68
3.1.5.18.1.2 ObjectPath Properties............................. 69
3.1.5.18.1.2.1 TermSets.......................................... 69
3.1.5.18.1.2.2 TermStore .............................................................. 69
3.1.5.18.2 Methods ........................................................................ 69
  3.1.5.18.2.1 Scalar Methods ......................................................... 69
    3.1.5.18.2.1.1 DeleteObject ..................................................... 69
    3.1.5.18.2.1.2 ExportObject .................................................... 69
    3.1.5.18.2.1.3 NormalizeName ................................................ 69
    3.1.5.18.2.1.4 AddContributor ................................................ 69
    3.1.5.18.2.1.5 AddGroupManager ......................................... 70
  3.1.5.18.2.2 ObjectPath Methods .............................................. 70
    3.1.5.18.2.2.1 CreateTermSet ................................................. 70
    3.1.5.18.2.2.2 GetChanges ................................................... 70
    3.1.5.18.2.2.3 GetTermSetsWithCustomProperty ....................... 70
3.1.5.19 Microsoft.SharePoint.Client.Taxonomy.TermGroupCollection ....... 72
  3.1.5.19.1 Properties .................................................................. 72
    3.1.5.19.1.1 Scalar Properties ................................................ 72
    3.1.5.19.1.2 ObjectPath Properties ......................................... 72
  3.1.5.19.2 Methods .................................................................. 72
    3.1.5.19.2.1 Scalar Methods ................................................... 72
    3.1.5.19.2.2 ObjectPath Methods ........................................... 73
      3.1.5.19.2.2.1 GetById ....................................................... 73
      3.1.5.19.2.2.2 GetByName .................................................. 73
3.1.5.20 Microsoft.SharePoint.Client.Taxonomy.TermSet .................... 73
  3.1.5.20.1 Properties .................................................................. 73
    3.1.5.20.1.1 Scalar Properties ................................................ 73
      3.1.5.20.1.1.1 Contact ......................................................... 73
      3.1.5.20.1.1.2 CreatedDate ............................................... 74
      3.1.5.20.1.1.3 CustomProperties ......................................... 74
      3.1.5.20.1.1.4 CustomSortOrder .......................................... 74
      3.1.5.20.1.1.5 Description .................................................. 75
      3.1.5.20.1.1.6 Id ............................................................... 75
      3.1.5.20.1.1.7 IsAvailableForTagging ................................... 75
      3.1.5.20.1.1.8 IsOpenForTermCreation .................................. 76
      3.1.5.20.1.1.9 LastModifiedDate ......................................... 76
      3.1.5.20.1.1.10 Name ......................................................... 76
      3.1.5.20.1.1.11 Owner ......................................................... 77
      3.1.5.20.1.1.12 Stakeholders .............................................. 78
      3.1.5.20.1.1.13 Names ......................................................... 78
    3.1.5.20.1.2 ObjectPath Properties ........................................... 78
      3.1.5.20.1.2.1 Group ........................................................ 78
      3.1.5.20.1.2.2 Terms ......................................................... 78
      3.1.5.20.1.2.3 TermStore .................................................... 78
  3.1.5.20.2 Methods .................................................................. 78
    3.1.5.20.2.1 Scalar Methods ................................................... 78
      3.1.5.20.2.1.1 DeleteCustomProperty ................................... 79
      3.1.5.20.2.1.2 DeleteObject ............................................... 79
      3.1.5.20.2.1.3 ExportObject .............................................. 79
      3.1.5.20.2.1.4 Move .......................................................... 79
      3.1.5.20.2.1.5 SetCustomProperty ....................................... 80
      3.1.5.20.2.1.6 DeleteAllCustomProperties ............................ 80
      3.1.5.20.2.1.7 NormalizeName ............................................ 80
      3.1.5.20.2.1.8 AddStakeholder ........................................... 81
      3.1.5.20.2.1.9 DeleteStakeholder ....................................... 81
    3.1.5.20.2.2 ObjectPath Methods ........................................... 82
      3.1.5.20.2.2.1 Copy .......................................................... 82
      3.1.5.20.2.2.2 CreateTerm ............................................... 82
      3.1.5.20.2.2.3 GetAllTerms ................................................ 83
      3.1.5.20.2.2.4 GetChanges ................................................ 84
      3.1.5.20.2.2.5 GetTerm ...................................................... 84
3.1.5.20.2.2.6  GetTerms ................................................................. 84
3.1.5.20.2.2.7  GetTermsWithCustomProperty .................................. 86
3.1.5.20.2.2.8  ReuseTerm ............................................................. 87
3.1.5.20.2.2.9  ReuseTermWithPinning ........................................... 88
3.1.5.20.2.2.10 GetAllTermsIncludeDeprecated ............................... 88
3.1.5.21  Microsoft.SharePoint.Client.Taxonomy.TermSetCollection ....... 89
  3.1.5.21.1 Properties .................................................................. 89
     3.1.5.21.1.1 Scalar Properties ................................................... 89
     3.1.5.21.1.2 ObjectPath Properties ........................................... 89
  3.1.5.21.2 Methods .................................................................. 89
     3.1.5.21.2.1 Scalar Methods ...................................................... 89
     3.1.5.21.2.2 ObjectPath Methods .............................................. 89
        3.1.5.21.2.2.1 GetById ......................................................... 89
        3.1.5.21.2.2.2 GetByName ..................................................... 89
  3.1.5.22  Microsoft.SharePoint.Client.Taxonomy.TermSetItem .......... 89
  3.1.5.22.1  Properties ................................................................ 90
     3.1.5.22.1.1 Scalar Properties ................................................... 90
     3.1.5.22.1.2 ObjectPath Properties ........................................... 90
  3.1.5.22.2 Methods ................................................................ 90
     3.1.5.22.2.1 Scalar Methods ...................................................... 90
        3.1.5.22.2.1.1 DeleteCustomProperty ..................................... 91
        3.1.5.22.2.1.2 DeleteObject .................................................. 91
        3.1.5.22.2.1.3 SetCustomProperty .......................................... 91
        3.1.5.22.2.1.4 DeleteAllCustomProperties ................................ 91
        3.1.5.22.2.1.5 NormalizeName .............................................. 91
        3.1.5.22.2.2 ObjectPath Methods ........................................... 91
        3.1.5.22.2.2.1 CreateTerm ..................................................... 91
        3.1.5.22.2.2.2 ReuseTerm ...................................................... 91
        3.1.5.22.2.2.3 GetTerms ......................................................... 91
        3.1.5.22.2.2.4 ReuseTermWithPinning ..................................... 91
  3.1.5.23  Microsoft.SharePoint.Client.Taxonomy.TermStore ............. 95
  3.1.5.23.1  Properties ................................................................ 95
     3.1.5.23.1.1 Scalar Properties ................................................... 95
     3.1.5.23.1.2 ObjectPath Properties ........................................... 95
  3.1.5.23.2 Methods ................................................................ 95
     3.1.5.23.2.1 Scalar Methods ...................................................... 95

[MS-EMMCSOM] - v20181211
Enterprise Managed Metadata Client-Side Object Model Protocol
Copyright © 2018 Microsoft Corporation
Release: December 11, 2018
3.1.5.23.2.1.1 CommitAll ......................................................... 98
3.1.5.23.2.1.2 AddLanguage....................................................... 98
3.1.5.23.2.1.3 DeleteLanguage .................................................. 99
3.1.5.23.2.1.4 RollbackAll......................................................... 99
3.1.5.23.2.1.5 UpdateCache ...................................................... 100
3.1.5.23.2.1.6 UpdateUsedTermsOnSite ....................................... 100
3.1.5.23.2.1.7 GetPackage ......................................................... 100
3.1.5.23.2.1.8 GetPackagesUpdateInformation .............................. 101
3.1.5.23.2.1.9 UploadPackages .................................................. 101
3.1.5.23.2.2 ObjectPath Methods .............................................. 102
3.1.5.23.2.2.1 CreateGroup ....................................................... 102
3.1.5.23.2.2.2 GetChanges ....................................................... 103
3.1.5.23.2.2.3 GetGroup ........................................................ 103
3.1.5.23.2.2.4 GetTerm .......................................................... 103
3.1.5.23.2.2.5 GetTerms ......................................................... 104
3.1.5.23.2.2.6 GetTermSet ........................................................ 105
3.1.5.23.2.2.7 GetTermsWithCustomProperty ................................ 106
3.1.5.23.2.2.8 GetSiteCollectionGroup ....................................... 107
3.1.5.23.2.2.9 GetTermInTermSet .............................................. 108
3.1.5.23.2.2.10 GetTermsById .................................................. 108
3.1.5.23.2.2.11 GetTermSetsByName ........................................... 109
3.1.5.23.2.2.12 GetTermSetsByTermLabel ..................................... 109
3.1.5.23.2.2.13 GetTermSetsWithCustomProperty ............................ 110
3.1.5.24 Microsoft.SharePoint.Client.Taxonomy.TermStoreCollection ....... 112
3.1.5.24.1 Properties .......................................................... 112
3.1.5.24.1.1 Scalar Properties .................................................. 112
3.1.5.24.1.2 ObjectPath Properties ............................................ 112
3.1.5.24.2 Methods ............................................................. 112
3.1.5.24.2.1 Scalar Methods .................................................... 112
3.1.5.24.2.2 ObjectPath Methods .............................................. 112
3.1.5.24.2.2.1 GetByRange ....................................................... 112
3.1.5.24.2.2.2 GetByName ....................................................... 112
3.1.5.25 Microsoft.SharePoint.Client.Taxonomy.MobileTaxonomyField ...... 113
3.1.5.25.1 Properties .......................................................... 113
3.1.5.25.1.1 Scalar Properties .................................................. 113
3.1.5.25.1.1.1 ReadOnly ........................................................ 113
3.1.5.25.1.2 ObjectPath Properties ............................................ 113
3.1.5.25.2 Methods ............................................................. 113
3.1.5.25.2.1 Scalar Methods .................................................... 113
3.1.5.25.2.2 ObjectPath Methods .............................................. 113
3.1.5.26 Microsoft.SharePoint.Client.Taxonomy.ChangeInformation .......... 113
3.1.5.26.1 Properties .......................................................... 113
3.1.5.26.1.1 Scalar Properties .................................................. 113
3.1.5.26.1.1.1 ItemType ........................................................ 113
3.1.5.26.1.1.2 OperationType .................................................. 114
3.1.5.26.1.1.3 StartTime ........................................................ 114
3.1.5.26.1.1.4 WithinTimeSpan ............................................... 114
3.1.5.26.1.2 ObjectPath Properties ............................................ 114
3.1.5.26.2 Methods ............................................................. 114
3.1.5.26.2.1 Scalar Methods .................................................... 114
3.1.5.26.2.2 ObjectPath Methods .............................................. 114
3.1.5.26.2.2.1 CSOM Constructor .............................................. 114
3.1.5.27 Microsoft.SharePoint.Client.Taxonomy.TaxonomyField ............. 114
3.1.5.27.1 Properties .......................................................... 115
3.1.5.27.1.1 Scalar Properties .................................................. 115
3.1.5.27.1.1.1 AllowMultipleValues .......................................... 115
3.1.5.27.1.1.2 AnchorId ......................................................... 115
3.1.5.27.1.1.3 CanBeDeleted ................................................... 115

[MS-EMMCSOM] - v20181211
Enterprise Managed Metadata Client-Side Object Model Protocol
Copyright © 2018 Microsoft Corporation
Release: December 11, 2018
| 3.1.5.27.1.1.4 | CreateValuesInEditForm | 115 |
| 3.1.5.27.1.1.5 | DefaultValueTyped | 115 |
| 3.1.5.27.1.1.6 | Description | 116 |
| 3.1.5.27.1.1.7 | Direction | 116 |
| 3.1.5.27.1.1.8 | EnforceUniqueValues | 116 |
| 3.1.5.27.1.1.9 | FieldTypeKind | 117 |
| 3.1.5.27.1.1.10 | Filterable | 117 |
| 3.1.5.27.1.1.11 | FromBaseType | 117 |
| 3.1.5.27.1.1.12 | Group | 117 |
| 3.1.5.27.1.1.13 | Hidden | 118 |
| 3.1.5.27.1.1.14 | Id | 118 |
| 3.1.5.27.1.1.15 | Indexed | 118 |
| 3.1.5.27.1.1.16 | InternaName | 118 |
| 3.1.5.27.1.1.17 | IsAnchorValid | 118 |
| 3.1.5.27.1.1.18 | IsKeyword | 119 |
| 3.1.5.27.1.1.19 | IsPathRendered | 119 |
| 3.1.5.27.1.1.20 | IsRelationShip | 119 |
| 3.1.5.27.1.1.21 | IsTermSetValid | 120 |
| 3.1.5.27.1.1.22 | LookupField | 120 |
| 3.1.5.27.1.1.23 | LookupList | 120 |
| 3.1.5.27.1.1.24 | LookupWebId | 121 |
| 3.1.5.27.1.1.25 | Open | 121 |
| 3.1.5.27.1.1.26 | PrimaryFieldId | 121 |
| 3.1.5.27.1.1.27 | ReadOnlyField | 121 |
| 3.1.5.27.1.1.28 | RelationshipDeleteBehavior | 122 |
| 3.1.5.27.1.1.29 | Required | 122 |
| 3.1.5.27.1.1.30 | SchemaXml | 123 |
| 3.1.5.27.1.1.31 | Scope | 123 |
| 3.1.5.27.1.1.32 | Sealed | 123 |
| 3.1.5.27.1.1.33 | Sortable | 123 |
| 3.1.5.27.1.1.34 | SspId | 123 |
| 3.1.5.27.1.1.35 | StaticName | 124 |
| 3.1.5.27.1.1.36 | TargetTemplate | 124 |
| 3.1.5.27.1.1.37 | TermSetId | 124 |
| 3.1.5.27.1.1.38 | TextField | 124 |
| 3.1.5.27.1.1.39 | Title | 124 |
| 3.1.5.27.1.1.40 | TypeAsString | 125 |
| 3.1.5.27.1.1.41 | TypeDisplayName | 125 |
| 3.1.5.27.1.1.42 | TypeShortDescription | 125 |
| 3.1.5.27.1.1.43 | UserCreated | 125 |
| 3.1.5.27.1.1.44 | ValidationFormula | 126 |
| 3.1.5.27.1.1.45 | ValidationMessage | 126 |
| 3.1.5.27.1.1.46 | EntityPropertyName | 126 |
| 3.1.5.27.1.1.47 | JSLink | 126 |
| 3.1.5.27.1.1.48 | SchemaXmlWithResourceTokens | 126 |
| 3.1.5.27.1.1.49 | AutoIndexed | 126 |
| 3.1.5.27.1.1.50 | ClientSideComponentId | 126 |
| 3.1.5.27.1.1.51 | ClientSideComponentProperties | 127 |
| 3.1.5.27.1.1.52 | CustomFormatter | 127 |
| 3.1.5.27.1.1.53 | DefaultFormula | 127 |
| 3.1.5.27.1.1.54 | DependentLookupInternalNames | 127 |
| 3.1.5.27.1.1.55 | IsDependentLookup | 127 |
| 3.1.5.27.1.1.56 | NoCraw | 127 |
| 3.1.5.27.1.1.57 | PinnedToFiltersPane | 127 |
| 3.1.5.27.1.1.58 | ShowInFiltersPane | 127 |
| 3.1.5.27.1.1.59 | UnlimitedLengthInDocumentLibrary | 128 |

3.1.5.27.1.2 ObjectPath Properties
3.1.5.27.1.2.1 DefaultValueTyped

[MS-EMMCSOM] - v20181211
Enterprise Managed Metadata Client-Side Object Model Protocol
Copyright © 2018 Microsoft Corporation
Release: December 11, 2018
3.1.5.27.1.2.2 DescriptionResource ..........................................................128
3.1.5.27.1.2.3 TitleResource ..................................................................128
3.1.5.27.2 Methods .................................................................................128
3.1.5.27.2.1 Scalar Methods ..................................................................128
3.1.5.27.2.1.1 DeleteObject ....................................................................128
3.1.5.27.2.1.2 GetFieldValueAsHtml ..........................................................129
3.1.5.27.2.1.3 GetFieldValueAsText ..........................................................129
3.1.5.27.2.1.4 GetValidatedString ............................................................130
3.1.5.27.2.1.5 Update ............................................................................130
3.1.5.27.2.1.6 UpdateAndPushChanges ....................................................130
3.1.5.27.2.1.7 ValidateSetValue ...............................................................131
3.1.5.27.2.1.8 GetFieldValueAsTaxonomyFieldValue ..................................131
3.1.5.27.2.1.9 SetFieldValueByCollection ................................................132
3.1.5.27.2.1.10 SetFieldValueByTerm .........................................................132
3.1.5.27.2.1.11 SetFieldValueByTermCollection .........................................133
3.1.5.27.2.1.12 SetFieldValueByValue ........................................................134
3.1.5.27.2.1.13 SetFieldValueByValueCollection .......................................134
3.1.5.27.2.1.14 SetShowInDisplayForm ....................................................135
3.1.5.27.2.1.15 SetShowInEditForm ..........................................................135
3.1.5.27.2.1.16 SetShowInNewForm ..........................................................135
3.1.5.27.2.2 ObjectPath Methods ............................................................135
3.1.5.27.2.2.1 GetFieldValueAsTaxonomyFieldValueCollection ..................135
3.1.5.28 Microsoft.SharePoint.Client.Taxonomy.TaxonomyFieldValueCollection...136
3.1.5.28.1 Properties ..............................................................................136
3.1.5.28.1.1 Scalar Properties .................................................................136
3.1.5.28.1.2 ObjectPath Properties ..........................................................136
3.1.5.28.2 Methods ...............................................................................136
3.1.5.28.2.1 Scalar Methods .................................................................136
3.1.5.28.2.1.1 PopulateFromLabelGuidPairs .............................................136
3.1.5.28.2.2 ObjectPath Methods ...........................................................137
3.1.5.28.2.2.1 CSOM Constructor ...........................................................137
3.1.5.29 Microsoft.SharePoint.Client.Taxonomy.CustomPropertyMatchInformation...137
3.1.5.29.1 Properties ..............................................................................137
3.1.5.29.1.1 Scalar Properties .................................................................137
3.1.5.29.1.1.1 CustomPropertyName .........................................................137
3.1.5.29.1.1.2 CustomPropertyValue .......................................................138
3.1.5.29.1.1.3 ResultCollectionSize ........................................................138
3.1.5.29.1.1.4 StringMatchOption ..........................................................138
3.1.5.29.1.1.5 TrimUnavailable ...............................................................138
3.1.5.29.1.2 ObjectPath Properties ........................................................138
3.1.5.29.2 Methods ...............................................................................138
3.1.5.29.2.1 Scalar Methods .................................................................138
3.1.5.29.2.2 ObjectPath Methods ...........................................................139
3.1.5.29.2.2.1 CSOM Constructor ...........................................................139
3.1.5.30 Microsoft.SharePoint.Client.Taxonomy.LabelMatchInformation ...........139
3.1.5.30.1 Properties ..............................................................................139
3.1.5.30.1.1 Scalar Properties .................................................................139
3.1.5.30.1.1.1 DefaultLabelOnly .............................................................139
3.1.5.30.1.1.2 ExcludeKeyword ..............................................................139
3.1.5.30.1.1.3 Lcid ..............................................................................139
3.1.5.30.1.1.4 ResultCollectionSize ........................................................139
3.1.5.30.1.1.5 StringMatchOption ..........................................................139
3.1.5.30.1.1.6 TermLabel .....................................................................140
3.1.5.30.1.1.7 TrimDeprecated ...............................................................140
3.1.5.30.1.1.8 TrimUnavailable ...............................................................140
3.1.5.30.1.2 ObjectPath Properties ........................................................140
3.1.5.30.2 Methods ...............................................................................140
3.1.5.30.2.1 Scalar Methods .................................................................140
1 Introduction

The Enterprise Managed Metadata Client-Side Object Model Protocol provides types, methods, and properties to enable a protocol client to create, read, update, and delete enterprise metadata on a collaboration server.

Sections 1.5, 1.8, 1.9, 2, and 3 of this specification are normative. All other sections and examples in this specification are informative.

1.1 Glossary

This document uses the following terms:

**Boolean**: An operation or expression that can be evaluated only as either true or false.

**contact**: A person, company, or other entity that is stored in a directory and is associated with one or more unique identifiers and attributes, such as an Internet message address or login name.

**content type**: A named and uniquely identifiable collection of settings and fields that store metadata for individual items in a SharePoint list. One or more content types can be associated with a list, which restricts the contents to items of those types.

**Coordinated Universal Time (UTC)**: A high-precision atomic time standard that approximately tracks Universal Time (UT). It is the basis for legal, civil time all over the Earth. Time zones around the world are expressed as positive and negative offsets from UTC. In this role, it is also referred to as Zulu time (Z) and Greenwich Mean Time (GMT). In these specifications, all references to UTC refer to the time at UTC-0 (or GMT).

**CSOM array**: An ordered collection of values that can be used in an XML request or JSON response text. The values are identified by their position and their position is determined by a zero-based integer index.

**CSOM Boolean**: A Boolean value that can be used in an XML request or JSON response text. A CSOM Boolean value is either "true" or "false".

**CSOM DateTime**: An Int64 value that represents the number of 100-nanosecond time intervals that have elapsed since 12:00:00, January 1, 0001. It can be used in an XML request or as a string in JSON response text. The value can represent time intervals through 23:59:59.9999999, December 31, 9999. It can also specify whether a local, UTC, or no time zone applies.

**CSOM dictionary**: An object that contains an unordered collection of key/value pairs that can be used in an XML request or JSON response text. Each key in a CSOM dictionary has a unique name.

**CSOM GUID**: A GUID, as described in [MS-DTYP], that can be used in an XML request or as a string in JSON response text.

**CSOM Int32**: A 32-bit, signed integer value, which is the INT32 type described in [MS-DTYP], that can be used in an XML request or as a number in JSON response text. The range of CSOM Int32 values is from "-2147483648" to "2147483647".

**CSOM Object**: An object that contains a set of members, which are named values and methods. It has a Unicode string value, which is referred to as a CSOM type name, that identifies its type.

**CSOM Stream**: A series of bytes that is used to transfer data and that supports reading, writing, and seeking.
**CSOM String**: A representation of text as a series of Unicode characters. It can be used in an XML request or **JSON** response text.

**CSOM TimeSpan**: A data type that represents a time interval (duration of time or elapsed time) that is measured as a positive or negative number of days, hours, minutes, seconds, and fractions of a second.

**default scalar property set**: A set of properties that are retrieved by default for an object. The properties map to fields in a storage schema.

**deprecated term**: A term that persists in the term store but cannot be used in future applications of metadata.

**display form**: A form that is used to display a list item.

**edit form**: A form that enables users to edit a list item.

**field**: A container for metadata within a SharePoint list and associated list items.

**global group**: An Active Directory group that allows user objects from its own domain and global groups from its own domain as members. Also called domain global group. Universal groups can contain global groups. A group object g is a **global group** if and only if GROUP_TYPE_ACCOUNT_GROUP is present in g! groupType; see [MS-ADTS] section 2.2.12, "Group Type Flags". A **global group** that is also a security-enabled group is valid for inclusion within ACLs anywhere in the forest. If a domain is in mixed mode, then a global group in that domain that is also a security-enabled group allows only user object as members. See also domain local group, security-enabled group.

**globally unique identifier (GUID)**: A term used interchangeably with universally unique identifier (UUID) in Microsoft protocol technical documents (TDs). Interchanging the usage of these terms does not imply or require a specific algorithm or mechanism to generate the value. Specifically, the use of this term does not imply or require that the algorithms described in [RFC4122] or [C706] must be used for generating the **GUID**. See also universally unique identifier (UUID).

**hierarchy**: A logical tree structure that organizes the members of a dimension such that each member has one parent member and zero or more child members.

**Hypertext Markup Language (HTML)**: An application of the Standard Generalized Markup Language (SGML) that uses tags to mark elements in a document, as described in [HTML].

**Hypertext Transfer Protocol (HTTP)**: An application-level protocol for distributed, collaborative, hypermedia information systems (text, graphic images, sound, video, and other multimedia files) on the World Wide Web.

**Hypertext Transfer Protocol Secure (HTTPS)**: An extension of HTTP that securely encrypts and decrypts web page requests. In some older protocols, "Hypertext Transfer Protocol over Secure Sockets Layer" is still used (Secure Sockets Layer has been deprecated). For more information, see [SSL3] and [RFC5246].

**JavaScript Object Notation (JSON)**: A text-based, data interchange format that is used to transmit structured data, typically in Asynchronous JavaScript + XML (AJAX) web applications, as described in [RFC7159]. The JSON format is based on the structure of ECMAScript (Jscript, JavaScript) objects.

**keyword**: One or more words or phrases that site administrators identified as important for a search service application. A keyword provides a way to apply business rules to search results for queries that use the keyword.
language code identifier (LCID): A 32-bit number that identifies the user interface human language dialect or variation that is supported by an application or a client computer.

list item: An individual entry within a SharePoint list. Each list item has a schema that maps to fields in the list that contains the item, depending on the content type of the item.

lookup field: A field of the Lookup type that enables users to select an item from another data source.

mobile device: A small computing device that is easily portable and can be used in various environments.

new form: A form that enables users to create a list item.

orphaned term: A term whose source term was deleted.

resource token: An encoded string that refers to an entry in a localized resource file.

site: A group of related pages and data within a SharePoint site collection. The structure and content of a site is based on a site definition. Also referred to as SharePoint site and web site.

site collection: A set of websites that are in the same content database, have the same owner, and share administration settings. A site collection can be identified by a GUID or the URL of the top-level site for the site collection. Each site collection contains a top-level site, can contain one or more subsites, and can have a shared navigational structure.

site collection group: A named collection of users and domain groups that can be managed for a site collection or site. A group can be used to assign permission levels, send email messages, and indicate site membership for multiple users simultaneously.

source term: A specific instance of a term, in a specific term set, that is used to define permissions for the term.

static CSOM method: A class method that is accessed through the type name rather than an instance of the class.

term: A concept or an idea that is stored and can be used as metadata.

term label: A string that is used as the display value for a term. Each term label is associated with a specific language.

term set: A collection of terms that are arranged into and stored as a hierarchy or a flat list.

term store: A database in which managed metadata is stored in the form of term sets and terms.

Uniform Resource Identifier (URI): A string that identifies a resource. The URI is an addressing mechanism defined in Internet Engineering Task Force (IETF) Uniform Resource Identifier (URI): Generic Syntax [RFC3986].

Uniform Resource Locator (URL): A string of characters in a standardized format that identifies a document or resource on the World Wide Web. The format is as specified in [RFC1738].

web application: A container in a configuration database that stores administrative settings and entry-point URLs for site collections.

website: A group of related pages and data within a SharePoint site collection. The structure and content of a site is based on a site definition. Also referred to as SharePoint site and site.

XML schema: A description of a type of XML document that is typically expressed in terms of constraints on the structure and content of documents of that type, in addition to the basic...
syntax constraints that are imposed by XML itself. An XML schema provides a view of a document type at a relatively high level of abstraction.

**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.

- [MS-CSOM] Microsoft Corporation, "SharePoint Client Query Protocol".
- [MS-PRIMEPF] Microsoft Corporation, "Deployment Package Format".
- [MS-WSSCAML] Microsoft Corporation, "Collaborative Application Markup Language (CAML) Structure".

#### 1.2.2 Informative References

None.

### 1.3 Overview

This protocol defines types, methods, and properties that a protocol client uses to access enterprise managed metadata in a term store on a protocol server. Some typical scenarios in which a protocol client could use this protocol are to retrieve terms, add terms, add term labels, or remove terms.
1.4 Relationship to Other Protocols

The Enterprise Managed Metadata Client-Side Object Model Protocol is a set of types, properties, and methods that can be accessed by using the SharePoint Client Query protocol as described in [MS-CSOM]. This protocol uses JavaScript Object Notation (JSON) as described in [RFC4627] to format data returned to a protocol client. This protocol also uses HTTP, as described in [RFC2616], and HTTPS, as described in [RFC2818]. The dependencies for this protocol are shown in the following layering diagram.

![Layering Diagram](image)

Figure 1: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a term store on a site that is identified by a Uniform Resource Locator (URL) that is known by protocol clients. This protocol assumes that authentication has been performed by underlying protocols. This protocol specifies properties, methods, and types used to access a store of structured metadata as described in [MS-EMMWCF].

1.6 Applicability Statement

This protocol can be used by a protocol client to manage enterprise metadata on a protocol server. This protocol is optimized to enable a protocol client to specify the exact set of data and operations to perform in a single batch, making it suitable for situations where the connection speed between the protocol client and the protocol server can be slow. This protocol is not suitable and is inefficient if both the protocol client and protocol server are on the same computer. In this case, the client can use an API that does not require communication over a network.

1.7 Versioning and Capability Negotiation

This protocol operates in the context of the authenticated user that made a request to the service, and it runs in the context of a particular site. Localization-specific properties, such as the label of a term, are returned by using the default language of the term store, if available. Certain properties can be read or written for specific locales, and are listed in the document.

1.8 Vendor-Extensible Fields

A protocol server can provide additional properties and methods beyond those specified in this document to provide extended collaboration server functionality; however, because no means are provided for a protocol client to discover that extended functionality, the functionality will be taken advantage of only by protocol clients that have built-in knowledge of the extensions.
A protocol client is expected to be tolerant of the situation where certain methods and properties specified in this specification are not available, because either the protocol server does not support the functionality or the administrator has disabled certain functionalities.

1.9 Standards Assignments

None.
2 Messages

2.1 Transport

Messages are transported by using the SharePoint Client Query Protocol, as specified in [MS-CSOM].

2.2 Message Syntax

None.
3 Protocol Details

3.1 Server Details

3.1.1 Abstract Data Model

This protocol deals with various types that are logically derived from a parent type. The following table lists the relation between such types.

<table>
<thead>
<tr>
<th>Parent Type</th>
<th>Derived Type(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Microsoft.SharePoint.Client.Taxonomy.ChangedSite (section 3.1.5.6)</td>
</tr>
<tr>
<td></td>
<td>Microsoft.SharePoint.Client.Taxonomy.ChangedTerm (section 3.1.5.7)</td>
</tr>
<tr>
<td></td>
<td>Microsoft.SharePoint.Client.Taxonomy.ChangedTermSet (section 3.1.5.8)</td>
</tr>
<tr>
<td></td>
<td>Microsoft.SharePoint.Client.Taxonomy.ChangedTermStore (section 3.1.5.9)</td>
</tr>
<tr>
<td></td>
<td>Microsoft.SharePoint.Client.Taxonomy.TermSet (section 3.1.5.20)</td>
</tr>
<tr>
<td></td>
<td>Microsoft.SharePoint.Client.Taxonomy.TermSetItem (section 3.1.5.22)</td>
</tr>
</tbody>
</table>

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Higher-Layer Triggered Events

None.

3.1.5 Message Processing Events and Sequencing Rules

3.1.5.1 Microsoft.SharePoint.Client.Taxonomy.ChangedGroup

TypeId: {D294C9CA-4667-44E4-8C6E-3EB510BAA358}
ShortName: SP.Taxonomy.ChangedGroup

Represents a change to a TermGroup (section 3.1.5.18) object.

Remarks:
The ChangedGroup object derives from the ChangedItem (section 3.1.5.2) base object.

3.1.5.1.1 Properties

3.1.5.1.1.1 Scalar Properties

3.1.5.1.1.1.1 ChangedBy
Type: CSOM String
Accessibility: Read Only
See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ChangedBy, as specified in section 3.1.5.2.1.1.1.

3.1.5.1.1.1.2 ChangedTime
Type: CSOM DateTime
Accessibility: Read Only
See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ChangedTime, as specified in section 3.1.5.2.1.1.2.

3.1.5.1.1.1.3 Id
Type: CSOM GUID
Accessibility: Read Only
See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.Id section 3.1.5.2.1.1.3.

3.1.5.1.1.1.4 ItemType
Type: Microsoft.SharePoint.Client.Taxonomy.ChangedItemType
Accessibility: Read Only
See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ItemType section 3.1.5.2.1.1.4.

3.1.5.1.1.1.5 Operation
Type: Microsoft.SharePoint.Client.Taxonomy.ChangedOperationType
Accessibility: Read Only
See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.Operation section 3.1.5.2.1.1.5.

3.1.5.1.2 ObjectPath Properties
None.

3.1.5.1.2 Methods
3.1.5.1.2.1 Scalar Methods
None.

3.1.5.1.2.2 ObjectPath Methods
None.

3.1.5.2 Microsoft.SharePoint.Client.Taxonomy.ChangedItem
TypeId: {78C8F4D6-FC4F-46D4-8D00-7BB050DD7059}
ShortName: SP.Taxonomy.ChangedItem
Represents one change to an item in the TermStore (section 3.1.5.23).

3.1.5.2.1 Properties

3.1.5.2.1.1 Scalar Properties

3.1.5.2.1.1.1 ChangedBy
Type: CSOM String
Accessibility: Read Only
Reserved for future use.

3.1.5.2.1.1.2 ChangedTime
Type: CSOM DateTime
Accessibility: Read Only
Gets the Coordinated Universal Time (UTC) time when the change happened.

3.1.5.2.1.1.3 Id
Type: CSOM GUID
Accessibility: Read Only
Gets the identifier of the changed object.
Remarks:
This property does not have a value when the change is global. For example, when the default language is changed in a TermStore (section 3.1.5.23) object, or when the changed object does not have an identifier, this property does not have a value.

3.1.5.2.1.1.4 ItemType
Type: Microsoft.SharePoint.Client.Taxonomy.ChangedItemType
Accessibility: Read Only
Gets the type, as specified by ChangedItemType (section 3.1.5.4), of the changed object.

3.1.5.2.1.1.5 Operation
Type: Microsoft.SharePoint.Client.Taxonomy.ChangedOperationType

Accessibility: Read Only

 Gets the type of change that occurred as specified by ChangedOperationType (section 3.1.5.5); for example, add, edit, or delete.

3.1.5.2.1.2 ObjectPath Properties
None.

3.1.5.2.2 Methods

3.1.5.2.2.1 Scalar Methods
None.

3.1.5.2.2.2 ObjectPath Methods
None.

3.1.5.3 Microsoft.SharePoint.Client.Taxonomy.ChangedItemCollection

TypeId: {2D685E53-7F47-4A60-82F6-22C92177BE4E}

ShortName: SP.Taxonomy.ChangedItemCollection

A collection of ChangedItem (section 3.1.5.2) objects.

3.1.5.3.1 Properties

3.1.5.3.1.1 Scalar Properties
None.

3.1.5.3.1.2 ObjectPath Properties
None.

3.1.5.3.2 Methods

3.1.5.3.2.1 Scalar Methods
None.

3.1.5.3.2.2 ObjectPath Methods
None.

3.1.5.4 Microsoft.SharePoint.Client.Taxonomy.ChangedItemType

ShortName: SP.Taxonomy.ChangedItemType

Flag: false

Represents the type of changed object.
3.1.5.4.1 Field Values

3.1.5.4.1.1 Unknown
Value: 0
The change is not valid.

3.1.5.4.1.2 Term
Value: 1
Represents a Term (section 3.1.16) object.

3.1.5.4.1.3 TermSet
Value: 2
Represents a TermSet (section 3.1.20) object.

3.1.5.4.1.4 Group
Value: 3
Represents a TermGroup (section 3.1.18) object.

3.1.5.4.1.5 TermStore
Value: 4
Represents a TermStore (section 3.1.23) object. The change indicated is a global change, such as the set of available languages that have been changed.

3.1.5.4.1.6 Site
Value: 5
Represents a reused term on a site.

3.1.5.5 Microsoft.SharePoint.Client.Taxonomy.ChangedOperationType
ShortName: SP.Taxonomy.ChangedOperationType
Flag: false
Indicates the type of operation that caused a change.

3.1.5.5.1 Field Values

3.1.5.5.1.1 Unknown
Value: 0
Represents an invalid operation.

3.1.5.5.1.2 Add
Value: 1
Represents the add operation.

3.1.5.5.1.3  Edit
Value: 2
Represents the edit operation.

3.1.5.5.1.4  DeleteObject
Value: 3
Represents the delete operation.

3.1.5.5.1.5  Move
Value: 4
Represents the move operation.

3.1.5.5.1.6  Copy
Value: 5
Represents the copy operation.

3.1.5.5.1.7  PathChange
Value: 6
Represents the path change operation.

3.1.5.5.1.8  Merge
Value: 7
Represents the merge operation.

3.1.5.5.1.9  ImportObject
Value: 8
Represents the import operation.

3.1.5.5.1.10  Restore
Value: 9
Represents the restore operation.

3.1.5.6  Microsoft.SharePoint.Client.Taxonomy.ChangedSite
TypeId: {FAB02250-F233-461F-812E-EE265E246459}
ShortName: SP.Taxonomy.ChangedSite
Represents an event that changed something in the hidden list.

Remarks:
The `ChangedSite` object derives from the `ChangedItem` (section 3.1.5.2) base object.

### 3.1.5.6.1 Properties

#### 3.1.5.6.1.1 Scalar Properties

##### 3.1.5.6.1.1.1 ChangedBy

**Type:** CSOM String  
**Accessibility:** Read Only

See `Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ChangedBy`, as specified in section 3.1.5.2.1.1.1.

##### 3.1.5.6.1.1.2 ChangedTime

**Type:** CSOM DateTime  
**Accessibility:** Read Only

See `Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ChangedTime`, as specified in section 3.1.5.2.1.1.2.

##### 3.1.5.6.1.1.3 Id

**Type:** CSOM GUID  
**Accessibility:** Read Only

See `Microsoft.SharePoint.Client.Taxonomy.ChangedItem.Id` section 3.1.5.2.1.1.3.

##### 3.1.5.6.1.1.4 ItemType

**Type:** `Microsoft.SharePoint.Client.Taxonomy.ChangedItemType`  
**Accessibility:** Read Only

See `Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ItemType` section 3.1.5.2.1.1.4.

##### 3.1.5.6.1.1.5 Operation

**Type:** `Microsoft.SharePoint.Client.Taxonomy.ChangedOperationType`  
**Accessibility:** Read Only

See `Microsoft.SharePoint.Client.Taxonomy.ChangedItem.Operation` section 3.1.5.2.1.1.5.

##### 3.1.5.6.1.1.6 SiteId

**Type:** CSOM GUID  
**Accessibility:** Read Only

Gets the site identifier.

##### 3.1.5.6.1.1.7 TermId

**Type:** CSOM GUID
**Accessibility:** Read Only

Gets the **Term** (section 3.1.5.16) identifier.

### 3.1.5.6.1.1.8 TermSetId

**Type:** CSOM GUID

**Accessibility:** Read Only

Gets the **TermSet** (section 3.1.5.20) identifier.

### 3.1.5.6.1.2 ObjectPath Properties

None.

### 3.1.5.6.2 Methods

#### 3.1.5.6.2.1 Scalar Methods

None.

#### 3.1.5.6.2.2 ObjectPath Methods

None.

### 3.1.5.7 Microsoft.SharePoint.Client.Taxonomy.ChangedTerm

**TypeId:** `{08A04A9A-C4CE-46D4-A2C3-5C369BC69780}`

**ShortName:** SP.Taxonomy.ChangedTerm

Indicates a change that happened to a **Term** (section 3.1.5.16) object.

**Remarks:**

The **ChangedTerm** object derives from the **ChangedItem** (section 3.1.5.2) base object.

#### 3.1.5.7.1 Properties

#### 3.1.5.7.1.1 Scalar Properties

#### 3.1.5.7.1.1.1 ChangedBy

**Type:** CSOM String

**Accessibility:** Read Only

See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ChangedBy, as specified in section 3.1.5.2.1.1.1.

#### 3.1.5.7.1.1.2 ChangedTime

**Type:** CSOM DateTime

**Accessibility:** Read Only
See `Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ChangedTime`, as specified in section 3.1.5.2.1.1.2.

### 3.1.5.7.1.1.3 GroupId

**Type:** CSOM GUID  
**Accessibility:** Read Only  

Gets the identifier of the TermGroup (section 3.1.5.18) object to which the Term (section 3.1.5.4) object belongs.

### 3.1.5.7.1.1.4 Id

**Type:** CSOM GUID  
**Accessibility:** Read Only  

See `Microsoft.SharePoint.Client.Taxonomy.ChangedItem.Id` section 3.1.5.2.1.1.3.

### 3.1.5.7.1.1.5 ItemType

**Type:** Microsoft.SharePoint.Client.Taxonomy.ChangedItemType  
**Accessibility:** Read Only  

See `Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ItemType` section 3.1.5.2.1.1.4.

### 3.1.5.7.1.1.6 Operation

**Type:** Microsoft.SharePoint.Client.Taxonomy.ChangedOperationType  
**Accessibility:** Read Only  

See `Microsoft.SharePoint.Client.Taxonomy.ChangedItem.Operation` section 3.1.5.2.1.1.5.

### 3.1.5.7.1.1.7 TermSetId

**Type:** CSOM GUID  
**Accessibility:** Read Only  

Gets the identifier of the TermSet (section 3.1.5.20) object to which the Term (section 3.1.5.16) object belongs.

### 3.1.5.7.1.1.8 ChangedCustomProperties

**Type:** CSOM array of CSOM String  
**Accessibility:** Read Only  

Specifies a collection of names of custom properties that have been changed.

### 3.1.5.7.1.1.9 ChangedLocalCustomProperties

**Type:** CSOM array of CSOM String  
**Accessibility:** Read Only  

Specifies a collection of names of local custom properties that have been changed.
3.1.5.7.1.1.10 LcidsForChangedDescriptions
Type: CSOM array of CSOM Int32
Accessibility: Read Only
Specifies language code identifiers (LCIDs) in which descriptions have changed.

3.1.5.7.1.1.11 LcidsForChangedLabels
Type: CSOM array of CSOM Int32
Accessibility: Read Only
Specifies language code identifiers (LCIDs) in which labels have changed.

3.1.5.7.2 Methods

3.1.5.7.2.1 Scalar Methods
None.

3.1.5.7.2.2 ObjectPath Methods
None.

3.1.5.8 Microsoft.SharePoint.Client.Taxonomy.ChangedTermSet
TypeId: {5E776E93-0414-44A7-9A54-A00C594ED47C}
ShortName: SP.Taxonomy.ChangedTermSet
Indicates a change that happened to a TermSet (section 3.1.5.20) object.

Remarks:
The ChangedTermSet object derives from the ChangedItem (section 3.1.5.2) base object.

3.1.5.8.1 Properties

3.1.5.8.1.1 Scalar Properties

3.1.5.8.1.1.1 ChangedBy
Type: CSOM String
Accessibility: Read Only
See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ChangedBy, as specified in section 3.1.5.2.1.1.1.

3.1.5.8.1.1.2 ChangedTime
Type: CSOM DateTime
3.1.5.8.1.1.3 FromGroupId

**Type:** CSOM GUID

**Accessibility:** Read Only

Gets the identifier of the TermGroup (section 3.1.5.18) object from which a TermSet (section 3.1.5.20) object is moved.

3.1.5.8.1.1.4 GroupId

**Type:** CSOM GUID

**Accessibility:** Read Only

Gets the identifier of the TermGroup (section 3.1.5.18) object to which the changed TermSet (section 3.1.5.20) object belongs.

3.1.5.8.1.1.5 Id

**Type:** CSOM GUID

**Accessibility:** Read Only

See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.Id section 3.1.5.2.1.1.3.

3.1.5.8.1.1.6 ItemType

**Type:** Microsoft.SharePoint.Client.Taxonomy.ChangedItemType

**Accessibility:** Read Only

See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ItemType section 3.1.5.2.1.1.4.

3.1.5.8.1.1.7 Operation

**Type:** Microsoft.SharePoint.Client.Taxonomy.ChangedOperationType

**Accessibility:** Read Only

See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.Operation section 3.1.5.2.1.1.5.

3.1.5.8.1.1.2 ObjectPath Properties

None.

3.1.5.8.2 Methods

3.1.5.8.2.1 Scalar Methods

None.

3.1.5.8.2.2 ObjectPath Methods


None.

3.1.5.9 Microsoft.SharePoint.Client.Taxonomy.ChangedTermStore

TypeId: {DB44A214-C48A-4607-8C77-4AC9EE96C263}

ShortName: SP.Taxonomy.ChangedTermStore

Indicates a change that happened to a TermStore (section 3.1.5.23) object.

Remarks:
The ChangedTermStore object derives from the ChangedItem (section 3.1.5.2) base object.

3.1.5.9.1 Properties

3.1.5.9.1.1 Scalar Properties

3.1.5.9.1.1.1 ChangedBy

Type: CSOM String

Accessibility: Read Only

See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ChangedBy, as specified in section 3.1.5.2.1.1.1.

3.1.5.9.1.1.2 ChangedLanguage

Type: CSOM Int32

Accessibility: Read Only

Gets an integer that represents the changed language.

3.1.5.9.1.1.3 ChangedTime

Type: CSOM DateTime

Accessibility: Read Only

See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ChangedTime, as specified in section 3.1.5.2.1.1.2.

3.1.5.9.1.1.4 Id

Type: CSOM GUID

Accessibility: Read Only

See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.Id section 3.1.5.2.1.1.3.

3.1.5.9.1.1.5 IsDefaultLanguageChanged

Type: CSOM Boolean

Accessibility: Read Only
Gets a Boolean value that indicates whether this change changed the default language of the TermStore (section 3.1.5.23) object.

3.1.5.9.1.1.6 IsFullFarmRestore

Type: CSOM Boolean

Accessibility: Read Only

Gets a Boolean value that indicates whether the TermStore (section 3.1.5.23) change is a full farm restore.

3.1.5.9.1.1.7 ItemType

Type: Microsoft.SharePoint.Client.Taxonomy.ChangedItemType

Accessibility: Read Only

See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.ItemType section 3.1.5.2.1.1.4.

3.1.5.9.1.1.8 Operation

Type: Microsoft.SharePoint.Client.Taxonomy.ChangedOperationType

Accessibility: Read Only

See Microsoft.SharePoint.Client.Taxonomy.ChangedItem.Operation section 3.1.5.2.1.1.5.

3.1.5.9.1.2 ObjectPath Properties

None.

3.1.5.9.2 Methods

3.1.5.9.2.1 Scalar Methods

None.

3.1.5.9.2.2 ObjectPath Methods

None.

3.1.5.10 Microsoft.SharePoint.Client.Taxonomy.Label

TypeId: {81503AE1-8747-4684-A172-163C7E009EF9}

ShortName: SP.Taxonomy.Label

Represents the name of a Term (section 3.1.5.16) object in a specific language.

A term can have one or more labels in the default language, and zero or more labels in the non-default language. If the term has labels in a language, one of the labels MUST be the default label.

3.1.5.10.1 Properties

3.1.5.10.1.1 Scalar Properties

3.1.5.10.1.1.1 IsDefaultForLanguage
**3.1.5.10.1.1.2 Language**

**Type:** CSOM Int32

**Accessibility:** Read/Write

Indicates the locale of the current Label (section 3.1.5.10) object.

**Remarks:**

Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The language of a default Label (section 3.1.5.10) cannot be changed.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The language is not a valid language or is not one of the working languages for the current TermStore (section 3.1.5.23).</td>
</tr>
</tbody>
</table>

**3.1.5.10.1.1.3 Value**

**Type:** CSOM String

**Accessibility:** Read/Write

Gets or sets the value of the current Label (section 3.1.5.10) object.

**Remarks:**

Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value is not valid for a Term (section 3.1.5.16) label. The Label cannot have more than 255 characters, and cannot contain any invalid characters including:</td>
</tr>
</tbody>
</table>
### Error Code Table

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>semicolon, double quote, left angle bracket, right angle bracket, ampersand, pipe, and tab.</td>
</tr>
</tbody>
</table>

### 3.1.5.10.1.2 ObjectPath Properties

#### 3.1.5.10.1.2.1 Term

**Type:** Microsoft.SharePoint.Client.Taxonomy.Term

**Accessibility:** Read Only

Returns the **Term** (section 3.1.5.16) object that the current **Label** (section 3.1.5) object is associated with.

#### 3.1.5.10.2 Methods

**3.1.5.10.2.1 Scalar Methods**

#### 3.1.5.10.2.1.1 DeleteObject

**Return Type:** None

Deletes the current **Label** (section 3.1.5) object.

**Remarks:**

Call the **CommitAll** (section 3.1.5.23.2.1.1) method to save the deletion to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The default label for a <strong>Term</strong> (section 3.1.5.16) object in the default language cannot be deleted. Deleting the last <strong>Label</strong> of a <strong>Term</strong> is not allowed. Delete the <strong>Term</strong> instead.</td>
</tr>
</tbody>
</table>

**Parameters:** None

#### 3.1.5.10.2.1.2 SetAsDefaultForLanguage

**Return Type:** None

Sets the current **Label** section (3.1.5.10) object as the default label for its language.

**Remarks:**

Call the **CommitAll** (section 3.1.5.23.2.1.1) method to save this change to the database.
Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
</tbody>
</table>

Parameters: None

3.1.5.10.2.2 ObjectPath Methods

None.

3.1.5.11 Microsoft.SharePoint.Client.Taxonomy.LabelCollection

TypeId: \{1849DC80-6AED-4ACC-A38F-18CAF1D7A216\}

ShortName: SP.Taxonomy.LabelCollection

Represents a collection of Label (section 3.1.5.10) objects.

3.1.5.11.1 Properties

3.1.5.11.1.1 Scalar Properties

None.

3.1.5.11.1.2 ObjectPath Properties

None.

3.1.5.11.2 Methods

3.1.5.11.2.1 Scalar Methods

None.

3.1.5.11.2.2 ObjectPath Methods

3.1.5.11.2.2.1 GetByValue

Return Type: Microsoft.SharePoint.Client.Taxonomy.Label

Gets a Label (section 3.1.5.10) object by value (section 3.1.5.10.1.1.3).

Parameters:

index: The value of the Label (section 3.1.5.10) object to retrieve.

Type: CSOM String

3.1.5.12 Microsoft.SharePoint.Client.Taxonomy.StringMatchOption

ShortName: SP.Taxonomy.StringMatchOption

Flag: false
Indicates what type of string matching to be performed when comparing strings.

### 3.1.5.12.1 Field Values

#### 3.1.5.12.1.1 StartsWith

**Value:** 0

Indicates that the string that is being matched MUST start with the specified value.

#### 3.1.5.12.1.2 ExactMatch

**Value:** 1

Indicates that the string that is being matched MUST match exactly all of the characters in the specified value.

### 3.1.5.13 Microsoft.SharePoint.Client.Taxonomy.TaxonomyFieldValue

**TypeId:** \{19E70ED0-4177-456B-8156-015E4D163FF8\}

**ShortName:** SP.Taxonomy.TaxonomyFieldValue

Represents a single value held in a TaxonomyField (section 3.1.5.27) object.

#### 3.1.5.13.1 Properties

##### 3.1.5.13.1.1 Scalar Properties

#### 3.1.5.13.1.1.1 Label

**Type:** CSOM String

**Accessibility:** Read/Write

Specifies the label of the TaxonomyField (section 3.1.5.27) object.

#### 3.1.5.13.1.1.2 TermGuid

**Type:** CSOM String

**Accessibility:** Read/Write

Specifies a string representing Term (section 3.1.5.16) GUID.

**Remarks:**

If an invalid internal identifier was specified when the TaxonomyFieldValue object was constructed, this property will return an empty string.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>MemberAccessException</td>
<td>The GUID of the Term (section 3.1.5.16) was not specified when the TaxonomyFieldValue (section 3.1.5.13) object was constructed and the TaxonomyFieldValue object is not bound</td>
</tr>
</tbody>
</table>
### 3.1.5.14.1.1 Scalar Properties

#### 3.1.5.14.1.1.1 CreatedDate

**Type:** CSOM DateTime

**Accessibility:** Read Only

Gets the date and time when this item was created.

#### 3.1.5.14.1.1.2 Id

**Type:** CSOM GUID

**Accessibility:** Read Only

Gets the identifier of the current TaxonomyItem (section 3.1.5.14).

#### 3.1.5.14.1.1.3 LastModifiedDate

**Type:** CSOM DateTime

**Accessibility:** Read Only

Gets the date and time when this item was most recently updated.
3.1.5.14.1.1.4 Name
Type: CSOM String
Accessibility: Read/Write
Specifies the name of the current TaxonomyItem (section 3.1.14) object.

3.1.5.14.1.2 ObjectPath Properties

3.1.5.14.1.2.1 TermStore
Type: Microsoft.SharePoint.Client.Taxonomy.TermStore
Accessibility: Read Only
Gets the TermStore (section 3.1.5.23) object from which the current TaxonomyItem (section 3.1.14) was retrieved.

3.1.5.14.2 Methods

3.1.5.14.2.1 Scalar Methods

3.1.5.14.2.1.1 DeleteObject
Return Type: None
Deletes the current TaxonomyItem (section 3.1.14) object.
Parameters: None

3.1.5.14.2.1.2 NormalizeName
This method is a static CSOM method.
Return Type: CSOM String
Normalizes a name of a TaxonomyItem (section 3.1.14) object.
Parameters:
name: The name that the NormalizeName normalizes.
Type: CSOM String
Remarks:
The name is normalized to trim consecutive spaces into one space and replace the ampersand (&) character with the wide character version of the character: ``(\uFF06)``.

3.1.5.14.2.2 ObjectPath Methods
None.

3.1.5.15 Microsoft.SharePoint.Client.Taxonomy.TaxonomySession
TypeId: `{981CBC68-9EDC-4F8D-872F-71146FCBB84F}`
ShortName: SP.Taxonomy.TaxonomySession
The container for all of the associated TermStore (section 3.1.5.23) objects for a site collection.

3.1.5.15.1 Properties

3.1.5.15.1.1 Scalar Properties

3.1.5.15.1.1.1 OfflineTermStoreNames

Type: CSOM array of CSOM String

Accessibility: Read Only

Gets an array of string values representing the names of TermStore (section 3.1.5.23) objects that are currently offline.

3.1.5.15.1.2 ObjectPath Properties

3.1.5.15.1.2.1 TermStores

Type: Microsoft.SharePoint.Client.Taxonomy.TermStoreCollection

Accessibility: Read Only

A collection of TermStore (section 3.1.5.23) objects that are associated with a Web application of the current site collection.

3.1.5.15.2 Methods

3.1.5.15.2.1 Scalar Methods

3.1.5.15.2.1.1 UpdateCache

Return Type: None

This method forces the taxonomy cache to communicate with the Managed Metadata Service to check for changes to any taxonomy objects.

Remarks:

A background thread already performs these checks at periodic intervals, so calling UpdateCache is normally unnecessary. However, it is sometimes useful immediately before a change (to reduce the likelihood of save conflict errors), or immediately after a change (if that change has been performed via a different Web Front End (WFE)).

This method updates the cache for each TermStore (section 3.1.5.23) object.

Parameters: None

3.1.5.15.2.2 ObjectPath Methods

3.1.5.15.2.2.1 GetTerm

Return Type: Microsoft.SharePoint.Client.Taxonomy.Term

Gets a Term (section 3.1.5.16) object that is based on the Id (section 3.1.5.16.1.1.4) property.

Remarks:
If the current Term (section 3.1.5.16) is part of multiple TermSet (section 3.1.5.20) objects, the server returns an arbitrary Term object from one of the TermSet objects. The server MUST return null if the Term cannot be found or if the user does not have sufficient permissions to access it.

Parameters:

termId: The Id (section 3.1.15.16.1.1.4) property of the Term (section 3.1.5.16) to get.

Type: CSOM GUID

3.1.5.15.2.2.2 GetTerms

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermCollection

Gets a collection of Term (section 3.1.15.16) objects from all TermSet (section 3.1.5.20) objects, which are not scoped to a Site ([MS-CSOMSPT] section 3.2.5.119) object, from all TermStore (section 3.1.5.23) objects, based on the label criteria provided.

Remarks:

The LabelMatchInformation values that are set MUST be one of these combinations:

- LabelMatchInformation.TermLabel and LabelMatchInformation.TrimUnavailable.
- LabelMatchInformation.TermLabel, LabelMatchInformation.Lcid, and LabelMatchInformation.TrimUnavailable.

If the combination of LabelMatchInformation (section 3.1.5.30) values is invalid, an ArgumentException or ArgumentNullException will be thrown indicating the invalid setting.

The LabelMatchInformation.TermLabel (section 3.1.5.30.1.1.6) cannot be null or empty, and MUST NOT exceed 255 characters in length. It also cannot contain any of the following illegal characters: ; " < > | & tab

The search is case-insensitive, and will match either the entire label or only check that it starts with the specified string depending on the value of the LabelMatchInformation.StringMatchOption (section 3.1.5.30.1.1.5) parameter. If the LabelMatchInformation.StringMatchOption is not set, it defaults to exact match.

If LabelMatchInformation.TrimUnavailable (section 3.1.5.30.1.1.8) is true, then terms that have Term.IsAvailableForTagging (section 3.1.5.16.1.1.5) set to false will be trimmed from the results; otherwise, all terms will be returned regardless of their Term.IsAvailableForTagging value.

If LabelMatchInformation.Lcid (section 3.1.5.30.1.1.3) is specified, then a Term will be returned if a Label in the specified Lcid matches the string provided. If no Lcid is specified, then all Term.Labels (section 3.1.5.16.1.2.1) in all working languages will be searched by this method. The Lcid value MUST be one of the working languages of the term store, specified by TermStore.Languages (section 3.1.5.23.1.1.4).

If LabelMatchInformation.DefaultLabelOnly (section 3.1.5.30.1.1.1) is true, then only the default labels will be searched.
The maximum number of results returned from this search is restricted by the `LabelMatchInformation.ResultCollectionSize` (section 3.1.5.30.1.1.4) parameter. If `LabelMatchInformation.ResultCollectionSize` is not set, then a maximum of 100 results is returned.

If `LabelMatchInformation.TrimDeprecated` (section 3.1.5.30.1.1.7) is true, then terms that have `Term.IsDeprecated` (section 3.1.5.16.1.1.6) set to true will be trimmed from the results; otherwise, all terms will be returned regardless of their `Term.IsDeprecated` value.

To get the most accurate results, it is recommended that any pending changes be committed to the database using `CommitAll` (section 3.1.5.23.2.1.1) prior to executing this search.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>Incorrect combination of <code>LabelMatchInformation</code> settings provided.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>Invalid value provided for one of the <code>LabelMatchInformation</code> settings.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of <code>LabelMatchInformation.TermLabel</code> is invalid. It either contains invalid characters or is too long.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>Some <code>LabelMatchInformation</code> settings cannot be null or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The <code>LabelMatchInformation.TermLabel</code> cannot be null or empty.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentException</td>
<td>The <code>LabelMatchInformation.Lcid</code> value MUST be one of the working languages in the current <code>TermStore</code>.</td>
</tr>
</tbody>
</table>

**Parameters:**

`labelMatchInformation`: Indicates the search criteria to use when retrieving `Term` objects.

Type: Microsoft.SharePoint.Client.Taxonomy.LabelMatchInformation

**3.1.5.15.2.2.3 GetTermsWithCustomProperty**

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermCollection

Gets a collection of all of the `Term` (section 3.1.5.16) objects, from all `TermStore` (section 3.1.5.23) objects, that have a custom property matching the criteria provided.

**Remarks:**

The `CustomPropertyMatchInformation` (section 3.1.5.29) values that are set MUST be one of these combinations:

- `CustomPropertyMatchInformation.CustomPropertyName` and `CustomPropertyMatchInformation.TrimUnavailable`.
- `CustomPropertyMatchInformation.CustomPropertyName`, `CustomPropertyMatchInformation.ResultCollectionSize`, and `CustomPropertyMatchInformation.TrimUnavailable`. 
CustomPropertyMatchInformation.CustomPropertyName, CustomPropertyMatchInformation.CustomPropertyValue, and CustomPropertyMatchInformation.TrimUnavailable.

CustomPropertyMatchInformation.CustomPropertyName, CustomPropertyMatchInformation.CustomPropertyValue, CustomPropertyMatchInformation.StringMatchOption, and CustomPropertyMatchInformation.TrimUnavailable.

CustomPropertyMatchInformation.CustomPropertyName, CustomPropertyMatchInformation.CustomPropertyValue, CustomPropertyMatchInformation.StringMatchOption, CustomPropertyMatchInformation.ResultCollectionSize, and CustomPropertyMatchInformation.TrimUnavailable.

Only Term objects within TermSet objects that are not scoped to a Site ([MS-CSOMSPT] section 3.2.5.119) and TermSet objects that are scoped to the current Site are searched.

If the combination of CustomPropertyMatchInformation values is invalid, an ArgumentException or ArgumentNullException will be thrown indicating the invalid setting.

The CustomPropertyMatchInformation.CustomPropertyName (section 3.1.5.29.1.1.1) MUST NOT be set to null or empty, and MUST NOT exceed 255 characters in length. It also MUST NOT contain any of the following illegal characters: ; " < > | & tab

The CustomPropertyMatchInformation.CustomPropertyValue (section 3.1.5.29.1.1.2) MUST NOT be set to null or empty, and MUST NOT exceed 255 characters in length. It also MUST NOT contain the illegal characters \t or \n.

The search is case-insensitive, and will match either the entire value or the custom property name or value, or only check that it starts with the specified string depending on the value of the CustomPropertyMatchInformation.StringMatchOption (section 3.1.5.29.1.1.4) parameter. If the CustomPropertyMatchInformation.StringMatchOption is not set, it defaults to exact match.

The CustomPropertyMatchInformation.ResultCollectionSize (section 3.1.5.29.1.1.3) value MUST be greater than 0. The maximum number of results returned from this search is restricted by the CustomPropertyMatchInformation.ResultCollectionSize parameter. If CustomPropertyMatchInformation.ResultCollectionSize is not set, then a maximum of 100 results is returned.

If CustomPropertyMatchInformation.TrimUnavailable (section 3.1.5.29.1.1.5) is set to true, then Term objects that have Term.IsAvailableForTagging (section 3.1.5.16.1.1.5) set to false will be trimmed from the results; otherwise, all matching terms will be returned regardless of their Term.IsAvailableForTagging value.

To get the most accurate results, it is recommended that any pending changes be committed to the database using CommitAll (section 3.1.5.23.2.1.1) prior to executing this search.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>Incorrect combination of CustomPropertyMatchInformation settings provided.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of CustomPropertyMatchInformation.CustomPropertyName is invalid. It either contains invalid characters or is too long.</td>
</tr>
</tbody>
</table>

43 / 156
### Error Code and Condition

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of <code>CustomPropertyMatchInformation.CustomPropertyValue</code> is invalid. It exceeds the maximum allowed length of 255.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of <code>CustomPropertyMatchInformation.CustomPropertyValue</code> is invalid. It contains invalid character <code>\t</code>, <code>\n</code>, or both.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>Some <code>CustomPropertyMatchInformation</code> settings MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The <code>CustomPropertyMatchInformation.CustomPropertyName</code> MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The <code>CustomPropertyMatchInformation.CustomPropertyValue</code> MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The <code>CustomPropertyMatchInformation.ResultCollectionSize</code> is invalid. It MUST be greater than 0.</td>
</tr>
</tbody>
</table>

### Parameters:

`customPropertyMatchInformation`: Indicates the search criteria to use when retrieving **Term** objects.

Type: `Microsoft.SharePoint.Client.Taxonomy.CustomPropertyMatchInformation`

#### 3.1.5.15.2.2.4 GetTermsInDefaultLanguage

**Return Type:** `Microsoft.SharePoint.Client.Taxonomy.TermCollection`

The `GetTermsInDefaultLanguage` method searches **Label** objects (section 3.1.5.10) in the **TermStore** (section 3.1.5.23) default language, and returns **Term** (section 3.1.5.16) objects from all **TermSet** objects (section 3.1.5.20) that are not scoped to a **site** from all **TermStore** objects with a **Label** matching the provided string.

**Parameters:**

`termLabel`: The string that is searched by `GetTermsInDefaultLanguage`.

Type: **CSOM String**

`defaultLabelOnly`: This parameter is "true" if the method only searches default **Label** objects; otherwise, "false" if it searches all **Label** objects.

Type: **CSOM Boolean**

`stringMatchOption`: Indicates how the string is matched, as specified by `StringMatchOption` (section 3.1.5.12).

Type: `Microsoft.SharePoint.Client.Taxonomy.StringMatchOption`

`resultCollectionSize`: The maximum number of **Term** objects to return from each **TermStore** object.
Type: CSOM Int32

trimUnavailable: A Boolean indicating whether to trim out Term objects that have the IsAvailableForTagging (section 3.1.5.20.1.1.7) property set to false.

Type: CSOM Boolean

trimDeprecated: This parameter is "true" if the method trims away the deprecated Term objects from the result collection; otherwise, "false" if it includes the deprecated Term objects in the result collection.

Type: CSOM Boolean

3.1.5.15.2.2.5 GetTermsInWorkingLocale

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermCollection

The GetTermsInWorkingLocale method searches Label objects (section 3.1.5.10) in the TermStore (section 3.1.5.23) working language, and returns Term objects (section 3.1.5.16) from all TermSet objects (section 3.1.5.20) that are not scoped to a site from all TermStore objects with a Label matching the provided string.

Parameters:

termLabel: The string that GetTermsinWorkingLocale searches.

Type: CSOM String

defaultLabelOnly: This parameter is "true" if the method only searches default Label objects; "false" if it searches all Label objects.

Type: CSOM Boolean

stringMatchOption: Indicates how the string is matched, as specified by StringMatchOption (section 3.1.5.12).

Type: Microsoft.SharePoint.Client.Taxonomy.StringMatchOption

resultCollectionSize: The maximum number of Term objects to return from each TermStore.

Type: CSOM Int32

trimUnavailable: A Boolean indicating whether to trim out Term objects that have the IsAvailableForTagging (section 3.1.5.20.1.1.7) property set to false.

Type: CSOM Boolean

trimDeprecated: This parameter is "true" if the method trims away the deprecated term Term (section 3.1.5.16) objects from the result collection; otherwise, "false" if it includes the deprecated Term objects in the result collection.

Type: CSOM Boolean

3.1.5.15.2.2.6 GetDefaultKeywordsTermStore

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermStore

Gets the TermStore (section 3.1.5.23) object to be used for keywords. This value is specified on the metadata application proxy connection. If multiple proxies associated with a Web application in a site are marked as the default keywords TermStore, then the server MUST return NULL because only
a single default keywords TermStore can exist. If no TermStore has been selected as the default, then the server MUST return NULL.

**Parameters:** None

### 3.1.5.15.2.2.7 GetDefaultSiteCollectionTermStore

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermStore

Gets the TermStore (section 3.1.5.23) object to be used for metadata that is specific to the site collection, such as column-specific TermSet (section 3.1.5.20) objects. This value is specified on the metadata application proxy connection. If multiple proxies associated with a Web application in a site are marked as the default keywords TermStore, then the server MUST return NULL because only a single default keywords TermStore can exist.

**Parameters:** None

### 3.1.5.15.2.2.8 GetTaxonomySession

This method is a **static CSOM method**.

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TaxonomySession

This method retrieves a TaxonomySession (section 3.1.5.15) object for performing taxonomy-related operations.

The TaxonomySession returned is based on the in-context Site ([MS-CSOMSPT] section 3.2.5.119). The client-side object model supports only one TaxonomySession instance for a given client context. If this method is called multiple times, it will return the same object.

Unlike the server-side object model, this method does not provide a parameter to update the cache. That can be accomplished by explicitly calling TaxonomySession.UpdateCache (section 3.1.5.15.2.1.1).

**Parameters:** None

### 3.1.5.15.2.2.9 GetTermsById

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermCollection

Gets a set of Term (section 3.1.5.16) objects based on Term.Id (section 3.1.5.16.1.1.4). If a Term is part of more than one TermSet (section 3.1.5.20) object, it will arbitrarily return the Term from one of the TermSet objects.

**Parameters:**

- `termIds`: A list of Term (section 3.1.5.16) Ids.
  
  Type: **CSOM array of CSOM GUID**

### 3.1.5.15.2.2.10 GetTermSetsByName

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermSetCollection

Gets all TermSet (section 3.1.5.20) objects from all TermStore (section 3.1.5.23) objects that have a name in the specified language.

**Parameters:**

- `termSetName`: The name of the TermSet object.
Type: **CSOM String**

*lcid*: LCID of the language.

Type: **CSOM Int32**

### 3.1.5.15.2.2.11 GetTermSetsByTermLabel

**Return Type**: Microsoft.SharePoint.Client.Taxonomy.TermSetCollection

Gets all of the **TermSet** (section 3.1.5.20) objects from all of the **TermStore** (section 3.1.5.23) objects that contain **Term** (section 3.1.5.16) objects with matching LCID-specific labels for all specified strings.

**Parameters**:

requiredTermLabels: A list of strings to search.

Type: **CSOM array of CSOM String**

*lcid*: The language to search in.

Type: **CSOM Int32**

### 3.1.5.16 Microsoft.SharePoint.Client.Taxonomy.Term

**TypeId**: `{5B8C81B7-7CD2-40DC-8525-5ECA12A4EB73}`

**ShortName**: SP.Taxonomy.Term

Represents a **term** or a **keyword** in a managed metadata **hierarchy**.

**Remarks**:

A **Term** is stored as a **TermSetItem** (section 3.1.5.22) object in the **TermStore** (section 3.1.5.23) object. It contains references to the parts that define a **TermSetItem**.

A **Term** can be reused under different parent objects. It can have custom properties associated with it. It can have multiple **Label** (section 3.1.5.10) objects associated with it for different **language code identifiers** (LCIDs).

When the **IsKeyword** (section 3.1.5.16.1.1.7) property is set to true, the **Term** represents a keyword. A keyword does not have any child objects and is stored in a keyword **TermSet** (section 3.1.5.20).

Initialize a new instance of this class by using the **CreateTerm** (section 3.1.5.16.2.2.3) method, the static **GetTerm** (section 3.1.5.20.2.2.5) method, or by getting it from a **TermCollection** (section 3.1.5.17) object.

### 3.1.5.16.1 Properties

#### 3.1.5.16.1.1 Scalar Properties

#### 3.1.5.16.1.1.1 CreatedDate

**Type**: **CSOM DateTime**

**Accessibility**: Read Only

Gets the date and time when this item was created.
3.1.5.16.1.1.2  CustomProperties

**Type:** CSOM dictionary

**Accessibility:** Read Only

Gets the collection of custom property objects from the read-only dictionary of the Term (section 3.1.5.16) object.

3.1.5.16.1.1.3  CustomSortOrder

**Type:** CSOM String

**Accessibility:** Read/Write

Specifies the custom sort order of the current Term (section 3.1.5.16) object.

The return value is a string that is composed of the child Term GUIDs and separated by a semicolon (‘:’). When the custom sort order is null or empty, the child Term objects of the current Term are sorted alphabetically based on the language of the current TermStore (section 3.1.5.23) object.

**Remarks:**

If the custom sort-order value contains an invalid GUID, or if one of the GUIDs does not belong to its child Term objects, then it is ignored. If any GUID that belongs to a child Term is not included, then it will be sorted alphabetically following the Term objects with custom sort order.

Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set (section 3.1.5.20) object. The operation failed.</td>
</tr>
</tbody>
</table>

3.1.5.16.1.1.4  Id

**Type:** CSOM GUID

**Accessibility:** Read Only

Gets the identifier of the current TaxonomyItem (section 3.1.5.14).

3.1.5.16.1.1.5  IsAvailableForTagging

**Type:** CSOM Boolean

**Accessibility:** Read/Write

Specifies the flag indicating whether a Term (section 3.1.5.16) object can be used to tag in the tagging user interface. The tagging user interface is a user interface for adding terms as tags to other items.

**Remarks:**
Call the **CommitAll** (section 3.1.5.23.2.1.1) method to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set (section 3.1.5.20) object. The operation failed.</td>
</tr>
</tbody>
</table>

### 3.1.5.16.1.1.6 IsDeprecated

**Type:** CSOM Boolean  
**Accessibility:** Read Only

Gets information about whether the current Term (section 3.1.5.16) object has been deprecated.

### 3.1.5.16.1.1.7 IsKeyword

**Type:** CSOM Boolean  
**Accessibility:** Read Only

Gets information about whether the current Term object is a keyword in the keyword TermSet (section 3.1.5.20).

### 3.1.5.16.1.1.8 IsReused

**Type:** CSOM Boolean  
**Accessibility:** Read Only

Gets information about whether the current Term (section 3.1.5.16) object is a reused term.

**Remarks:**

If the current Term is reused, there is at least one member in the ReusedTerms (section 3.1.5.16.1.2.3) collection. Use the SourceTerm (section 3.1.5.16.1.2.4) property to find out if the current Term is the source Term for reuse.

### 3.1.5.16.1.1.9 IsRoot

**Type:** CSOM Boolean  
**Accessibility:** Read Only

Gets information about whether the current Term (section 3.1.5.16) object is at the root of the TermSet (section 3.1.5.20) object.

Returns `true` if the current Term is at the root of TermSet; otherwise `false`.

### 3.1.5.16.1.1.10 IsSourceTerm

**Type:** CSOM Boolean
**Accessibility:** Read Only

Gets information about whether the current Term (section 3.1.5.16) is the source Term among reused terms.

**Remarks:**

If the **IsReused** (section 3.1.5.16.1.1.8) property is **false**, the current Term is not reused and this property will return **true**. If the current Term is reused, then this property returns **true** if the current Term is also the source Term.

**3.1.5.16.1.1.11 LastModifiedDate**

**Type:** CSOM DateTime

**Accessibility:** Read Only

Gets the date and time when this item was most recently updated.

**3.1.5.16.1.1.12 Name**

**Type:** CSOM String

**Accessibility:** Read/Write

Specifies the name of the current Term (section 3.1.5.16) object.

**Remarks:**

The name is the default **Label** (section 3.1.5.10) of the working language, as specified by **WorkingLanguage** (section 3.1.5.23.1.1.6) property, of the current TermStore (section 3.1.5.23).

The **Name** value will be normalized to trim consecutive spaces into one and replace any "&" characters with the wide character version of the character ("uFF06"). It MUST be non-empty, cannot exceed 255 characters, and cannot contain any of the following characters: ;, "", <, >, |, & , or the tab-character.

Call the **CommitAll** (section 3.1.5.23.2.1.1) to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set (section 3.1.5.20) object. The operation failed.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The name is not valid because it either contains invalid characters or is too long.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The name is null or an empty string.</td>
</tr>
</tbody>
</table>

**3.1.5.16.1.1.13 Owner**

**Type:** CSOM String
**Accessibility:** Read/Write

Specifies the owner of the current Term (section 3.1.5.16) object.

**Remarks:**

The Owner property MUST NOT be empty or null. It cannot exceed 255 characters.

Call the CommitAll (section 3.1.5.23.2.1.1) to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set (section 3.1.5.20) object. The operation failed.</td>
</tr>
</tbody>
</table>

3.1.5.16.1.1.14  TermsCount

**Type:** CSOM Int32

**Accessibility:** Read Only

Gets the number of child Term (section 3.1.5.16) objects.

3.1.5.16.1.1.15  IsPinned

**Type:** CSOM Boolean

**Accessibility:** Read Only

Specifies whether this Term (section 3.1.5.16) is pinned.

3.1.5.16.1.1.16  LocalCustomProperties

**Type:** CSOM dictionary

**Accessibility:** Read Only

Specifies the custom properties that have been overridden for this Term (section 3.1.5.16).

3.1.5.16.1.1.17  MergedTermIds

**Type:** CSOM array of CSOM GUID

**Accessibility:** Read Only

Specifies a read-only collection of term identifiers. Term (section 3.1.5.16) objects identified by the returned identifiers have been merged into the current Term.

3.1.5.16.1.1.18  Description

**Type:** CSOM String
**Accessibility:** Read Only

Specifies the description of the Term (section 3.1.5.16) in the same language as Name (section 3.1.5.16.1.1.12). If the Term does not have a description, an empty string is returned.

### 3.1.5.16.1.19 PathOfTerm

**Type:** CSOM String

**Accessibility:** Read Only

Gets the path for this Term (section 3.1.5.16) in the TermStore (section 3.1.5.23) default language.

### 3.1.5.16.1.20 IsPinnedRoot

**Type:** CSOM Boolean

**Accessibility:** Read Only

Indicates whether this term is pinned and is the root of the pinned tree.

### 3.1.5.16.1.2 ObjectPath Properties

#### 3.1.5.16.1.2.1 Labels

**Type:** Microsoft.SharePoint.Client.Taxonomy.LabelCollection

**Accessibility:** Read Only

Specifies a collection of Label (section 3.1.5.10) objects for the current Term (section 3.1.5.16) object.

**Remarks:**

The collection of Label objects is returned for all working languages defined by the Languages (section 3.1.5.23.1.1.4) property.

#### 3.1.5.16.1.2.2 Parent

**Type:** Microsoft.SharePoint.Client.Taxonomy.Term

**Accessibility:** Read Only

Specifies the parent Term (section 3.1.5.16) object of the current Term.

This value is null if the Term is the root Term, or if the protocol client does not have permissions to access the parent Term.

**Remarks:**

This value is specific to the current TermSet (section 3.1.5.20) object.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147467261</td>
<td>ArgumentException</td>
<td>A null value was passed as the Term.</td>
</tr>
</tbody>
</table>
3.1.5.16.1.2.3 ReusedTerms

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermCollection

**Accessibility:** Read Only

Specifies all of the instances of the current **Term** (section 3.1.5.16) object that are reused in other **TermSet** (section 3.1.5.20) objects.

**Remarks:**

The returned collection of **Term** objects does not include the current **Term**.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>A null value was passed as the ** Term**.</td>
</tr>
</tbody>
</table>

3.1.5.16.1.2.4 SourceTerm

**Type:** Microsoft.SharePoint.Client.Taxonomy.Term

**Accessibility:** Read Only

Specifies the source **Term** (section 3.1.5.16) object if the current **Term** is a reused term.

**Remarks:**

Returns null if the protocol client does not have permissions to access the source **Term**.

3.1.5.16.1.2.5 Terms

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermCollection

**Accessibility:** Read Only

Specifies a collection of all immediate child **Term** (section 3.1.5.16) objects of the current **Term** object.

**Remarks:**

The returned collection of terms is sorted based on the custom sort order as specified by **CustomSortOrder** (section 3.1.5.16.1.1.3). If no custom sort order is defined, then it is sorted by **Label** (section 3.1.5.10) based on the alphabetical order that is determined by the **WorkingLanguage** (section 3.1.5.23.1.1.6) object.

3.1.5.16.1.2.6 TermSet

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermSet

**Accessibility:** Read Only

Specifies the **TermSet** (section 3.1.5.20) object for the current **Term** (section 3.1.5.16) object.

**Exceptions:**
### 3.1.5.16.1.2.7 TermSets

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermSetCollection

**Accessibility:** Read Only

Specifies the collection of all **TermSet** (section 3.1.5.20) objects where the current **Term** (section 3.1.5.16) object is either stored or reused.

**Remarks:**

The returned collection includes the **TermSet** object where the current **Term** is stored, and all of the **TermSet** objects where the **Term** is reused.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The <strong>TermSet</strong> of the <strong>Term</strong> is not valid.</td>
</tr>
</tbody>
</table>

### 3.1.5.16.1.2.8 TermStore

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermStore

**Accessibility:** Read Only

Specifies the **TermStore** (section 3.1.5.23) object from which the current **TaxonomyItem** (section 3.1.5.14) was retrieved.

### 3.1.5.16.1.2.9 PinSourceTermSet

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermSet

**Accessibility:** Read Only

Specifies the **TermSet** (section 3.1.5.20) where the **Term** (section 3.1.5.16) that this **Term** is pinned to exists. If this **Term** is not pinned, returns **null**.

### 3.1.5.16.2 Methods

#### 3.1.5.16.2.1 Scalar Methods

**3.1.5.16.2.1.1 DeleteCustomProperty**

**Return Type:** None

Deletes a custom property from the current **Term** (section 3.1.5.16) object.

**Parameters:**
name: The name of the custom property.

Type: **CSOM String**

**Remarks:**

The name MUST NOT be empty, exceed 255 characters, or contain any of the following characters: ; " < > | & or the tab character.

Call the **CommitAll** (section 3.1.5.23.2.1.1) to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The operation is not valid in the orphaned terms term set (section 3.1.5.20). The operation failed.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The custom property name is not valid: it contains characters that are not valid, or is too long.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentException</td>
<td>The custom property name is null or an empty string.</td>
</tr>
</tbody>
</table>

### 3.1.5.16.2.1.2 **DeleteObject**

**Return Type:** None

Deletes this **Term** (section 3.1.16) object and its child objects from the **TermStore** (section 3.1.5.23) object.

**Remarks:**

Deleting a source **Term** while it is still reused elsewhere causes the source **Term** to be moved to the orphaned terms term set object, and the **Term** becomes an orphan. Orphaned **Term** objects cannot be directly deleted. When all reused **Term** objects are deleted, the orphan is also deleted.

Call the **CommitAll** (section 3.1.5.23.2.1.1) to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The operation is not valid in orphaned terms term set (section 3.1.5.20). The operation failed.</td>
</tr>
</tbody>
</table>

**Parameters:** None
3.1.5.16.2.1.3  Deprecate

Return Type: None

Deprecates or enables the current Term (section 3.1.16) object.

Parameters:

doDeprecate: Indicates whether to deprecate or enable this Term.

Type: CSOM Boolean

Remarks:

Call the CommitAll (section 3.1.5.23.2.1.1) to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The operation is not valid in the orphaned terms term set (section 3.1.5.20). The operation failed.</td>
</tr>
</tbody>
</table>

3.1.5.16.2.1.4  GetDefaultLabel

Return Type: CSOM String

Specifies the default Label (section 3.1.5.10) for this Term (section 3.1.16) based on the language code identifier (LCID).

Parameters:

lcid: The LCID the default Label is associated with.

Type: CSOM Int32

Remarks:

Returns an empty string when there is no default Label for the given LCID.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The lcid is not valid.</td>
</tr>
</tbody>
</table>

3.1.5.16.2.1.5  GetDescription

Return Type: CSOM String

Specifies the description of the Term object in the language (WorkingLanguage (section 3.1.5.23.1.1.6)) of the current TermStore (section 3.1.5.23) object.

Parameters:
lcid: The description of the language code identifier (LCID).

Type: CSOM Int32

Remarks:
Returns the description of the Term for the working language or the current TermStore, or an empty string if the Term has no description in that language.

3.1.5.16.2.1.6 GetPath

Return Type: CSOM String

Specifies the path for the current Term (section 3.1.16) in the specified language code identifier (LCID).

Parameters:
lcid: The LCID for the default Label (section 3.1.5.10) object

Type: CSOM Int32

Remarks:
The path is composed of the default Label objects of the ancestor Term objects separated by semicolons (;).

If the lcid does not specify a valid TermStore (section 3.1.5.23) language, then the default language is used.

3.1.5.16.2.1.7 Move

Return Type: None

Moves the current Term (section 3.1.5.16) object to be a child of a different Term.

Parameters:
newParent: The Term that becomes the new parent after the current Term object is moved after it.

Type: Microsoft.SharePoint.Client.Taxonomy.TermSetItem

Remarks:
Call the CommitAll (section 3.1.5.23.2.1.1) to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The new parent Term is null.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The new parent Term is not valid.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The keyword TermSet (section 3.1.5.20) only allows a flat list of Term objects.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Cannot move a Term to itself.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Error Type Name</td>
<td>Condition</td>
</tr>
<tr>
<td>------------</td>
<td>--------------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Cannot move a Term to its descendent Term objects.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is not valid in the orphaned terms term set. The operation failed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Cannot move a Term to a TermSet where other reused Term objects are not allowed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Cannot move a Term to a TermSet that has another reused instance of this Term.</td>
</tr>
</tbody>
</table>

### 3.1.5.16.2.1.8 ReassignSourceTerm

**Return Type:** None

Reassigns the source Term status to one of the other reused instances of this Term (section 3.1.5.16) object.

**Parameters:**

reusedTerm: One of the Term objects from the ReusedTerms (section 3.1.5.16.1.2.3) collection that becomes the new source term.

Type: Microsoft.SharePoint.Client.Taxonomy.Term

**Remarks:**

This method is valid only if the current Term is the source Term. Use this method before attempting to delete a source Term.

If the source Term is in the orphaned terms term set and the operation succeeded, then the old source Term is removed from the orphaned terms term set. Call the CommitAll (section 3.1.5.23.2.1.1) to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The reused Term to become the new source is null or not valid.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The current Term MUST be the source Term to reassign the current setting to another Term.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The source Term cannot be reassigned to itself.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The reusedTerm parameter MUST be a member of the ReusedTerms collection for this Term.</td>
</tr>
</tbody>
</table>

### 3.1.5.16.2.1.9 SetCustomProperty
**Return Type:** None

Adds a new custom property or updates the value of an existing property.

**Parameters:**

- `name`: The custom property name to set.
  
  **Type:** CSOM String

- `value`: The value to set for the specified custom property.
  
  **Type:** CSOM String

**Remarks:**

This method tries to find the specified custom property name and set it with the specified value. The property names do not need to be the same case to match. If the custom property name does not exist, then a new property is created.

The `name` **MUST** be non-empty, cannot exceed 255 characters, and cannot contain any of the following characters: `; " < > | &` or the tab character.

The `value` cannot exceed 1000 characters.

The leading and trailing space of the custom property name and value will be trimmed.

Call the [CommitAll](#) (section 3.1.5.23.2.1.1) to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is not valid in the <strong>orphaned terms term set</strong>. The operation failed.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The custom property name is not valid. It contains characters that are not valid or is too long.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The custom property name is null or an empty string.</td>
</tr>
</tbody>
</table>

### 3.1.5.16.2.1.10 SetDescription

**Return Type:** None

Sets the description specified by the **language code identifier (LCID)** of this Term (section 3.1.5.16) object.

**Parameters:**

- `description`: The new description specified by the LCID.
  
  **Type:** CSOM String

- `lcid`: The LCID to set `description` for.
Type: **CSOM Int32**

**Remarks:**

The description MUST NOT be empty, exceed 1000 characters, or contain the tab character.

Call the **CommitAll** (section 3.1.5.23.2.1.1) to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is not valid in the orphaned terms term set. The operation failed.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The custom property name is not valid. It contains characters that are not valid or is too long.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The custom property name is null or an empty string.</td>
</tr>
</tbody>
</table>

3.1.5.16.2.1.11 **DeleteAllCustomProperties**

**Return Type:** None

Deletes all shared custom properties for this **Term** (section 3.1.5.16).

**Parameters:** None

3.1.5.16.2.1.12 **DeleteAllLocalCustomProperties**

**Return Type:** None

Deletes all local custom properties for this **Term** (section 3.1.5.16).

**Parameters:** None

3.1.5.16.2.1.13 **DeleteLocalCustomProperty**

**Return Type:** None

Deletes a local custom property for this **Term** (section 3.1.5.16).

**Parameters:**

*name*: Name of the property to be deleted.

Type: **CSOM String**

3.1.5.16.2.1.14 **GetIsDescendantOf**

**Return Type:** **CSOM Boolean**

Indicates whether the current **Term** (section 3.1.5.16) object is a descendent **Term** of the specified **Term**.

**Parameters:**
ancestorTerm: The **Term** to verify is or is not an ancestor.

Type: Microsoft.SharePoint.Client.Taxonomy.Term

Returns **true** if the **Term** is a descendent of the specified **Term**; otherwise **false**.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The <strong>Term</strong> to check is null.</td>
</tr>
</tbody>
</table>

### 3.1.5.16.2.1.15 NormalizeName

This method is a **static CSOM method**.

**Return Type:** CSOM String

Normalizes a name of a **TaxonomyItem** (section 3.1.5.14) object.

**Parameters:**

- **name**: The name to normalize
  
  Type: CSOM String

### 3.1.5.16.2.1.16 SetLocalCustomProperty

**Return Type:** None

Adds a new local custom property if one with the same name does not already exists. If a local property with the same name already exists, this method updates that property with the value indicated in the **value** parameter.

**Parameters:**

- **name**: Name of the property to be added or updated.
  
  Type: **CSOM String**

- **value**: Value of the property to be added or updated.
  
  Type: CSOM String

### 3.1.5.16.2.2 ObjectPath Methods

#### 3.1.5.16.2.2.1 Copy

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.Term

Creates and returns a sibling copy of the current **Term** (section 3.1.5.16) object within the current **TermSet** (section 3.1.5.20) object.

**Parameters:**

- **doCopyChildren**: Indicates whether to copy this **Term** and all of its child terms.
  
  Type: **CSOM Boolean**
Remarks:
The copied Term will have the same properties as the current Term, except for the Name (section 3.1.5.16.1.1.12) property. If the Term is copied with its child Term objects, the newly-created child terms will have the same properties as the child terms of the current Term, including the Name property. The copied child terms will preserve the same hierarchy.

Call the CommitAll (section 3.1.5.23.2.1.1) to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is not valid in the orphaned terms term set. The operation failed.</td>
</tr>
</tbody>
</table>

3.1.5.16.2.2.2 CreateLabel

Return Type: Microsoft.SharePoint.Client.Taxonomy.Label

Creates a new Label (section 3.1.5.10) for this Term (section 3.1.5.16).

Parameters:

`labelName`: The new Label name.

Type: CSOM String

`lcid`: The language code identifier (LCID) to create the Label for.

Type: CSOM Int32

`isDefault`: Indicates whether this Label will be the default Label of the Lcid.

Type: CSOM Boolean

Remarks:

The `labelName` value will be normalized to trim consecutive spaces into one and replace the "&" character with the wide character version of the character (\uFF06). It MUST NOT be empty, exceed 255 characters, or contain any of the following characters: ; " < > | & or the tab character.

The name of the Label object is not case-sensitive. If the name to be created is a non-default Label, it MUST NOT have the same name as any existing Label of the Lcid. If it is a default Label, then it can have the same name as existing Label. Instead of creating a new Label, the existing Label will become the default Label for the Lcid.

Call the CommitAll (section 3.1.5.23.2.1.1) to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client does not have sufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Error Type Name</td>
<td>Condition</td>
</tr>
<tr>
<td>------------</td>
<td>---------------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is not valid in the orphaned terms term set (section 3.1.5.20). The operation failed.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The label name is not valid, it contains characters that are not valid, or is too long.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentException</td>
<td>The label name is null or an empty string.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The lcid is not a supported language in the term store.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The Label will be created as a non-default Label, and the name already exists for this lcid.</td>
</tr>
</tbody>
</table>

### 3.1.5.16.2.2.3 CreateTerm

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.Term

Creates a new Term (section 3.1.5.16) object under this TaxonomyItem (section 3.1.5.14) object. The name is the default Label (section 3.1.5.10) for the specified language code identifier (LCID).

**Parameters:**

- **name:** Default label for the lcid.
  - Type: **CSOM String**
  - *lcid:* The LCID to create the Label for.
    - Type: **CSOM Int32**
    - *newTermId:* The GUID for the Term.
      - Type: **CSOM GUID**

**Remarks:**

If the specified LCID is not the default language of the TermStore (section 3.1.5.23), a Label (section 3.1.5.10) with the name and the default language of the TermStore will also be created to ensure the Term has a default Label for the default language of the TermStore.

If the current TermSetItem (section 3.1.5.22) is the keyword TermSet, then the Label is created for the TermStore default language.

The Name value will be normalized to trim consecutive spaces into one and replace the "&" character with the wide character version of the character (\uFF06). It MUST NOT be empty, exceed 255 characters, or contain any of the following characters: ; " < > | & or the tab character.

The current user will be set as the owner of the new Term.

If the newTermId already exists in the TermStore, an exception will be thrown.

Call the CommitAll (section 3.1.5.23.2.1.1) to save this change to the database.

**Exceptions:**
Error Code | Error Type Name                       | Condition                                                                 |
------------|--------------------------------------|---------------------------------------------------------------------------|
-2147024891 | UnauthorizedAccessException          | The protocol client has insufficient permissions to perform this operation. |
-1          | TermStoreOperationException            | This operation is not valid in the orphaned terms term set (section 3.1.5.20). The operation failed. |
-1          | TermStoreOperationException            | Keyword term set (section 3.1.5.20) only allows a flat list of Term objects. |
-2147024809 | ArgumentException                   | The name is not valid, it contains characters that are not valid, or is too long. |
-2147467261 | ArgumentNullException                 | The name is null or empty string.                                          |
-2146233086 | ArgumentOutOfRangeException           | The lcid is not valid.                                                     |

### 3.1.5.16.2.2.4 GetAllLabels

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.LabelCollection

Specifies all of the Label (section 3.1.5.10) objects in the provided language code identifier (LCID).

**Parameters:**

lcid: The LCID that the Label objects are associated with.

Type: CSOM Int32

**Remarks:**

If Label objects exist for the specified lcid, the default Label is added first to the LabelCollection (section 3.1.5.11) object, followed by the non-default Labels.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The lcid is not valid.</td>
</tr>
</tbody>
</table>

### 3.1.5.16.2.2.5 GetTerms

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermCollection

The GetTerms method returns all Term (section 3.1.5.16) instances that are descendants of the current Term with a Label (section 3.1.5.10) in the provided language code identifier (LCID) and that match the Label name string provided. This method only checks default Labels if defaultLabelOnly is true.

**Parameters:**

termLabel: The term label name to match against.

Type: CSOM String
lcid: This method searches only labels in this LCID.

Type: **CSOM Int32**

defaultLabelOnly: Indicates whether to search only default labels.

Type: **CSOM Boolean**

stringMatchOption: Indicates whether the search is exact match or starts with the string.

Type: Microsoft.SharePoint.Client.Taxonomy.StringMatchOption

resultCollectionSize: Maximum number of results returned.

Type: CSOM Int32

trimUnavailable: Indicates whether to include not available for tagging Term in the results.

Type: CSOM Boolean

**Remarks:**

The termLabel value will be normalized to trim consecutive spaces into one, and replace the "&" character with the wide character version of the character (\uFF06). It MUST NOT be empty, exceed 255 characters, or contain any of the following characters: ; " < > | & or the tab character.

The search results are directly from TermStore (section 3.1.5.23). The results are not necessarily in sync with changes that have been made but not committed.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The lcid is not a supported language in the term store.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The termLabel is not valid: contains characters that are not valid or is too long.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The termLabel is null, or an empty string.</td>
</tr>
</tbody>
</table>

3.1.5.16.2.2.6 Merge

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.Term

Merges the current Term (section 3.1.5.16) object with the specified Term and returns the combined Term.

**Parameters:**

termToMerge: The Term to merge into.

Type: Microsoft.SharePoint.Client.Taxonomy.Term

**Remarks:**

The current Term is the Term to merge from, and the termToMerge is the Term to merge into, otherwise called the target Term. After the Merge operation is complete, both Term objects become reused, and the targetTerm becomes the source of the reused Term objects. The target Term GUID
becomes the GUID of the merged Term. The current Term GUID is added to MergedTermIds (section 3.1.5.16.1.1.17).

The Term GUID in the custom sort order of the parents of the current Term, or any of its reused copies, is updated with the target Term GUID.

The custom sort order is merged in the following way: if the target Term has a custom sort order, then the source Term sort order is appended to it. Otherwise, if the target Term does not have a custom sort order but the source Term does, then a custom sort order is enforced based on the alphabetical order in the current working language WorkingLanguage (section 3.1.5.23.1.1.6), and the source Term custom sort order is appended to it.

Call the CommitAll (section 3.1.5.23.2.1.1) to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The Term to merge into is null.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The Term to merge into is not valid.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is not valid in the orphaned terms term set (section 3.1.5.20). The operation failed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Cannot merge a Term to its reused Term.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Only a source Term can be merged into another source Term.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Merge is not allowed for the two specified Term objects. The source Term and the target Term of the merge operation are reused in the same TermSet object, and MUST be siblings in that TermSet so that they can be merged.</td>
</tr>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
</tbody>
</table>

3.1.5.16.2.2.7 ReuseTerm

ReturnType: Microsoft.SharePoint.Client.Taxonomy.Term

Creates a new reused Term (section 3.1.5.16) object under this TaxonomyItem (section 3.1.5.14) object based on a source Term.

Parameters:

sourceTerm: The source Term for the reused Term.

Type: Microsoft.SharePoint.Client.Taxonomy.Term

reuseBranch: Reuses the TermSetItem (section 3.1.5.22) and its child Term objects.

Type: CSOM Boolean

Remarks:
Reuses a term from a different TermSet to become a child of the current TermSetItem (section 3.1.5.22). The target TermSet cannot be the keyword or orphaned terms term set. Keywords cannot be reused.

Call the CommitAll (section 3.1.5.23.2.1.1) to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is not valid in the orphaned terms term set (section 3.1.5.20). The operation failed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Keywords are not allowed to be reused.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Reusing a deprecated Term is not allowed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Term objects cannot be shared multiple times in the same TermSet (section 3.1.5.20).</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The target TermSet cannot be the keywords term set or the orphaned terms term set.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The source Term is null or not valid</td>
</tr>
</tbody>
</table>

### 3.1.5.16.2.2.8 ReuseTermWithPinning

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.Term

Creates a new Term (section 3.1.5.16) under this TaxonomyItem (section 3.1.5.14) instance based on a source term and makes this Term pinned so that only the local custom properties, IsAvailableForTagging (section 3.1.5.20.1.1.7) and custom sort order for this term set can be changed in this term set.

**Parameters:**

sourceTerm: The source term.

Type: Microsoft.SharePoint.Client.Taxonomy.Term

### 3.1.5.17 Microsoft.SharePoint.Client.Taxonomy.TermCollection

**TypeId:** {AC6D0DA8-9769-4425-A395-6413A9A0367E}

**ShortName:** SP.Taxonomy.TermCollection

Represents collection of Term (section 3.1.5.16) objects.

#### 3.1.5.17.1 Properties

#### 3.1.5.17.1.1 Scalar Properties

None.

#### 3.1.5.17.1.2 ObjectPath Properties
3.1.5.17.2 Methods

3.1.5.17.2.1 Scalar Methods

None.

3.1.5.17.2.2 ObjectPath Methods

3.1.5.17.2.2.1 GetByIndex

Return Type: Microsoft.SharePoint.Client.Taxonomy.Term

Returns the Term (section 3.1.5.16) specified by its GUID index.

Parameters:

index: The GUID index of the Term to be returned.

Type: CSOM GUID

3.1.5.17.2.2.2 GetByIndex

Return Type: Microsoft.SharePoint.Client.Taxonomy.Term

Returns the Term (section 3.1.5.16) specified by its index name.

Parameters:

index: The index name of the Term to be returned.

Type: CSOM String

3.1.5.18 Microsoft.SharePoint.Client.Taxonomy.TermGroup

TypeId: {65D76872-0B65-42DE-8EBD-D76F6D3491C6}

ShortName: SP.Taxonomy.TermGroup

Represents the top-level container in a TermStore (section 3.1.5.23) object.

Remarks:

The TermGroup object derives from the TaxonomyItem (section 3.1.5.14) base object.

The ContributorPrincipalNames and GroupManagerPrincipalNames properties are not included in the default scalar property set for this type.

3.1.5.18.1 Properties

3.1.5.18.1.1 Scalar Properties

3.1.5.18.1.1.1 CreatedDate

Type: CSOM DateTime

Accessibility: Read Only
Specifies the date and time when this item was created.

3.1.5.18.1.1.2  Description

Type: CSOM String
Accessibility: Read/Write

Specifies the description of this TermGroup (section 3.1.5.18) object.

3.1.5.18.1.1.3  Id

Type: CSOM GUID
Accessibility: Read Only

Specifies the identifier of the current TaxonomyItem (section 3.1.5.14) object.

3.1.5.18.1.1.4  IsSiteCollectionGroup

Type: CSOM Boolean
Accessibility: Read Only

Specifies information about whether this TermGroup (section 3.1.5.18) object is used only by a specific site collection (section 3.1.5.19).

3.1.5.18.1.1.5  IsSystemGroup

Type: CSOM Boolean
Accessibility: Read Only

Specifies information about whether this TermGroup (section 3.1.5.18) object is the system TermGroup. The system TermGroup is the TermGroup where TermSet (section 3.1.5.20) objects are used by the system itself.

3.1.5.18.1.1.6  LastModifiedDate

Type: CSOM DateTime
Accessibility: Read Only

Specifies the date and time when this item was most recently updated.

3.1.5.18.1.1.7  Name

Type: CSOM String
Accessibility: Read/Write

Specifies the name of this TermGroup (section 3.1.5.18) object.

3.1.5.18.1.1.8  ContributorPrincipalNames

Type: CSOM array of CSOM String
Accessibility: Read Only

Returns the principal names indicating the users who are contributors for this group.
3.1.5.18.1.1.9 GroupManagerPrincipalNames

Type: CSOM array of CSOM String

Accessibility: Read Only

Returns the principal names indicating the users who are group managers for this group.

3.1.5.18.1.2 ObjectPath Properties

3.1.5.18.1.2.1 TermSets

Type: Microsoft.SharePoint.Client.Taxonomy.TermSetCollection

Accessibility: Read Only

Specifies a collection of the child TermSet (section 3.1.5.20) instances of this TermGroup (section 3.1.5.18).

3.1.5.18.1.2.2 TermStore

Type: Microsoft.SharePoint.Client.Taxonomy.TermStore

Accessibility: Read Only

Specifies the TermStore (section 3.1.5.23) object from which the current TaxonomyItem (section 3.1.5.14) was retrieved.

3.1.5.18.2 Methods

3.1.5.18.2.1 Scalar Methods

3.1.5.18.2.1.1 DeleteObject

Return Type: None

Deletes this TermGroup (section 3.1.18) from the TermStore (section 3.1.5.23) object.

Parameters: None

3.1.5.18.2.1.2 ExportObject

Return Type: CSOM String

This method is not implemented.

Parameters: None

3.1.5.18.2.1.3 NormalizeName

This method is a static CSOM method.

Return Type: CSOM String

Normalizes a name of a TaxonomyItem (section 3.1.5.14) object.

Parameters:

name: The name to normalize.
3.1.5.18.2.1.4  AddContributor

Return Type: None

Adds a contributor to the Group object.

Parameters:

principalName: The login name of the user to be added as a contributor.

Type: CSOM String

3.1.5.18.2.1.5  AddGroupManager

Return Type: None

Adds a group manager to the Group object.

Parameters:

principalName: The login name of the user to be added as a group manager.

Type: CSOM String

3.1.5.18.2.2  ObjectPath Methods

3.1.5.18.2.2.1  CreateTermSet

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermSet

Creates a new TermSet (section 3.1.5.20) object in the current TermGroup (section 3.1.5.18) object by using the provided language and the specified unique identifier.

Parameters:

name: The name of the new TermSet object that is being created.

Type: CSOM String

newTermSetId: The unique identifier of the new TermSet object that is being created.

Type: CSOM GUID

lcid: The language code identifier (LCID) for the language that the new TermSet name is written in.

Type: CSOM Int32

3.1.5.18.2.2.2  GetChanges

Return Type: Microsoft.SharePoint.Client.Taxonomy.ChangedItemCollection

Specifies a collection of ChangedItem (section 3.1.5.2) objects that represent changes to this TermGroup (section 3.1.5.18) since a specified time.

Parameters:

changeInformation: A Coordinated Universal Time (UTC) indicating the earliest change to be returned, as specified by ChangeInformation (section 3.1.5.26).
Type: Microsoft.SharePoint.Client.Taxonomy.ChangeInformation

3.1.5.18.2.2.3 GetTermSetsWithCustomProperty

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermSetCollection

Gets a collection of all of the TermSet (section 3.1.5.20) objects in this Group (section 3.1.5.18) that the current user has permissions to read, that have a custom property matching the criteria provided.

Remarks:

The CustomPropertyMatchInformation (section 3.1.5.29) values that are set MUST be one of these combinations:

- CustomPropertyMatchInformation.CustomPropertyName.
- CustomPropertyMatchInformation.CustomPropertyName and CustomPropertyMatchInformation.CustomPropertyValue.
- CustomPropertyMatchInformation.CustomPropertyName, CustomPropertyMatchInformation.CustomPropertyValue, and CustomPropertyMatchInformation.StringMatchOption.
- CustomPropertyMatchInformation.CustomPropertyName, CustomPropertyMatchInformation.CustomPropertyValue, CustomPropertyMatchInformation.StringMatchOption, and CustomPropertyMatchInformation.ResultCollectionSize.

If the combination of CustomPropertyMatchInformation values is invalid, an ArgumentException or ArgumentNullException will be thrown indicating the invalid setting.

The CustomPropertyMatchInformation.CustomPropertyName (section 3.1.5.29.1.1.1) MUST NOT be set to null or empty, and MUST NOT exceed 255 characters in length. It also MUST NOT contain any of the following illegal characters: ; " < > | & tab

The CustomPropertyMatchInformation.CustomPropertyValue (section 3.1.5.29.1.1.2) MUST NOT be set to null or empty, and MUST NOT exceed 255 characters in length. It also MUST NOT contain the illegal characters \t or \n.

The search is case-insensitive, and will match either the entire value or the custom property name or value, or only check that it starts with the specified string depending on the value of the CustomPropertyMatchInformation.StringMatchOption (section 3.1.5.29.1.1.4) parameter. If the CustomPropertyMatchInformation.StringMatchOption is not set, it defaults to exact match.

The CustomPropertyMatchInformation.ResultCollectionSize (section 3.1.5.29.1.1.3) value MUST be greater than 0. The maximum number of results returned from this search is restricted by the CustomPropertyMatchInformation.ResultCollectionSize parameter. If CustomPropertyMatchInformation.ResultCollectionSize is not set, then a maximum of 100 results is returned.

To get the most accurate results, it is recommended that any pending changes be committed to the database using CommitAll (section 3.1.5.23.2.1.1) prior to executing this search.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>Incorrect combination of CustomPropertyMatchInformation settings provided.</td>
</tr>
<tr>
<td>Error Code</td>
<td>Error Type Name</td>
<td>Condition</td>
</tr>
<tr>
<td>------------------</td>
<td>------------------------</td>
<td>---------------------------------------------------------------------------</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of CustomPropertyMatchInformation.CustomPropertyName is invalid. It either contains invalid characters or is too long.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of CustomPropertyMatchInformation.CustomPropertyValue is invalid. It exceeds the maximum allowed length of 255.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of CustomPropertyMatchInformation.CustomPropertyValue is invalid. It contains invalid character 't', 'n', or both.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>CustomPropertyMatchInformation.TrimUnavailable MUST NOT be set.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>Some CustomPropertyMatchInformation settings MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The CustomPropertyMatchInformation.CustomPropertyName MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The CustomPropertyMatchInformation.CustomPropertyValue MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentException</td>
<td>The CustomPropertyMatchInformation.ResultCollectionSize is invalid. It MUST be greater than 0.</td>
</tr>
</tbody>
</table>

**Parameters:**

`CustomPropertyMatchInformation`: Indicates the search criteria to use when retrieving TermSet objects.

Type: Microsoft.SharePoint.Client.Taxonomy.CustomPropertyMatchInformation

**3.1.5.19 Microsoft.SharePoint.Client.Taxonomy.TermGroupCollection**

**TypeId**: `{FEF2ACDD-8FB1-40B9-A8B0-CCD9512A9105}`

**ShortName**: SP.Taxonomy.TermGroupCollection

A collection of TermGroup (section 3.1.15) objects.

**3.1.5.19.1 Properties**

**3.1.5.19.1.1 Scalar Properties**

None.

**3.1.5.19.1.2 ObjectPath Properties**

None.
3.1.5.19.2 Methods

3.1.5.19.2.1 Scalar Methods

None.

3.1.5.19.2.2 ObjectPath Methods

3.1.5.19.2.2.1 GetByIndex

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermGroup

Returns the **TermGroup** (section 3.1.5.18) specified by its **GUID** index.

**Parameters:**

*index:* The GUID index of the **TermGroup**.

Type: **CSOM GUID**

3.1.5.19.2.2.2 GetByName

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermGroup

Returns the **TermGroup** (section 3.1.5.18) specified by its index name.

**Parameters:**

*index:* The name of the **TermGroup** index.

Type: **CSOM String**

3.1.5.20 Microsoft.SharePoint.Client.Taxonomy.TermSet

**TypeId:** {E26FEB13-2940-4DB9-A52B-12B160113A80}

**ShortName:** SP.Taxonomy.TermSet

Represents a hierarchical or flat set of **Term** (section 3.1.5.16) objects known as a **term set**.

The **TermSet** object contains one or more child **Term** objects, and the root level of the hierarchy can be accessed through the **Terms** (section 3.1.5.20.1.2.2) property. Get child **Term** objects by using search methods, such as **GetTerms** (section 3.1.5.20.2.2.6). Permissions for editing the **TermSet** object and its child **Term** objects are defined on the parent **TermGroup** (section 3.1.5.18) object of the current **TermSet**.

The **TermSet** object also provides properties to get and set values of the **TermSet**, such as name and description, and a way to delete **TermSet** objects. Additionally, the **TermSet** object provides methods for checking changes on the **TermSet**.

The **TermSet** object derives from the **TermSetItem** (section 3.1.5.22) base object, which provides additional behavior such as the ability to create child **Term** objects.

3.1.5.20.1 Properties

3.1.5.20.1.1 Scalar Properties

3.1.5.20.1.1.1 Contact
Type: **CSOM String**

**Accessibility:** Read/Write

Specifies the contact for the current TermSet (section 3.1.5.20) object.

**Remarks:**

It is recommended that the Contact value be an email address for a user who can be contacted for term suggestions and feedback. Contact information cannot be set on the orphaned terms term set. Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set. The operation failed.</td>
</tr>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
</tbody>
</table>

### 3.1.5.20.1.1.2 CreatedDate

**Type:** **CSOM DateTime**

**Accessibility:** Read Only

Specifies the date and time when this item was created.

### 3.1.5.20.1.1.3 CustomProperties

**Type:** **CSOM dictionary**

**Accessibility:** Read Only

Specifies the custom properties for this TermSet (section 3.1.5.20).

### 3.1.5.20.1.1.4 CustomSortOrder

**Type:** **CSOM String**

**Accessibility:** Read/Write

Specifies the custom sort order of the current Term (section 3.1.5.16) object.

The property value is a string composed of the child Term GUIDs that are separated by a colon. When the custom sort order is null or empty, the child Term objects of the current Term is sorted alphabetically based on the working language of the current TermStore (section 3.1.5.23) object.

**Remarks:**

If the custom sort order value contains invalid GUID or one of the GUIDs does not belong to its child Term objects, then it is ignored. If any child Term GUID is not included, then they will be sorted alphabetically following the Term objects with custom sort order.

Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

**Exceptions:**
### 3.1.5.20.1.1.5 Description

**Type:** CSOM String

**Accessibility:** Read/Write

Specifies the description for the current TermSet (section 3.1.5.20) object.

**Remarks:**

The description can be null or empty. The description cannot exceed 1000 characters, and it cannot contain any tab characters. Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The input description value is too long or contains characters that are not valid.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set (section 3.1.5.20). The operation failed.</td>
</tr>
</tbody>
</table>

### 3.1.5.20.1.1.6 Id

**Type:** CSOM GUID

**Accessibility:** Read Only

Specifies the GUID identifier of the current TaxonomyItem (section 3.1.14).

### 3.1.5.20.1.1.7 IsAvailableForTagging

**Type:** CSOM Boolean

**Accessibility:** Read/Write

Specifies whether the TermSet (section 3.1.5.20) object appears in the tagging user interface.

**Remarks:**

The property cannot be set in the orphaned terms term set. Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.
Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set (section 3.1.5.20). The operation failed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Modifying tagging related properties of the orphaned terms term set is not allowed.</td>
</tr>
</tbody>
</table>

3.1.5.20.1.1.8 IsOpenForTermCreation

Type: CSOM Boolean

Accessibility: Read/Write

Indicates whether the TermSet can be updated by all users, or alternatively just by the specified contributors for the TermGroup (section 3.1.5.18) object.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set (section 3.1.5.20). The operation failed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Modifying tagging related properties of the orphaned terms term set is not allowed.</td>
</tr>
</tbody>
</table>

Remarks:

A true value indicates that all users can add terms to the current TermSet. A value of false indicates that only contributors assigned to the TermGroup can add terms to the current TermSet. Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

3.1.5.20.1.1.9 LastModifiedDate

Type: CSOM DateTime

Accessibility: Read Only

Specifies the date and time when this item was most recently updated.

3.1.5.20.1.1.10 Name

Type: CSOM String

Accessibility: Read/Write

Specifies the name for the current TermSet (section 3.1.5.20) object.

Remarks:
Retrieving this property gets the name of the `TermSet` in the current `WorkingLanguage` (section 3.1.5.23.1.1.6) property. If there is no name available for the `WorkingLanguage`, then the name for the `DefaultLanguage` is returned. Setting this property sets the name for the current `WorkingLanguage`. The name cannot exceed 255 characters, and it cannot contain the following characters: ; " < > | & or the tab character. Renaming is not permitted on any system term set objects, such as the `keywords` term set or the `orphaned terms` term set. Call the `CommitAll` (section 3.1.5.23.2.1.1) method to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value is invalid. It either contains invalid characters or is too long.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The renaming of a system term set is not allowed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set. The operation failed.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The name cannot be null or empty.</td>
</tr>
</tbody>
</table>

3.1.5.20.1.1.11  Owner

**Type:** CSOM String

**Accessibility:** Read/Write

Specifies the user who is responsible for the current `TermSet` (section 3.1.5.20).

Returns the login name of the user who is the owner of the `TermSet`.

**Remarks:**

The `Owner` value can be an empty string. It cannot exceed 255 characters. `Owner` cannot be set on the `orphaned terms term set`. Call the `CommitAll` (section 3.1.5.23.2.1.1) method to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The user name is too long.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The user name cannot be found.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The value is null.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set. The operation failed.</td>
</tr>
</tbody>
</table>

[MS-EMMCSOM] - v20181211
Enterprise Managed Metadata Client-Side Object Model Protocol
Copyright © 2018 Microsoft Corporation
Release: December 11, 2018
3.1.5.20.1.1.12 Stakeholders

Type: CSOM array of CSOM String

Accessibility: Read Only

Gets the collection of stakeholders for the current TermSet (section 3.1.5.20) object.

A stakeholder is a user who has an interest in the current TermSet. A TermSet can have 0 or more stakeholders. The returned collection can be empty. The returned collection is read-only. To add a stakeholder, use the AddStakeholder (section 3.1.5.20.2.1.8) method. To delete a stakeholder, use the DeleteStakeholder (section 3.1.5.20.2.1.9) method.

3.1.5.20.1.1.13 Names

Type: CSOM dictionary

Accessibility: Read Only

Gets the name for this TermSet.

3.1.5.20.1.2 ObjectPath Properties

3.1.5.20.1.2.1 Group

Type: Microsoft.SharePoint.Client.Taxonomy.TermGroup

Accessibility: Read Only

Specifies the parent TermGroup (section 3.1.5.18) object of the current TermSet (section 3.1.5.20) object.

3.1.5.20.1.2.2 Terms

Type: Microsoft.SharePoint.Client.Taxonomy.TermCollection

Accessibility: Read Only

Specifies a collection of child Term (section 3.1.5.16) objects at the root of the TermSet (section 3.1.5.20) object.

Remarks:

The return TermCollection (section 3.1.5.17) is sorted based on the custom sort order. If there is no custom sort order defined, then it is sorted by label based on the alphabetical order of the WorkingLanguage (section 3.1.5.23.1.1.6) property.

3.1.5.20.1.2.3 TermStore

Type: Microsoft.SharePoint.Client.Taxonomy.TermStore

Accessibility: Read Only

Specifies the TermStore (section 3.1.5.23) object from which the current TaxonomyItem (section 3.1.5.14) was retrieved.

3.1.5.20.2 Methods

3.1.5.20.2.1 Scalar Methods
3.1.5.20.2.1.1 DeleteCustomProperty

Return Type: None

Deletes a custom property for this TermSet (section 3.1.5.20).

Parameters:

name: The name of the custom property to be deleted.

Type: CSOM String

3.1.5.20.2.1.2 DeleteObject

Return Type: None

Deletes the current TermSet (section 3.1.5.20) from the current TermGroup (section 3.1.5.18) object.

Remarks:

This method deletes the current TermSet object and all of its child Term (section 3.1.5.16) objects. Any source Term members in the current TermSet object that are reused in other TermSet objects will be moved to the orphaned terms term set. The DeleteObject operation is not permitted on any system term set object, such as the keywords term set or the orphaned terms term set. Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set. The operation failed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Deleting a system term set is not allowed.</td>
</tr>
</tbody>
</table>

Parameters: None

3.1.5.20.2.1.3 ExportObject

Return Type: CSOM String

This method is not implemented.

Parameters: None

3.1.5.20.2.1.4 Move

Return Type: None

Moves the current TermSet (section 3.1.5.20) into a different TermGroup (section 3.1.5.18) object.

Parameters:

targetGroup: The TermGroup (section 3.1.5.18) into which the current TermSet is moved.
Type: Microsoft.SharePoint.Client.Taxonomy.TermGroup

**Remarks:**

Moves the current TermSet and all of its child Term (section 3.1.5.16) objects from the current TermGroup to the target TermGroup. Move is not permitted on any system term set objects, such as the keywords term set or the orphaned terms term set. Moving a TermSet to the System TermGroup is also not permitted. Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The targetGroup MUST NOT be null.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Moving a system term set is not allowed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Moving a TermSet to system TermGroup is not allowed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>TermSet names MUST be unique within a TermGroup.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is not valid in the orphaned terms term set. The operation failed.</td>
</tr>
</tbody>
</table>

### 3.1.5.20.2.1.5 SetCustomProperty

**Return Type:** None

Adds a new custom property if one with the same name does not already exist. Updates the value of the property with the new value if the property already exists.

**Parameters:**

- **name:** The name of the property to be updated.
  
  Type: CSOM String

- **value:** The value of the property to be added or updated.
  
  Type: CSOM String

### 3.1.5.20.2.1.6 DeleteAllCustomProperties

**Return Type:** None

Deletes all of the custom properties from this TermSet (section 3.1.5.20).

**Parameters:** None

### 3.1.5.20.2.1.7 NormalizeName

This method is a static CSOM method.
Return Type: **CSOM String**

Normalizes a name of a **TaxonomyItem** (section 3.1.5.14) object.

**Parameters:**

*name*: The name to normalize.

Type: CSOM String

### 3.1.5.20.2.1.8 AddStakeholder

**Return Type:** None

 Adds a stakeholder to the **TermSet** (section 3.1.5.20) object.

**Remarks:**

A stakeholder is a user who has an interest in the current **TermSet** object. A **TermSet** can have 0 or more stakeholders.

Call the **CommitAll** (section 3.1.5.23.2.1.1) method to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The <strong>stakeholderName</strong> is too long.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The <strong>stakeholderName</strong> cannot be found.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The <strong>stakeholderName</strong> is <strong>null</strong> or empty.</td>
</tr>
</tbody>
</table>

**Parameters:**

*stakeholderName*: The login name of the user to be added as a stakeholder.

Type: **CSOM String**

### 3.1.5.20.2.1.9 DeleteStakeholder

**Return Type:** None

Deletes a stakeholder from the **TermSet** (section 3.1.5.20) object.

**Remarks:**

Call the **CommitAll** (section 3.1.5.23.2.1.1) method to save this change to the database.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
</tbody>
</table>
Parameters:

stakeholderName: The login name of an existing stakeholder for the current TermSet object.

Type: CSOM String

3.1.5.20.2.2 ObjectPath Methods

3.1.5.20.2.2.1 Copy

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermSet

Creates a copy of the TermSet (section 3.1.5.20) object under the current parent TermGroup (section 3.1.5.18) object.

Remarks:

All Term (section 3.1.5.16) objects in the current TermSet become reused by the new TermSet. The copied TermSet will have a variation of the original TermSet name, such as having the text 'Copy of ' prefixed to the name. Copy is not permitted on any system term set or the orphaned terms term set. Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set object. The operation failed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Copying a system term set is not allowed.</td>
</tr>
</tbody>
</table>

Parameters: None

3.1.5.20.2.2.2 CreateTerm

Return Type: Microsoft.SharePoint.Client.Taxonomy.Term

Creates a new Term (section 3.1.5.16) object under this TaxonomyItem (section 3.1.5.14) object. The name is the default Label (section 3.1.5.10) for the specified language code identifier (LCID).

Parameters:

name: Default label for the LCID.

Type: CSOM String

lcid: LCID to create the label on.
Type: CSOM Int32

newTermId: GUID for the Term.

Type: CSOM GUID

Remarks:

If the specified LCID is not the default language of the TermStore (section 3.1.5.23), a Label with the name and the default language of the TermStore will also be created to ensure the Term has a default Label for the default language of the TermStore.

If the current TermSetItem is the keyword TermSet, then Label is created for the TermStore default language.

The name value will be normalized to trim consecutive spaces into one and replace the "&" character with the wide character version of the character (\uFF06). The leading and trailing spaces will be trimmed. It MUST NOT be empty, exceed 255 characters, or contain any of the following characters: ; " < > | & or the tab character.

The current user will be set as the owner of the new Term.

If the newTermId already exists in the TermStore, an exception will be thrown.

Call the CommitAll (section 3.1.5.23.1.1) method to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is not valid in the orphaned terms term set. The operation failed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Keyword TermSet (section 3.1.5.20) only allows a flat list of Term objects.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The name is not valid, it contains invalid characters, or is too long.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentException</td>
<td>The name is null or an empty string.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The lcid is not valid.</td>
</tr>
</tbody>
</table>

3.1.5.20.2.2.3 GetAllTerms

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermCollection

Returns a collection that contains a flat list of all Term (section 3.1.5.16) objects in the TermSet (section 3.1.5.20) object, as specified by TermCollection (section 3.1.5.17).

Remarks:

This method is used to do bulk operations on all Term objects in a TermSet without traversing the hierarchy. There is no specific order to the Term objects in the collection. The collection includes deprecated term Term objects. To walk through the Term hierarchy, use the Terms (section 3.1.5.20.1.2.2) property instead.
**Parameters:** None

### 3.1.5.20.2.2.4 GetChanges

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.ChangedItemCollection

Specifies a collection of `ChangedItem` (section 3.1.5.2) objects that represent changes to the current `TermSet` (section 3.1.5.20) since a specified time.

**Parameters:**

`changeInformation`: A **Coordinated Universal Time (UTC)** indicating the earliest change to be included in the result collection, as specified by `ChangeInformation` (section 3.1.5.26).

Type: Microsoft.SharePoint.Client.Taxonomy.ChangeInformation

### 3.1.5.20.2.2.5 GetTerm

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.Term

Specifies a `Term` (section 3.1.5.16) from the current `TermSet` (section 3.1.5.20) based on its `GUID` identifier.

**Parameters:**

`termId`: The identifier of the `Term` to get.

Type: **CSOM GUID**

**Remarks:**

A `null` is returned if the `Term` cannot be found or if the user does not have sufficient permissions to access it.

### 3.1.5.20.2.2.6 GetTerms

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermCollection

Gets a collection of all `Term` (section 3.1.5.16) objects from the current `TermSet` (section 3.1.5.20) object based on the label criteria provided.

**Remarks:**

The `LabelMatchInformation` values that are set MUST be one of these combinations:

- `LabelMatchInformation.TermLabel` and `LabelMatchInformation.TrimUnavailable`.
- `LabelMatchInformation.TermLabel`, `LabelMatchInformation.Lcid`, and `LabelMatchInformation.TrimUnavailable`.

---

[MS-EMMCSOM] - v20181211
Enterprise Managed Metadata Client-Side Object Model Protocol
Copyright © 2018 Microsoft Corporation
Release: December 11, 2018
If the combination of **LabelMatchInformation** (section 3.1.5.30) values is invalid, an **ArgumentException** or **ArgumentNullException** will be thrown indicating the invalid setting.

The **LabelMatchInformation.TermLabel** (section 3.1.5.30.1.1.6) cannot be null or empty, and MUST NOT exceed 255 characters in length. It also cannot contain any of the following illegal characters: ; " < > | & tab

The search is case-insensitive, and will match either the entire label or only check that it starts with the specified string depending on the value of the **LabelMatchInformation.StringMatchOption** (section 3.1.5.30.1.1.5) parameter. If the **LabelMatchInformation.StringMatchOption** is not set, it defaults to exact match.

If **LabelMatchInformation.TrimUnavailable** (section 3.1.5.30.1.1.8) is true, then terms that have **Term.IsAvailableForTagging** (section 3.1.5.16.1.1.5) set to false will be trimmed from the results; otherwise, all terms will be returned regardless of their **Term.IsAvailableForTagging** value.

If **LabelMatchInformation.Lcid** (section 3.1.5.30.1.1.3) is specified, then a **Term** will be returned if a **Label** in the specified **Lcid** matches the string provided. If no **Lcid** is specified, then all **Term.Labels** (section 3.1.5.16.1.2.1) in all working languages will be searched by this method. The **Lcid** value MUST be one of the working languages of the **term store**, specified by **TermStore.Languages** (section 3.1.5.23.1.1.4).

If **LabelMatchInformation.DefaultLabelOnly** (section 3.1.5.30.1.1.1) is true, then only the default labels will be searched.

The maximum number of results returned from this search is restricted by the **LabelMatchInformation.ResultCollectionSize** (section 3.1.5.30.1.1.4) parameter. If **LabelMatchInformation.ResultCollectionSize** is not set, then a maximum of 100 results is returned.

To get the most accurate results, it is recommended that any pending changes be committed to the database using **CommitAll** (section 3.1.5.23.2.1.1) prior to executing this search.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>Incorrect combination of <strong>LabelMatchInformation</strong> settings provided.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>Invalid value provided for one of the <strong>LabelMatchInformation</strong> settings.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of <strong>LabelMatchInformation.TermLabel</strong> is invalid. It either contains invalid characters or is too long.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>Some <strong>LabelMatchInformation</strong> settings cannot be null or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The <strong>LabelMatchInformation.TermLabel</strong> cannot be null or empty.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The <strong>LabelMatchInformation.Lcid</strong> value MUST be one of the working languages in the current <strong>TermStore</strong>.</td>
</tr>
</tbody>
</table>

**Parameters:**
**labelMatchInformation**: Indicates the search criteria to use when retrieving **Term** objects.

Type: Microsoft.SharePoint.Client.Taxonomy.LabelMatchInformation

### 3.1.5.20.2.2.7 GetTermsWithCustomProperty

**Return Type**: Microsoft.SharePoint.Client.Taxonomy.TermCollection

Gets a collection of all **Term** (section 3.1.5.16) objects from the current **TermSet** (section 3.1.5.20) object that have a custom property matching the criteria provided.

**Remarks**:

The **CustomPropertyMatchInformation** (section 3.1.5.29) values that are set MUST be one of these combinations:

- **CustomPropertyMatchInformation.CustomPropertyName** and **CustomPropertyMatchInformation.TrimUnavailable**.
- **CustomPropertyMatchInformation.CustomPropertyName**, **CustomPropertyMatchInformation.ResultCollectionSize**, and **CustomPropertyMatchInformation.TrimUnavailable**.
- **CustomPropertyMatchInformation.CustomPropertyName**, **CustomPropertyMatchInformation.CustomPropertyValue**, and **CustomPropertyMatchInformation.TrimUnavailable**.
- **CustomPropertyMatchInformation.CustomPropertyName**, **CustomPropertyMatchInformation.CustomPropertyValue**, **CustomPropertyMatchInformation.StringMatchOption**, and **CustomPropertyMatchInformation.TrimUnavailable**.
- **CustomPropertyMatchInformation.CustomPropertyName**, **CustomPropertyMatchInformation.CustomPropertyValue**, **CustomPropertyMatchInformation.StringMatchOption**, **CustomPropertyMatchInformation.ResultCollectionSize**, and **CustomPropertyMatchInformation.TrimUnavailable**.

If the combination of **CustomPropertyMatchInformation** values is invalid, an **ArgumentException** or **ArgumentNullException** will be thrown indicating the invalid setting.

The **CustomPropertyMatchInformation.CustomPropertyName** (section 3.1.5.29.1.1.1) MUST NOT be set to null or empty, and MUST NOT exceed 255 characters in length. It also MUST NOT contain any of the following illegal characters: ; " < > | & tab

The **CustomPropertyMatchInformation.CustomPropertyValue** (section 3.1.5.29.1.1.2) MUST NOT be set to null or empty, and MUST NOT exceed 255 characters in length. It also MUST NOT contain the illegal characters \t or \n.

The search is case-insensitive, and will match either the entire value or the custom property name or value, or only check that it starts with the specified string depending on the value of the **CustomPropertyMatchInformation.StringMatchOption** (section 3.1.5.29.1.1.4) parameter. If the **CustomPropertyMatchInformation.StringMatchOption** is not set, it defaults to exact match.

The **CustomPropertyMatchInformation.ResultCollectionSize** (section 3.1.5.29.1.1.3) value MUST be greater than 0. The maximum number of results returned from this search is restricted by the **CustomPropertyMatchInformation.ResultCollectionSize** parameter. If **CustomPropertyMatchInformation.ResultCollectionSize** is not set, then a maximum of 100 results is returned.
If `CustomPropertyMatchInformation.TrimUnavailable` (section 3.1.5.29.1.1.5) is set to true, then `Term` objects that have `Term.IsAvailableForTagging` (section 3.1.5.16.1.1.5) set to false will be trimmed from the results; otherwise, all matching terms will be returned regardless of their `Term.IsAvailableForTagging` value.

To get the most accurate results, it is recommended that any pending changes be committed to the database using `CommitAll` (section 3.1.5.23.2.1.1) prior to executing this search.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>Incorrect combination of <code>CustomPropertyMatchInformation</code> settings provided.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of <code>CustomPropertyMatchInformation.CustomPropertyName</code> is invalid. It either contains invalid characters or is too long.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of <code>CustomPropertyMatchInformation.CustomPropertyValue</code> is invalid. It exceeds the maximum allowed length of 255.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of <code>CustomPropertyMatchInformation.CustomPropertyValue</code> is invalid. It contains invalid character 't', 'n', or both.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>Some <code>CustomPropertyMatchInformation</code> settings MUST NOT be <code>null</code> or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The <code>CustomPropertyMatchInformation.CustomPropertyName</code> MUST NOT be <code>null</code> or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The <code>CustomPropertyMatchInformation.CustomPropertyValue</code> MUST NOT be <code>null</code> or empty.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The <code>CustomPropertyMatchInformation.ResultCollectionSize</code> is invalid. It MUST be greater than 0.</td>
</tr>
</tbody>
</table>

**Parameters:**

`customPropertyMatchInformation`: Indicates the search criteria to use when retrieving `Term` objects.

Type: `Microsoft.SharePoint.Client.Taxonomy.CustomPropertyMatchInformation`

### 3.1.5.20.2.2.8 ReuseTerm

**Return Type:** `Microsoft.SharePoint.Client.Taxonomy.Term`

Creates a new reused `Term` (section 3.1.5.16) object under this `TaxonomyItem` (section 3.1.5.14) object based on a `source term`.

**Parameters:**

[MS-EMMCSOM] - v20181211

Enterprise Managed Metadata Client-Side Object Model Protocol
Copyright © 2018 Microsoft Corporation
Release: December 11, 2018
sourceTerm: The source Term for the reused Term.
Type: Microsoft.SharePoint.Client.Taxonomy.Term

reuseBranch: Reuses the TermSetItem (section 3.1.5.22) and its child Term objects.
Type: CSOM Boolean

Remarks:
Reuses a Term (section 3.1.5.16) from a different TermSet to become a child of the current TermSetItem. The target TermSet cannot be the keyword or orphaned term TermSet. Keywords cannot be reused.

Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned term TermSet (section 3.1.5.20). The operation failed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Keywords are not allowed to be reused.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Reusing a deprecated term Term is not allowed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Term objects cannot be shared multiple times in the same TermSet.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The target term set cannot be the keywords term set or the orphaned terms term set.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The source Term is null or not valid.</td>
</tr>
</tbody>
</table>

3.1.5.20.2.2.9 ReuseTermWithPinning

Return Type: Microsoft.SharePoint.Client.Taxonomy.Term

Pins the given Term (section 3.1.5.16) to the given target TermSet (section 3.1.5.20).

Parameters:

sourceTerm: The source term.
Type: Microsoft.SharePoint.Client.Taxonomy.Term

3.1.5.20.2.2.10 GetAllTermsIncludeDeprecated

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermCollection

Returns a TermCollection (section 3.1.5.17) that contains a flat list of all Term (section 3.1.5.16) objects, including deprecated ones in the TermSet (section 3.1.5.20).

Parameters: None
3.1.5.21    Microsoft.SharePoint.Client.Taxonomy.TermSetCollection

TypeId: {659DF998-3955-45FA-B64E-3D6CA277A97A}

ShortName: SP.Taxonomy.TermSetCollection

Represents a collection of TermSet (section 3.1.5.20) objects.

3.1.5.21.1  Properties

3.1.5.21.1.1 Scalar Properties

None.

3.1.5.21.1.2 ObjectPath Properties

None.

3.1.5.21.2  Methods

3.1.5.21.2.1 Scalar Methods

None.

3.1.5.21.2.2 ObjectPath Methods

3.1.5.21.2.2.1 GetById

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermSet

Returns a TermSet (section 3.1.5.20) specified by its GUID index.

Parameters:

index: The index of the TermSet to be returned.

Type: CSOM GUID

3.1.5.21.2.2.2 GetByName

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermSet

Returns the TermSet (section 3.1.5.20) specified by its index name.

Parameters:

index: The index name of the TermSet.

Type: CSOM String

3.1.5.22    Microsoft.SharePoint.Client.Taxonomy.TermSetItem

TypeId: {A99E4A8F-010B-4E56-9B29-B7BD6EC51263}

ShortName: SP.Taxonomy.TermSetItem

Provides an abstraction of the TaxonomyItem (section 3.1.5.14) object that is a parent of Term (section 3.1.5.16) objects.
**TermSetItem** is an abstract class and MUST be overridden. It contains references to the parts that define a **TaxonomyItem**; for example, **Name** (section 3.1.5.22.1.1.7), **Id** (section 3.1.5.22.1.1.4), and so on. **TermSetItem** also contains a collection of child **Term** objects and provides an interface to retrieve or update the child **Term** objects. If the **TermSetItem** is an **orphaned term** or **keyword**, it MUST NOT have any child objects.

Each **TermSetItem** has a user associated with it as the owner. It contains a flag that indicates whether it can be used in the tagging user interface. It also has a **CustomSortOrder** (section 3.1.5.22.1.1.3) property that defines the sort order of the child **Terms**.

### 3.1.5.22.1 Properties

#### 3.1.5.22.1.1 Scalar Properties

#### 3.1.5.22.1.1.1 CreatedDate

**Type:** CSOM DateTime  
**Accessibility:** Read Only

Specifies the date and time when this item was created.

#### 3.1.5.22.1.1.2 CustomProperties

**Type:** CSOM dictionary  
**Accessibility:** Read Only

Specifies the collection of effective custom property instances for this **Term** (section 3.1.5.16) or **TermSet** (section 3.1.5.20).

#### 3.1.5.22.1.1.3 CustomSortOrder

**Type:** CSOM String  
**Accessibility:** Read/Write

Specifies the custom sort order for the child items of the current item.

#### 3.1.5.22.1.1.4 Id

**Type:** CSOM GUID  
**Accessibility:** Read Only

Specifies the **GUID** identifier of the current **TaxonomyItem** (section 3.1.5.14) object.

#### 3.1.5.22.1.1.5 IsAvailableForTagging

**Type:** CSOM Boolean  
**Accessibility:** Read/Write

Specifies the information about whether the **TaxonomyItem** (section 3.1.5.14) appears in the tagging user interface.

#### 3.1.5.22.1.1.6 LastModifiedDate

**Type:** CSOM DateTime
**Accessibility:** Read Only

Specifies the date and time when this item was most recently updated.

### 3.1.5.22.1.1.7 Name

**Type:** CSOM String

**Accessibility:** Read/Write

Specifies the name of the current TaxonomyItem (section 3.1.5.14) object.

### 3.1.5.22.1.1.8 Owner

**Type:** CSOM String

**Accessibility:** Read/Write

Specifies the name of the protocol client user who is responsible for this TaxonomyItem (section 3.1.5.14) object.

### 3.1.5.22.1.2 ObjectPath Properties

#### 3.1.5.22.1.2.1 Terms

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermCollection

**Accessibility:** Read Only

Gets a collection of the child Term (section 3.1.5.16) objects of this TaxonomyItem (section 3.1.5.14).

Returns a TermCollection (section 3.1.5.17).

#### 3.1.5.22.1.2.2 TermStore

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermStore

**Accessibility:** Read Only

Gets the TermStore (section 3.1.5.23) object from which the current TaxonomyItem (section 3.1.5.14) was retrieved.

### 3.1.5.22.2 Methods

#### 3.1.5.22.2.1 Scalar Methods

#### 3.1.5.22.2.1.1 DeleteCustomProperty

**Return Type:** None

Deletes a custom property.

**Parameters:**

- **name:** the name of the property to be deleted.

  **Type:** CSOM String
3.1.5.22.2.1.2 DeleteObject

Return Type: None

Deletes the current TaxonomyItem (section 3.1.5.22) object.

Parameters: None

3.1.5.22.2.1.3 SetCustomProperty

Return Type: None

Adds a new custom property or updates the value of an existing property.

Parameters:

name: The custom property name to set.

Type: CSOM String

value: The value to set for the specified custom property.

Type: CSOM String

3.1.5.22.2.1.4 DeleteAllCustomProperties

Return Type: None

Deletes all custom properties.

Parameters: None

3.1.5.22.2.1.5 NormalizeName

This method is a static CSOM method.

Return Type: CSOM String

Normalizes a name of a TaxonomyItem (section 3.1.5.14) object.

Parameters:

name: The name to normalize.

Type: CSOM String

3.1.5.22.2.2 ObjectPath Methods

3.1.5.22.2.2.1 CreateTerm

Return Type: Microsoft.SharePoint.Client.Taxonomy.Term

Creates a new Term (section 3.1.5.16) object under this TaxonomyItem (section 3.1.5.14) object.

The name is the default Label (section 3.1.5.10) for the specified language code identifier (LCID).

Parameters:

name: Default label for the Lcid.

Type: CSOM String
lcid: The LCID to create the label on.

Type: CSOM Int32

ewTermId: GUID for the Term.

Type: CSOM GUID

Remarks:

If the specified LCID is not the default language of the TermStore (section 3.1.5.23), a Label with the name and the default language of the TermStore will also be created to ensure the Term has a default Label for the default language of the TermStore.

If the current TermSetItem (section 3.1.5.22) is the keyword TermSet, then Label is created for the TermStore default language.

The name value will be normalized to trim consecutive spaces into one space and replace the "&" character with the wide character version of the character (\uFF06). The leading and trailing spaces will be trimmed. It MUST NOT be empty, exceed 255 characters, contain any of the following characters ;, ", <, >, |, &, or the tab character.

The current user will be set as the owner of the new Term.

If the newTermId already exists in the TermStore, an exception will be thrown.

CommitAll (section 3.1.5.23.2.1.1) method MUST be called to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set. The operation failed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Keyword term set only allows a flat list of Term objects.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The name is not valid, it contains characters that are not valid, or is too long.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The name is null or an empty string.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentException</td>
<td>The lcid is not valid.</td>
</tr>
</tbody>
</table>

3.1.5.22.2.2 ReuseTerm

Return Type: Microsoft.SharePoint.Client.Taxonomy.Term

Creates a new reused Term (section 3.1.5.16) object under this TaxonomyItem (section 3.1.5.14) object based on a source term.

Parameters:

sourceTerm: The source Term for the reused Term.

Type: Microsoft.SharePoint.Client.Taxonomy.Term
reuseBranch: Reuses the TermSetItem (section 3.1.5.22) and its child Term objects.

Type: CSOM Boolean

Remarks:
Reuses a Term from a different TermSet to become a child of the current TermSetItem. The target TermSet cannot be the keyword or orphaned term term set. Keywords cannot be reused.

CommitAll (section 3.1.5.23.2.1.1) method MUST be called to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The current user has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>This operation is invalid in the orphaned terms term set. The operation failed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Keywords are not allowed to be reused.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Reusing a deprecated Term is not allowed.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>Terms cannot be shared multiple times in the same term set.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The target TermSet cannot be the keywords term set or the orphaned terms term set.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The source Term is null or not valid.</td>
</tr>
</tbody>
</table>

3.1.5.22.2.2.3 GetTerms

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermCollection

Gets a paged collection containing the child Term objects of this TermSetItem object.

Returns a TermCollection (section 3.1.5.17).

Parameters:
pagingLimit: The maximum number of items to return in the collection.

Type: CSOM Int32

Remarks:
This method has the same results as Terms (section 3.1.5.22.1.2.1), except that it returns a collection that supports paging.

Call the CommitAll (section 3.1.5.23.2.1.1) method before using a paged collection. The paged collection does not reflect uncommitted changes if the collection size is greater than the pagingLimit.

The returned term collection is sorted based on the custom sort order. If there is no custom sort order defined, then it is sorted based on alphabetic order of the WorkingLanguage. section (3.1.5.23.1.1.6) property values.

Exceptions:
### Error Code

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The <code>pagingLimit</code> MUST be greater than zero.</td>
</tr>
</tbody>
</table>

### 3.1.5.22.2.2.4 ReuseTermWithPinning

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.Term

Creates a new Term (section 3.1.5.16) under this TaxonomyItem (section 3.1.5.14) based on a source term and makes this Term pinned so that only the local custom properties, IsAvailableForTagging (section 3.1.5.22.1.1.5) and custom sort order for this TermSet (section 3.1.5.20) can be changes in this TermSet.

**Parameters:**

- **sourceTerm:** The source term.

  Type: Microsoft.SharePoint.Client.Taxonomy.Term

### 3.1.5.23 Microsoft.SharePoint.Client.Taxonomy.TermStore

**TypeId:** `{9D8A8884-B1DC-4DBC-81C5-DDEA8AD3184C}`

**ShortName:** SP.Taxonomy.TermStore

Represents a term store that contains metadata within child TermGroup (section 3.1.5.18) objects, TermSet (section 3.1.5.20) objects, and Term (section 3.1.5.16) objects.

**Remarks:**

The TermStore contains zero or more TermGroup objects, which are used to organize Term objects within TermSets. Access the groups through the Groups (section 3.1.5.23.1.2.1) property.

The TermStore class also provides methods for creating new child TermGroup objects. Additionally, it provides properties that are used to get and set values of the TermStore object such as the set of working languages, and access to the keyword and orphaned term TermSet objects.

Use methods of the TermStore class such as the GetTerms (section 3.1.5.23.2.2.5) and GetTermSets (section 3.1.5.23.2.2.7) to search for and get Term and TermSet objects across all child groups.

Finally, the TermStore provides the CommitAll (section 3.1.5.23.2.1.1) method to commit changes to the protocol database. CommitAll is the only way to persist changes to the database when updates are made in child objects, such as TermGroup, TermSet, and Term.

The ContentTypePublishingHub property is not included in the default scalar property set for this type.

### 3.1.5.23.1 Properties

#### 3.1.5.23.1.1 Scalar Properties

#### 3.1.5.23.1.1.1 DefaultLanguage

**Type:** CSOM Int32

**Accessibility:** Read/Write
Specifies the language code identifier (LCID) of the default working language for the current TermStore (section 3.1.5.23) object.

Remarks:

The DefaultLanguage property is one of the LCIDs in the Languages (section 3.1.5.23.1.1.4) collection. Attempting to set a value that is not in the Languages collection automatically adds it to the Languages collection. Setting a new DefaultLanguage value is an expensive operation, which will create a new Label (section 3.1.5.10) object for each Term in the TermStore if one does not already exist for the LCID of the new DefaultLanguage. If a new Label MUST be created on a Term, the value will be a copy of the value of the Label on that Term (section 3.1.5.16) for the former DefaultLanguage. Generally the DefaultLanguage is used as the fallback language if a WorkingLanguage (section 3.1.5.23.1.1.6) has not been specified for the current TermStore instance. For temporary language changing it is preferable to setting the WorkingLanguage. This method automatically calls CommitAll (section 3.1.5.23.2.1.1), so any changes that are pending prior calling this method will automatically be persisted to the database.

3.1.5.23.1.1.2  
**Id**

Type: CSOM GUID

Accessibility: Read Only

Gets the unique identifier of the TermStore (section 3.1.5.23) object.

3.1.5.23.1.1.3  
**IsOnline**

Type: CSOM Boolean

Accessibility: Read Only

Indicates whether this TermStore (section 3.1.5.23) is currently online, or available.

Remarks:

This property is false if a TermStore is not available or instantiated.

3.1.5.23.1.1.4  
**Languages**

Type: CSOM array of CSOM Int32

Accessibility: Read Only

Gets an integer collection of language code identifiers (LCIDs) that represent the working languages for this TermStore (section 3.1.5.23).

3.1.5.23.1.1.5  
**Name**

Type: CSOM String

Accessibility: Read Only

Gets the name of the current TermStore (section 3.1.5.23) object.

3.1.5.23.1.1.6  
**WorkingLanguage**

Type: CSOM Int32

Accessibility: Read/Write
Specifies the current working language of the **TermStore** (section 3.1.5.23) object.

**Remarks:**

The **WorkingLanguage** is one of the LCIDs in the **Languages** (section 3.1.5.23.1.1.4) collection. Values that are not present in the **Languages** collection are skipped. If a **WorkingLanguage** is not explicitly set, it defaults to the LCID of the current user interface culture if it is one of the **Languages**, or alternately it defaults to the **DefaultLanguage** (section 3.1.5.23.1.1.1). Setting the **WorkingLanguage** only has affect during the lifetime of this current **TermStore** instance. The value is not persisted to the database. It is useful for purposes of switching languages temporarily in user interface code.

### 3.1.5.23.1.1.7  **ContentTypePublishingHub**

**Type:** CSOM String

**Accessibility:** Read Only

Gets the Uniform Resource Identifier (URI) of the site used as the content type syndication hub<1>, if the hub exists.

**Remarks:**

Returns null if the hub does not exist.

### 3.1.5.23.1.2  **ObjectPath Properties**

#### 3.1.5.23.1.2.1  **Groups**

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermGroupCollection

**Accessibility:** Read Only

Gets a collection of the child **TermGroup** (section 3.1.5.18) objects of this **TermStore** (section 3.1.5.23) object.

**Remarks:**

If the protocol client is a farm or tenant admin then all **TermGroup** objects will be returned, including site collection groups. Otherwise, all global groups will be returned, and only the site collection group for the current site will be returned if it exists. A farm admin is a farm administrator who can manage the configuration of the full farm. A tenant admin is a tenant administrator who can manage the configuration of farm administrator-delegated features and can manage all sites for their subscription from a single place, the tenant administration site.

#### 3.1.5.23.1.2.2  **KeywordsTermSet**

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermSet

**Accessibility:** Read Only

Gets the **TermSet** (section 3.1.5.20) object for this **TermStore** (section 3.1.5.23) object.

**Remarks:**

Contains the set of keyword **Term** (section 3.1.5.16) objects.

#### 3.1.5.23.1.2.3  **OrphanedTermsTermSet**

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermSet
**Accessibility:** Read Only

Gets the TermSet (section 3.1.5.20) for this TermStore (section 3.1.5.23) that contains the orphaned terms.

### 3.1.5.23.1.2.4 SystemGroup

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermGroup

**Accessibility:** Read Only

Gets the TermGroup (section 3.1.5.18) object for this TermStore (section 3.1.5.23) object, which contains the set of system term sets (keywords and orphaned terms).

### 3.1.5.23.1.2.5 HashTagsTermSet

**Type:** Microsoft.SharePoint.Client.Taxonomy.TermSet

**Accessibility:** Read Only

Gets the TermSet (section 3.1.5.20) object for this TermStore (section 3.1.5.23), which contains the set of #tags.

**Remarks:**

The #tags are used in microblogging and newsfeeds.

### 3.1.5.23.2 Methods

#### 3.1.5.23.2.1 Scalar Methods

##### 3.1.5.23.2.1.1 CommitAll

**Return Type:** None

Commits all updates to the database that have occurred since the last commit or rollback.

When objects are edited in the TermStore object, or in child objects such as TermGroup (section 3.1.5.18), TermSet (section 3.1.5.20), or Term (section 3.1.5.16), the changes are pending until the CommitAll method is called. After it is called, the updates that were made since the last CommitAll call are persistied to the term store's database. A successful call to CommitAll flushes the cache so that other consumers of the TermStore object will see the updates.

**CommitAll** is automatically called during each ExecuteQuery command, which is used to execute the current set of data retrieval queries and method invocations.

**Parameters:** None

##### 3.1.5.23.2.1.2 AddLanguage

**Return Type:** None

Adds a working language to the TermStore (section 3.1.5.23) object.

**Parameters:**

*lcid:* The language code identifier (LCID) of the working language to add.

**Type:** CSOM Int32
Remarks:

Adding additional working languages enables Label (section 3.1.5.10) objects to be added to Term (section 3.1.5.16) objects in multiple languages, enables the addition of multiple languages for TermSet (section 3.1.5.20) names, and so on. Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
</tbody>
</table>

3.1.5.23.2.1.3 DeleteLanguage

Return Type: None

Deletes a working language from the TermStore object.

Parameters:

lcid: The language code identifier (LCID) for the language to be deleted.

Type: CSOM Int32

Remarks:

The language being deleted cannot be the DefaultLanguage (section 3.1.5.23.1.1.1) or the last language in the Languages (section 3.1.5.23.1.1.4) collection. Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The last language cannot be deleted.</td>
</tr>
<tr>
<td>-1</td>
<td>TermStoreOperationException</td>
<td>The default language cannot be deleted.</td>
</tr>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
</tbody>
</table>

3.1.5.23.2.1.4 RollbackAll

Return Type: None

Discards all updates that have occurred since the last commit or rollback.

Remarks:

When objects are edited in the TermStore (section 3.1.5.23) object, or in child objects such as TermGroup (section 3.1.5.18), TermSet (section 3.1.5.20), or Term (section 3.1.5.16), the changes are pending until the CommitAll (section 3.1.5.23.2.1.1) method or RollbackAll (section 3.1.5.23.2.1.4) method is called. If TermStore.RollbackAll is called, the updates that were made since the last TermStore.CommitAll or TermStore.RollbackAll are discarded.
CommitAll is automatically called for client-side code during each ExecuteQuery command which is used to execute the current set of data retrieval queries and method invocations.

**Parameters:** None

### 3.1.5.23.2.1.5 UpdateCache

**Return Type:** None

This method forces the taxonomy cache to communicate with the Managed Metadata Service to check for changes to any taxonomy objects in this TermStore (section 3.1.5.23).

**Remarks:**

A background thread already performs these checks at periodic intervals, so calling UpdateCache is normally unnecessary. However, it is sometimes useful immediately before a change (to reduce the likelihood of save conflict errors), or immediately after a change (if that change has been performed via a different Web Front End (WFE)).

**Parameters:** None

### 3.1.5.23.2.1.6 UpdateUsedTermsOnSite

**Return Type:** None

This method makes sure that the current TermStore (section 3.1.5.23) object is aware of all child Term (section 3.1.5.16) objects that are used in the requested Site ([MS-CSOMSPT] section 3.2.5.119).

**Parameters:**

- **site:** The Site ([MS-CSOMSPT] section 3.2.5.119) to check against for Term usage.
  
  **Type:** Microsoft.SharePoint.Client.Site

### 3.1.5.23.2.1.7 GetPackage

**Return Type:** CSOM Stream

This method gets the PRIME package ([MS-PRIMEPF]) for the requested ID and type of package.

**Parameters:**

- **packageId:** The Id of the package. This value MUST be retrieved from the GetPackagesUpdateInformation method (section 3.1.5.23.2.1.8), from the value of the PId attribute.
  
  **Type:** CSOM String

- **packageType:** The type of the PRIME package. This value MUST be retrieved from the GetPackagesUpdateInformation method, from the value of the PType attribute.
  
  **Type:** CSOM GUID

- **farmVersion:** The version info of the farm. This parameter is for future use. You can pass in any values but those will be ignored.
  
  **Type:** CSOM String

- **siteCompatibilityLevel:** The site compatibility level. This parameter is for future use. You can pass in any values but those will be ignored.
Type: CSOM Int32

3.1.5.23.2.1.8 GetPackagesUpdateInformation

Return Type: CSOM Stream

This method gets the updated information of the package from a given time stamp. Returned information includes Package ID, type, and time stamp of the recent update in the XML document encoded with UTF8.

Here is the schema and example of the package update information:

example:

```xml
<?xml version="1.0" encoding="utf-8"?>
<Root>
  <P PId="0x00A7470EADF4194E2E9ED1031B61DA08840300312EEAE4648F0D54192F4E2D3OE8672CE" PType="b4ad3a44-d934-4c91-81f463aceade443" PP="true" />
  <P PId="0x00A7470EADF4194E2E9ED1031B61DA08840300312EEAE4648F0D54192F4E2D3OE8672CE" PType="b4ad3a44-d934-4c91-81f463aceade443" PP="true" />
  <T>2017-06-02T02:10:51.997Z</T>
</Root>
```

schema:

```xml
<xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
    xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Root">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="P" maxOccurs="unbounded" minOccurs="0">
          <xs:complexType>
            <xs:simpleContent>
              <xs:extension base="xs:string">
                <xs:attribute type="xs:string" name="PId" use="required" />
                <xs:attribute type="xs:string" name="PType" use="required" />
                <xs:attribute type="xs:string" name="PP" use="required" />
              </xs:extension>
            </xs:simpleContent>
          </xs:complexType>
        </xs:element>
        <xs:element type="xs:dateTime" name="T" />
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

P is a package entry. PId is the content type package id, PType is the package type GUID, for content type, it is "b4ad3a44-d934-4c91-81f463aceade443", PP means whether the package is published.

Parameters:

lastUpdateTime: Timestamp of the last known update.

Type: CSOM DateTime

3.1.5.23.2.1.9 UploadPackages

Return Type: None

This method uploads the content type of the PRIME package.
Parameters:

packageId: The Id of the package. This value MUST be retrieved from the GetPackagesUpdateInformation method, from the value of the PId attribute.

Type: CSOM String

stream: The compressed content type of the PRIME package.

Type: CSOM Stream

farmVersion: The version info of the farm. This parameter is for future use. You can pass in any values but those will be ignored.

Type: CSOM String

siteCompatibilityLevel: The site compatibility level. This parameter is for future use. You can pass in any values but those will be ignored.

Type: CSOM Int32

3.1.5.23.2.2 ObjectPath Methods

3.1.5.23.2.2.1 CreateGroup

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermGroup

Creates a new TermGroup (section 3.1.5.18) object in this TermStore (section 3.1.5.23).

Remarks:

The name cannot exceed 255 characters, and it cannot contain the following characters: ;, ", <, >, |, [, & ], or the tab-character.

Additionally, the name MUST be a unique TermGroup name within the current TermStore. Call the CommitAll (section 3.1.5.23.2.1.1) method to save this change to the database.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024891</td>
<td>UnauthorizedAccessException</td>
<td>The protocol client has insufficient permissions to perform this operation.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The name parameter is not valid. It either contains invalid characters, or is too long.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>TermGroup names MUST be unique.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentException</td>
<td>The name cannot be null or empty.</td>
</tr>
</tbody>
</table>

Parameters:

name: The name of the new TermGroup being created.

Type: CSOM String

groupId: The identifier of the new TermGroup being created.
Type: **CSOM GUID**

### 3.1.5.23.2.2.2 GetChanges

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.ChangedItemCollection

Gets a collection of **ChangedItem** (section 3.1.5.2) objects, as specified by **ChangedItemCollection** (section 3.1.5.3), that represent changes to this **TermStore** (section 3.1.5.23) since a specified time.

**Parameters:**

* changeInformation: A **Coordinated Universal Time (UTC)** time indicating the earliest change to be included in the result collection. Specified by **ChangeInformation** (section 3.1.5.26).

**Type:** Microsoft.SharePoint.Client.Taxonomy.ChangeInformation

**Remarks:**

The changes returned in the collection include changes in the **TermStore** and its child **TermGroup** (section 3.1.5.18), **TermSet** (section 3.1.5.20) objects, and **Term** (section 3.1.5.16) objects. The collection returned can be empty.

### 3.1.5.23.2.2.3 GetGroup

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermGroup

Gets a **TermGroup** (section 3.1.5.18) object based on the identifier of the **TermGroup**.

**Parameters:**

* id: The identifier of the **TermGroup** to get.

**Type:** **CSOM GUID**

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The id value cannot be a Guid.Empty value.</td>
</tr>
</tbody>
</table>

### 3.1.5.23.2.2.4 GetTerm

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.Term

Gets a **Term** (section 3.1.5.16) based on its identifier.

**Parameters:**

* termId: The ID of the Term object to be retrieved

**Type:** **CSOM GUID**

**Remarks:**

If the current **Term** is a member of more than one **TermSet** (section 3.1.5.20) objects, it will arbitrarily return the term from one of the **TermSet** objects. If the **Term** cannot be found or if the protocol client does not have sufficient permissions to access it, the method returns **null**. This method
searches for the Term object across all child groups and term sets that the protocol client has permissions to read.

Exceptions:

| Error Code   | Error Type Name       | Condition                                                      |
|--------------|-----------------------|                                                                |
| -2147467261  | ArgumentNullException | The termId value cannot be Guid.Empty.                          |

3.1.5.23.2.2.5 GetTerms

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermCollection

Gets a collection of Term (section 3.1.5.16) objects from all TermSet (section 3.1.5.20) objects in this TermStore (section 3.1.5.23) that the protocol client has permissions to read and based on the label criteria provided.

Remarks:

The LabelMatchInformation values that are set MUST be one of these combinations:

- LabelMatchInformation.TermLabel and LabelMatchInformation.TrimUnavailable.
- LabelMatchInformation.TermLabel, LabelMatchInformation.Lcid, and LabelMatchInformation.TrimUnavailable.

If the combination of LabelMatchInformation (section 3.1.5.30) values is invalid, an ArgumentException or ArgumentNullException will be thrown indicating the invalid setting.

The LabelMatchInformation.TermLabel (section 3.1.5.30.1.1.6) cannot be null or empty, and MUST NOT exceed 255 characters in length. It also cannot contain any of the following illegal characters: ; " < > | & and the tab character.

The search is case-insensitive, and will match either the entire label or only check that it starts with the specified string depending on the value of the LabelMatchInformation.StringMatchOption (section 3.1.5.30.1.1.5) parameter. If the LabelMatchInformation.StringMatchOption is not set, it defaults to exact match.

If LabelMatchInformation.TrimUnavailable (section 3.1.5.30.1.1.8) is true, then terms that have Term.IsAvailableForTagging (section 3.1.5.16.1.1.5) set to false will be trimmed from the results; otherwise, all terms will be returned regardless of their Term.IsAvailableForTagging value.

If LabelMatchInformation.Lcid (section 3.1.5.30.1.1.3) is specified, then a Term will be returned if a Label in the specified Lcid matches the string provided. If no Lcid is specified, then all Term.Labels (section 3.1.5.16.1.2.1) in all working languages will be searched by this method. The Lcid value MUST be one of the working languages of the term store, specified by TermStore.Languages (section 3.1.5.23.1.1.4).
If LabelMatchInformation.DefaultLabelOnly (section 3.1.5.30.1.1.1) is true, then only the default labels will be searched.

The maximum number of results returned from this search is restricted by the LabelMatchInformation.ResultCollectionSize (section 3.1.5.30.1.1.4) parameter. If LabelMatchInformation.ResultCollectionSize is not set, then a maximum of 100 results is returned.

If LabelMatchInformation.ExcludeKeyword (section 3.1.5.30.1.1.2) is true, then Term objects in the keywords term set will be excluded from the search; otherwise, all terms, including keywords, will be included in the search.

If LabelMatchInformation.TrimDeprecated (section 3.1.5.30.1.1.7) is true, then terms that have Term.IsDeprecated (section 3.1.5.16.1.1.6) set to true will be trimmed from the results; otherwise, all terms will be returned regardless of their Term.IsDeprecated value.

To get the most accurate results, it is recommended that any pending changes be committed to the database using CommitAll (section 3.1.5.23.2.1.1) prior to executing this search.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>Incorrect combination of LabelMatchInformation settings provided.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>Invalid value provided for one of the LabelMatchInformation settings.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of LabelMatchInformation.TermLabel is invalid. It either contains invalid characters or is too long.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>Some LabelMatchInformation settings cannot be null or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The LabelMatchInformation.TermLabel cannot be null or empty.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The LabelMatchInformation.Lcid value MUST be one of the working languages in the current TermStore.</td>
</tr>
</tbody>
</table>

Parameters:

labelMatchInformation: Indicates the search criteria to use when retrieving Term objects.

Type: Microsoft.SharePoint.Client.Taxonomy.LabelMatchInformation

3.1.5.23.2.2.6 GetTermSet

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermSet

This method retrieves a TermSet (section 3.1.5.20) object based on a TermSet identifier.

Parameters:

termSetId: The identifier of the TermSet to return.

Type: CSOM GUID
Remarks:

A null will be returned if the TermSet cannot be found or if the protocol client does not have sufficient permissions to access it. This method searches across all TermGroup (section 3.1.5.18) objects in the TermStore (section 3.1.5.23).

3.1.5.23.2.2.7 GetTermsWithCustomProperty

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermCollection

Gets a collection of all of the Term (section 3.1.5.16) objects, from all TermSet (section 3.1.5.20) objects in this TermStore (section 3.1.5.23) that the protocol client has permissions to read, that have a custom property matching the criteria provided.

Remarks:

The CustomPropertyMatchInformation (section 3.1.5.29) values that are set MUST be one of these combinations:

- CustomPropertyMatchInformation CustomPropertyName and CustomPropertyMatchInformation.TrimUnavailable.
- CustomPropertyMatchInformation CustomPropertyName, CustomPropertyMatchInformation ResultCollectionSize, and CustomPropertyMatchInformation TrimUnavailable.
- CustomPropertyMatchInformation CustomPropertyName, CustomPropertyMatchInformation CustomPropertyValue, and CustomPropertyMatchInformation TrimUnavailable.
- CustomPropertyMatchInformation CustomPropertyName, CustomPropertyMatchInformation CustomPropertyValue, CustomPropertyMatchInformation StringMatchOption, and CustomPropertyMatchInformation TrimUnavailable.
- CustomPropertyMatchInformation CustomPropertyName, CustomPropertyMatchInformation CustomPropertyValue, CustomPropertyMatchInformation StringMatchOption, CustomPropertyMatchInformation ResultCollectionSize, and CustomPropertyMatchInformation TrimUnavailable.

If the combination of CustomPropertyMatchInformation values is invalid, an ArgumentException or ArgumentNullException will be thrown indicating the invalid setting.

The CustomPropertyMatchInformation CustomPropertyName (section 3.1.5.29.1.1.1) MUST NOT be set to null or empty, and MUST NOT exceed 255 characters in length. It also MUST NOT contain any of the following illegal characters: ; " < > | & tab

The CustomPropertyMatchInformation CustomPropertyValue (section 3.1.5.29.1.1.2) MUST NOT be set to null or empty, and MUST NOT exceed 255 characters in length. It also MUST NOT contain the illegal characters \t or \n.

The search is case-insensitive, and will match either the entire value or the custom property name or value, or only check that it starts with the specified string depending on the value of the CustomPropertyMatchInformation StringMatchOption (section 3.1.5.29.1.1.4) parameter. If the CustomPropertyMatchInformation StringMatchOption is not set, it defaults to exact match.

The CustomPropertyMatchInformation ResultCollectionSize (section 3.1.5.29.1.1.3) value MUST be greater than 0. The maximum number of results returned from this search is restricted by the CustomPropertyMatchInformation ResultCollectionSize parameter. If
CustomPropertyMatchInformation.ResultCollectionSize is not set, then a maximum of 100 results is returned.

If CustomPropertyMatchInformation.TrimUnavailable (section 3.1.5.29.1.1.5) is set to true, then Term objects that have Term.IsAvailableForTagging (section 3.1.5.16.1.1.5) set to false will be trimmed from the results; otherwise, all matching terms will be returned regardless of their Term.IsAvailableForTagging value.

To get the most accurate results, it is recommended that any pending changes be committed to the database using CommitAll (section 3.1.5.23.1.1) prior to executing this search.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>Incorrect combination of CustomPropertyMatchInformation settings provided.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of CustomPropertyMatchInformation.CustomPropertyName is invalid. It either contains invalid characters or is too long.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of CustomPropertyMatchInformation.CustomPropertyValue is invalid. It exceeds the maximum allowed length of 255.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of CustomPropertyMatchInformation.CustomPropertyValue is invalid. It contains invalid character '\t', '\n', or both.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>Some CustomPropertyMatchInformation settings MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The CustomPropertyMatchInformation.CustomPropertyName MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The CustomPropertyMatchInformation.CustomPropertyValue MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The CustomPropertyMatchInformation.ResultCollectionSize is invalid. It MUST be greater than 0.</td>
</tr>
</tbody>
</table>

Parameters:

customPropertyMatchInformation: Indicates the search criteria to use when retrieving Term objects.

Type: Microsoft.SharePoint.Client.Taxonomy.CustomPropertyMatchInformation

3.1.5.23.2.2.8 GetSiteCollectionGroup

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermGroup

Gets the TermGroup (section 3.1.5.18) object related to a specified site collection or NULL if no such TermGroup (section 3.1.5.18) object exists and the createIfMissing parameter is false.
Parameters:

currentSite: The site collection containing the TermGroup (section 3.1.5.18) object to retrieve

Type: Microsoft.SharePoint.Client.Site

Microsoft.SharePoint.Client.Site is specified in [MS-CSOMSPT] section 3.2.5.119.

createIfMissing: Indicates whether to create a new TermGroup (section 3.1.5.18) object if one does not exist yet for the specified site collection.

Type: CSOM Boolean

3.1.5.23.2.2.9 GetTermInTermSet

Return Type: Microsoft.SharePoint.Client.Taxonomy.Term

Gets a Term (section 3.1.5.16) object that is retrieved based on the Id (section 3.1.5.20.1.1.6) of the TermSet (section 3.1.5.20) object and the Id (section 3.1.5.16.1.1.4) of the Term. Returns null if the Term cannot be found or if the user does not have sufficient permissions to access the Term.

Parameters:

termSetId: The Id (section 3.1.5.20.1.1.6) of the TermSet that the Term is a member of.

Type: CSOM GUID

termId: The Id (section 3.1.5.16.1.1.4) property of the Term to get.

Type: CSOM GUID

3.1.5.23.2.2.10 GetTermsById

Return Type: Microsoft.SharePoint.Client.Taxonomy.TermCollection

Gets a collection of Term (section 3.1.5.16) objects that are based on the Id (section 3.1.5.16.1.1.4) of the Term, or an empty collection or null if no terms match.

Remarks:

If a Term is a member of multiple TermSet (section 3.1.5.20) objects, it arbitrarily returns the Term from one of the TermSet objects. If an ID from the termIds parameter cannot be matched against the ID of a Term in this TermStore (section 3.1.5.23) object or if the user does not have sufficient permissions to access it, the Term will not be included in the result collection that is returned. This method searches for the Term across all of the child TermGroup (section 3.1.5.18) objects and TermSet objects that the user has permissions to read.

If the termIds parameter contains entries with the same ID, the returned item collection also contains duplicates. If this search finds no terms, then either null or an empty collection will be returned.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147467261</td>
<td>ArgumentException</td>
<td>The termIds parameter cannot be null.</td>
</tr>
</tbody>
</table>

Parameters:

termIds: The Ids (section 3.1.5.16.1.1.4) of the Term objects to be retrieved.
Type: CSOM array of CSOM GUID

3.1.5.23.2.2.11 GetTermSetsByName

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermSetCollection

This method retrieves a collection of all TermSet (section 3.1.5.20) objects in this TermStore (section 3.1.5.23) that the current user has permissions to read that have a matching TermSet name in the provided LCID.

**Remarks:**

The termSetName parameter cannot be null or empty, and MUST NOT exceed 255 characters in length. It also cannot contain any of the following illegal characters: ; " < > | & and the tab character. The search is case-insensitive.

To get the most accurate results, it is recommended that any pending changes be committed to the database using CommitAll (section 3.1.5.23.2.1.1) prior to executing this search.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147467261</td>
<td>ArgumentException</td>
<td>The termSetName parameter cannot be null or empty.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of the termSetName parameter is invalid. It contains invalid characters or is too long.</td>
</tr>
</tbody>
</table>

**Parameters:**

termSetName: The Name (section 3.1.5.20.1.1.10) of the TermSet to be retrieved.

Type: CSOM String

lcid: The language code identifier (LCID) for the language the TermSet name is in.

Type: CSOM Int32

3.1.5.23.2.2.12 GetTermSetsByTermLabel

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermSetCollection

This method retrieves a collection of all TermSet (section 3.1.5.20) objects in this TermStore (section 3.1.5.23) that the current user has permissions to read that contain Term (section 3.1.5.16) objects with labels matching or starting with each of the specified strings for a specified LCID.

**Remarks:**

The value of the labels in requiredTermLabels cannot be null or empty, and MUST NOT exceed 255 characters in length. They also MUST NOT contain any of the following illegal characters ; " < > | & and tab. The lcid value MUST be one of the working languages of the term store, specified by TermStore.Languages (section 3.1.5.23.1.1.4). The search is case-insensitive. A TermSet will be included in the returned collection if, for each label value in requiredTermLabels, the TermSet contains a Term with a label that matches or starts with the value for the specified lcid. There is no requirement for a single Term to contain all of the specified labels. To get the most accurate results, it is recommended that any pending changes be committed to the database using CommitAll (section 3.1.5.23.2.1.1) prior to executing this search.
Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The requiredTermLabels value cannot be null.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The labels in requiredTermLabels cannot be null or empty.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of one or more labels in requiredTermLabels is invalid. It probably contains invalid characters or is too long.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentOutOfRangeException</td>
<td>The lcid value MUST be one of the working languages in the current term store.</td>
</tr>
</tbody>
</table>

Parameters:

requiredTermLabels: A set of labels to match against the Term.Labels (section 3.1.5.16.1.2.1) of each TermSet.

Type: **CSOM array of CSOM String**

lcid: The language code identifier (LCID) for the language of the Term.Labels to perform the matching on.

Type: **CSOM Int32**

**3.1.5.23.2.2.13 GetTermSetsWithCustomProperty**

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermSetCollection

Gets a collection of all of the TermSet (section 3.1.5.20) objects in this TermStore (section 3.1.5.23) that the current user has permissions to read, that have a custom property matching the criteria provided.

**Remarks:**

The CustomPropertyMatchInformation (section 3.1.5.29) values that are set MUST be one of these combinations:

- CustomPropertyMatchInformation.CustomPropertyName.
- CustomPropertyMatchInformation.CustomPropertyName and CustomPropertyMatchInformation.CustomPropertyValue.
- CustomPropertyMatchInformation.CustomPropertyName, CustomPropertyMatchInformation.CustomPropertyValue, and CustomPropertyMatchInformation.StringMatchOption.
- CustomPropertyMatchInformation.CustomPropertyName, CustomPropertyMatchInformation.CustomPropertyValue, CustomPropertyMatchInformation.StringMatchOption, and CustomPropertyMatchInformation.ResultCollectionSize.

If the combination of CustomPropertyMatchInformation values is invalid, an ArgumentException or ArgumentNullException will be thrown indicating the invalid setting.
The `CustomPropertyMatchInformation.CustomPropertyName` (section 3.1.5.29.1.1.1) MUST NOT be set to null or empty, and MUST NOT exceed 255 characters in length. It also MUST NOT contain any of the following illegal characters: ; " < > | & tab

The `CustomPropertyMatchInformation.CustomPropertyValue` (section 3.1.5.29.1.1.2) MUST NOT be set to null or empty, and MUST NOT exceed 255 characters in length. It also MUST NOT contain the illegal characters \t or \n.

The search is case-insensitive, and will match either the entire value or the custom property name or value, or only check that it starts with the specified string depending on the value of the `CustomPropertyMatchInformation.StringMatchOption` (section 3.1.5.29.1.1.4) parameter. If the `CustomPropertyMatchInformation.StringMatchOption` is not set, it defaults to exact match.

The `CustomPropertyMatchInformation.ResultCollectionSize` (section 3.1.5.29.1.1.3) value MUST be greater than 0. The maximum number of results returned from this search is restricted by the `CustomPropertyMatchInformation.ResultCollectionSize` parameter. If `CustomPropertyMatchInformation.ResultCollectionSize` is not set, then a maximum of 100 results is returned.

To get the most accurate results, it is recommended that any pending changes be committed to the database using `CommitAll` (section 3.1.5.23.2.1.1) prior to executing this search.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>Incorrect combination of <code>CustomPropertyMatchInformation</code> settings provided.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of <code>CustomPropertyMatchInformation.CustomPropertyName</code> is invalid. It either contains invalid characters or is too long.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of <code>CustomPropertyMatchInformation.CustomPropertyValue</code> is invalid. It exceeds the maximum allowed length of 255.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value of <code>CustomPropertyMatchInformation.CustomPropertyValue</code> is invalid. It contains invalid character \t, \n, or both.</td>
</tr>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td><code>CustomPropertyMatchInformation.TrimUnavailable</code> MUST NOT be set.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>Some <code>CustomPropertyMatchInformation</code> settings MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The <code>CustomPropertyMatchInformation.CustomPropertyName</code> MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2147467261</td>
<td>ArgumentNullException</td>
<td>The <code>CustomPropertyMatchInformation.CustomPropertyValue</code> MUST NOT be null or empty.</td>
</tr>
<tr>
<td>-2146233086</td>
<td>ArgumentException</td>
<td>The <code>CustomPropertyMatchInformation.ResultCollectionSize</code> is invalid. It MUST be greater than 0.</td>
</tr>
</tbody>
</table>
Parameters:

_customPropertyMatchInformation_: Indicates the search criteria to use when retrieving **TermSet** objects.

Type: Microsoft.SharePoint.Client.Taxonomy.CustomPropertyMatchInformation

### 3.1.5.24 Microsoft.SharePoint.Client.Taxonomy.TermStoreCollection

**TypeId:** {E4A57A1E-7778-43FF-A88A-D5A61B231AFA}

**ShortName:** SP.Taxonomy.TermStoreCollection

Represents a collection of **TermStore** (section 3.1.5.23) objects.

### 3.1.5.24.1 Properties

#### 3.1.5.24.1.1 Scalar Properties

None.

#### 3.1.5.24.1.2 ObjectPath Properties

None.

### 3.1.5.24.2 Methods

#### 3.1.5.24.2.1 Scalar Methods

None.

#### 3.1.5.24.2.2 ObjectPath Methods

**3.1.5.24.2.2.1 GetById**

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermStore

Returns the **TermStore** (section 3.1.5.23) specified by its **GUID** index.

**Parameters:**

`index`: The **GUID** index of the **TermStore** to be returned.

Type: **CSOM GUID**

**3.1.5.24.2.2.2 GetByName**

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TermStore

Returns the **TermStore** (section 3.1.5.23) specified by its index name.

**Parameters:**

`index`: The index name of the **TermStore** to be returned.

Type: **CSOM String**
3.1.5.25  Microsoft.SharePoint.Client.Taxonomy.MobileTaxonomyField

**TypeId:** {21743F26-63E3-432D-9093-89973BB23664}

**ShortName:** SP.Taxonomy.MobileTaxonomyField

Provides support for taxonomy fields on a mobile page. A mobile page is a page that can be accessed by a mobile device.

### 3.1.5.25.1  Properties

#### 3.1.5.25.1.1  Scalar Properties

#### 3.1.5.25.1.1.1  ReadOnly

**Type:** CSOM Boolean

**Accessibility:** Read Only

A Boolean value that indicates whether the current MobileTaxonomyField (section 3.1.5.25) object is read-only.

#### 3.1.5.25.1.2  ObjectPath Properties

None.

### 3.1.5.25.2  Methods

#### 3.1.5.25.2.1  Scalar Methods

None.

#### 3.1.5.25.2.2  ObjectPath Methods

None.

3.1.5.26  Microsoft.SharePoint.Client.Taxonomy.ChangeInformation

**TypeId:** {1F849FB0-4FCB-4A54-9B01-9152B9E482D3}

**ShortName:** SP.Taxonomy.ChangeInformation

An object that supports overloads for TermGroup.GetChanges (section 3.1.5.18.2.2.2), TermSet.GetChanges (section 3.1.5.20.2.2.4), and TermStore.GetChanges (section 3.1.5.23.2.2.2) methods by combining the change information in one structure.

### 3.1.5.26.1  Properties

#### 3.1.5.26.1.1  Scalar Properties

#### 3.1.5.26.1.1.1  ItemType

**Type:** Microsoft.SharePoint.Client.Taxonomy.ChangedItemType

**Accessibility:** Read/Write

Specifies the ChangedItemType (section 3.1.5.4).
Default is **Unknown** (section 3.1.5.4.1.1).

### 3.1.5.26.1.1.2 OperationType

**Type:** Microsoft.SharePoint.Client.Taxonomy.ChangedOperationType  
**Accessibility:** Read/Write  
Specifies the ChangedOperationType (section 3.1.5.5).  
Default is **Unknown** (section 3.1.5.5.1.1).

### 3.1.5.26.1.1.3 StartTime

**Type:** CSOM DateTime  
**Accessibility:** Read/Write  
Specifies the date and time that indicate the earliest change to be returned.

### 3.1.5.26.1.1.4 WithinTimeSpan

**Type:** CSOM TimeSpan  
**Accessibility:** Read/Write  
Specifies a time span within which changes are returned.

### 3.1.5.26.1.2 ObjectPath Properties

None.

### 3.1.5.26.2 Methods

#### 3.1.5.26.2.1 Scalar Methods

None.

#### 3.1.5.26.2.2 ObjectPath Methods

##### 3.1.5.26.2.2.1 CSOM Constructor

Creates a ChangeInformation (section 3.1.5.26) object.  
**Parameters:** None

### 3.1.5.27 Microsoft.SharePoint.Client.Taxonomy.TaxonomyField

**TypeId:** {0831C0F3-C1B9-4D8A-A339-0160F42257B4}  
**ShortName:** SP.Taxonomy.TaxonomyField  
Represents a taxonomy field.

After setting the properties of TaxonomyField, the client MUST call the Update (section 3.1.5.27.2.1.5) method for the change to take effect in the database.

The NoCrawl and SchemaXmlWithResourceTokens properties are not included in the default scalar property set for this type.
3.1.5.27.1  Properties

3.1.5.27.1.1  Scalar Properties

3.1.5.27.1.1.1  AllowMultipleValues

*Type:* CSOM Boolean  
*Accessibility:* Read/Write

Specifies a Boolean value that specifies whether multiple Term (section 3.1.5.16) objects can be used in the TaxonomyField (section 3.1.5.27) object.

3.1.5.27.1.1.2  AnchorId

*Type:* CSOM GUID  
*Accessibility:* Read/Write

Specifies the GUID of the anchor Term (section 3.1.5.16) object for a TaxonomyField (section 3.1.5.27) object.

*Remarks:*  
An anchor Term is the Term in the hierarchical TermStore (section 3.1.5.23) that serves as the root node of the Term objects that are permissible for the current TaxonomyField object.

3.1.5.27.1.1.3  CanBeDeleted

*Type:* CSOM Boolean  
*Accessibility:* Read Only

Gets a Boolean value that indicates whether the TaxonomyField (section 3.1.5.27) can be deleted.

3.1.5.27.1.1.4  CreateValuesInEditForm

*Type:* CSOM Boolean  
*Accessibility:* Read/Write

Specifies a Boolean value that specifies whether the new Term (section 3.1.5.16) objects can be added to the TermSet (section 3.1.5.20) while typing in the TaxonomyField (section 3.1.5.27) editor control.

*Remarks:*  
If this property is true and the TermSet is open, then typing text in the taxonomy field editor control will display a Create New option in the Suggestions drop-down list if that Term does not already exist at the root level of the TermSet. Selecting the Create New option will create a new Term with that text as the default Label (section 3.1.5.10) object.

This property does not apply to the keywords term set, because it allows new Term objects to be added by selecting Create New in the Suggestions drop-down list.

It is not possible to create Term objects that are created by using the Create New at an arbitrary position in the hierarchy of Term objects in the TermSet. All Term objects created with this option are created at the root level of the TermSet.

3.1.5.27.1.1.5  DefaultValue
Type: **CSOM String**

**Accessibility:** Read/Write

Specifies the default value for a field.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>The list that the <strong>TaxonomyField</strong> belongs to has an external data source.</td>
</tr>
</tbody>
</table>

### 3.1.5.27.1.1.6 Description

**Type:** **CSOM String**

**Accessibility:** Read/Write

Specifies a description for a **TaxonomyField** (section 3.1.5.27).

### 3.1.5.27.1.1.7 Direction

**Type:** **CSOM String**

**Accessibility:** Read/Write

Gets or set the reading order for the **TaxonomyField** (section 3.1.5.27).

Returns a string that contains "LTR" if the reading order is left-to-right, "RTL" if it is right-to-left, or none.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>The list that the <strong>TaxonomyField</strong> belongs to has an external data source and direction is not equal to none.</td>
</tr>
</tbody>
</table>

### 3.1.5.27.1.1.8 EnforceUniqueValues

**Type:** **CSOM Boolean**

**Accessibility:** Read/Write

Specifies a **Boolean** value that specifies whether to enforce uniqueness in column values. The default is **false**.

**Remarks:**

The value cannot be set to **false** for the built-in **TaxonomyFields** (section 3.1.5.27) or for the site columns.

**Exceptions:**

---
### 3.1.5.27.1.1.9 FieldTypeKind

**Type:** Microsoft.SharePoint.Client.FieldType

**Accessibility:** Read/Write

Specifies the type of the **TaxonomyField** (section 3.1.5.27).

**Microsoft.SharePoint.Client.FieldType** is specified in [MS-CSOMSPT] section 3.2.5.58.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>The list has an external data source and trying to set the value to true.</td>
</tr>
</tbody>
</table>

### 3.1.5.27.1.1.10 Filterable

**Type:** CSOM Boolean

**Accessibility:** Read Only

There is a **Boolean** value that indicates whether the **TaxonomyField** (section 3.1.5.27) can be filtered.

**Remarks:**

This property is **true** if the **TaxonomyField** can be filtered; otherwise, **false**.

### 3.1.5.27.1.1.11 FromBaseType

**Type:** CSOM Boolean

**Accessibility:** Read Only

There is a **Boolean** value that indicates whether the **TaxonomyField** (section 3.1.5.27) derives from a base **TaxonomyField** type.

**Remarks:**

If the **FromBaseType** property returns **true**, the **TaxonomyField** cannot be converted to another **TaxonomyField** type, nor can it be deleted. This property is **true** if the **TaxonomyField** derives from a base **TaxonomyField** type; otherwise, **false**.

### 3.1.5.27.1.1.12 Group

**Type:** CSOM String

**Accessibility:** Read/Write
Specifies the column group to which the TaxonomyField (section 3.1.5.27) belongs.

It MUST NOT be NULL. It MUST NOT be empty. Its length MUST be equal to or less than 128.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>This TaxonomyField belongs to a list that has an external data source.</td>
</tr>
</tbody>
</table>

### 3.1.5.27.1.1.13 Hidden

**Type:** CSOM Boolean

**Accessibility:** Read/Write

Specifies a Boolean value that specifies whether the TaxonomyField (section 3.1.5.27) is displayed in the list.

This property is true if the TaxonomyField is hidden; otherwise, false.

### 3.1.5.27.1.1.14 Id

**Type:** CSOM GUID

**Accessibility:** Read Only

Gets the GUID of the TaxonomyField (section 3.1.5.27).

### 3.1.5.27.1.1.15 Indexed

**Type:** CSOM Boolean

**Accessibility:** Read Only

A Boolean value that specifies whether this TaxonomyField (section 3.1.5.27) is indexed.

**Remarks:**

This property is true to specify that the field is indexed; otherwise, false.

### 3.1.5.27.1.1.16 InternalName

**Type:** CSOM String

**Accessibility:** Read Only

Gets the internal name that is used for the TaxonomyField (section 3.1.5.27).

It MUST NOT be NULL. It MUST NOT be empty.

### 3.1.5.27.1.1.17 IsAnchorValid

**Type:** CSOM Boolean

**Accessibility:** Read Only
Gets a **Boolean** value that specifies whether the **Term** (section 3.1.5.16) object identified by the **AnchorId** (section 3.1.5.27.1.1.2) property is valid.

**Remarks:**

This property returns **true** if the **TaxonomyField** (section 3.1.5.27) points to the **keywords term set** (section 3.1.5.20), or if the **TaxonomyField** points to the entire **TermSet** (the **AnchorId** property returns **Guid.Empty**).

### 3.1.5.27.1.1.18 IsKeyword

**Type:** CSOM Boolean

**Accessibility:** Read/Write

Specifies a **Boolean** value that indicates whether the **TaxonomyField** (section 3.1.5.27) value points to the **keywords term set** (section 3.1.5.20) object.

**Remarks:**

The keywords term set is open and cannot contain hierarchical taxonomies. A **TaxonomyField** object that points to the keywords term set allows multiple values by default.

When editing a **TaxonomyField** object with this property set to **true**, new **Term** (section 3.1.5.16) objects can be added to the keywords term set by selecting **Create New** in the **Suggestions** drop-down menu in the **TaxonomyField** editor control.

Because the **TermSet** does not support hierarchical taxonomies, the **Term** picker dialog box cannot be opened to pick **Term** objects for a **TaxonomyField** that has this property set to **true**.

### 3.1.5.27.1.1.19 IsPathRendered

**Type:** CSOM Boolean

**Accessibility:** Read/Write

Specifies a **Boolean** value that specifies whether the default **Label** (section 3.1.5.10) objects of all the parent **Term** (section 3.1.5.16) objects of a **Term** in the **TaxonomyField** (section 3.1.5.27) object will be rendered in addition to the default label of that **Term**.

**Remarks:**

If this property is set to **true**, the **TaxonomyField** is rendered as a series of default labels separated by colon characters:

```
[RootLabel:....:ParentLabel:Label]
```

If this property is set to **false**, only the default **Label** of each **Term** is rendered.

If the **IsPathRendered** property is set, the **Update** (section 3.1.5.27.2.1.5) method MUST be called for changes to take effect in the database.

### 3.1.5.27.1.1.20 IsRelationship

**Type:** CSOM Boolean

**Accessibility:** Read/Write

Specifies a **Boolean** value that indicates whether a **TaxonomyField** (section 3.1.5.27) is discoverable from the list where it looks for its value.
Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>The property is read and the list that this field belongs to is an external list.</td>
</tr>
</tbody>
</table>

3.1.5.27.1.1.21  IsTermSetValid

Type: CSOM Boolean

Accessibility: Read Only

Gets a Boolean value that specifies whether the TermSet (section 3.1.5.20) object identified by the TermSetId (section 3.1.5.27.1.1.37) property exists and is available for tagging.

Remarks:

This property returns true if the TermSetId property points to the keywords term set.

3.1.5.27.1.1.22  LookupField

Type: CSOM String

Accessibility: Read/Write

Specifies the internal name of the TaxonomyField (section 3.1.5.27) in a related list that is the source of this TaxonomyField object’s value.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>The property is read and the list that this TaxonomyField belongs to is an external list.</td>
</tr>
</tbody>
</table>

3.1.5.27.1.1.23  LookupList

Type: CSOM String

Accessibility: Read/Write

Specifies the string representation of the GUID that identifies the list that is the source of this TaxonomyField (section 3.1.5.27) object’s value.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>The property is read and the list that this TaxonomyField belongs to is an external list.</td>
</tr>
</tbody>
</table>
3.1.5.27.1.1.24   **LookupWebId**

**Type:** CSOM GUID  

**Accessibility:** Read/Write

Specifies the identifier of the *website* that contains the list that is the source of this *TaxonomyField* (section 3.1.5.27) object’s value.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>The property is read and the list that this <em>TaxonomyField</em> belongs to is an external list.</td>
</tr>
</tbody>
</table>

3.1.5.27.1.1.25   **Open**

**Type:** CSOM Boolean  

**Accessibility:** Read/Write

Specifies a *Boolean* value that specifies whether the *TaxonomyField* (section 3.1.5.27) object is linked to an open *TermSet* (section 3.1.5.20) object or a closed *TermSet*.

**Remarks:**

A *TermSet* is open if new *Term* (section 3.1.5.16) objects can be added to it by users that do not have permissions to normally add to the *TermSet*. It is closed if new terms cannot be added to it by users that do not have permissions.

3.1.5.27.1.1.26   **PrimaryFieldId**

**Type:** CSOM String  

**Accessibility:** Read/Write

Specifies a string representation of the *GUID* that identifies the primary *lookup field* if the field is a dependent lookup field.

**Remarks:**

If the *TaxonomyField* (section 3.1.5.27) is a secondary field in a multi-column lookup, this property returns the string representation of the *GUID* that identifies the primary lookup field; otherwise, it returns an empty string.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>The list that this <em>TaxonomyField</em> belongs to is an external list.</td>
</tr>
</tbody>
</table>

3.1.5.27.1.1.27   **ReadOnlyField**

**Type:** CSOM Boolean
**Accessibility:** Read/Write

Specifies a **Boolean** value that specifies whether values in the **TaxonomyField** (section 3.1.5.27) can be modified.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>Thrown when attempting to change the value of <strong>ReadOnlyField</strong> when this <strong>TaxonomyField</strong> has an external data source.</td>
</tr>
</tbody>
</table>

### 3.1.5.27.1.1.28  RelationshipDeleteBehavior

**Type:** Microsoft.SharePoint.Client.RelationshipDeleteBehaviorType

**Accessibility:** Read/Write

Specifies the delete behavior of the **lookup field**.

**Microsoft.SharePoint.Client.RelationshipDeleteBehaviorType** is specified in [MS-CSOMSPT] section 3.2.5.109.

**Remarks:**

Possible values include **Cascade, Restrict** and **None**.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>The property is read and the list that this <strong>TaxonomyField</strong> (section 3.1.5.27) belongs to is an external list.</td>
</tr>
</tbody>
</table>

### 3.1.5.27.1.1.29  Required

**Type:** CSOM Boolean

**Accessibility:** Read/Write

Specifies a **Boolean** value that specifies whether the protocol client MUST enter a value for the **TaxonomyField** (section 3.1.5.27) when editing or creating **list items**.

**Remarks:**

This property is **true** if the **TaxonomyField** requires values; otherwise, **false**.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>This <strong>TaxonomyField</strong> belongs to a list that has an external data source and <strong>Required</strong> is set to <strong>true</strong>.</td>
</tr>
</tbody>
</table>
3.1.5.27.1.1.30  SchemaXml

**Type:** CSOM String

**Accessibility:** Read/Write

Specifies the XML schema that defines the TaxonomyField (section 3.1.5.27).

The XML schema MUST be an XML document that conforms to the FieldDefinitions type, as specified in [MS-WSSCAML] section 2.3.2.9, or a string that is null or empty.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>The list that the TaxonomyField belongs to has an external data source and the set value is not an empty or a null string.</td>
</tr>
</tbody>
</table>

3.1.5.27.1.1.31  Scope

**Type:** CSOM String

**Accessibility:** Read Only

Gets the website relative path to the list where the TaxonomyField (section 3.1.5.27) collection is used.

3.1.5.27.1.1.32  Sealed

**Type:** CSOM Boolean

**Accessibility:** Read/Write

Gets a Boolean value that specifies whether the TaxonomyField (section 3.1.5.27) can be removed from a content type or list.

3.1.5.27.1.1.33  Sortable

**Type:** CSOM Boolean

**Accessibility:** Read Only

Gets a Boolean value that specifies whether the TaxonomyField (section 3.1.5.27) can be sorted.

3.1.5.27.1.1.34  SspId

**Type:** CSOM GUID

**Accessibility:** Read/Write

Specifies the GUID that identifies the TermStore (section 3.1.5.23) object, which contains the Enterprise keywords for the site that the current TaxonomyField (section 3.1.5.27) belongs to.

**Remarks:**
If the GUID that identifies the TermStore cannot be found, a Guid.Empty object is returned.

3.1.5.27.1.1.35  StaticName

Type: CSOM String

Accessibility: Read/Write

Specifies a static name for the TaxonomyField (section 3.1.5.27).

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>This TaxonomyField belongs to a list that has an external data source, and StaticName is changed.</td>
</tr>
</tbody>
</table>

3.1.5.27.1.1.36  TargetTemplate

Type: CSOM String

Accessibility: Read/Write

Specifies the Web-relative Uniform Resource Locator (URL) of the target page that is used to construct the hyperlink on each Term (section 3.1.5.16) object when the TaxonomyField (section 3.1.5.27) object is rendered.

3.1.5.27.1.1.37  TermSetId

Type: CSOM GUID

Accessibility: Read/Write

Specifies the GUID of the TermSet (section 3.1.5.20) object that contains the Term (section 3.1.5.16) objects used by the current TaxonomyField () object.

Remarks:

If the site, the TaxonomySession (section 3.1.5.15) object, the TermStore (section 3.1.5.23) object, or TermSet object cannot be found, the property returns an empty object.

3.1.5.27.1.1.38  TextField

Type: CSOM GUID

Accessibility: Read Only

Specifies the GUID that identifies the hidden text field in an item.

Remarks:

Every TaxonomyField (section 3.1.5.27) object contains a related hidden text field that contains a string representation of the TaxonomyField value. The hidden text field is identified by the GUID returned by this property.

3.1.5.27.1.1.39  Title
Type: **CSOM String**

Accessibility: Read/Write

Specifies the display name for the **TaxonomyField** (section 3.1.5.27).

### 3.1.5.27.1.1.40 TypeAsString

**Type:** CSOM String

**Accessibility:** Read/Write

Gets or sets the type of the **TaxonomyField** (section 3.1.5.27) as a string value.

It MUST NOT be NULL. It MUST NOT be empty.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.NotSupportedException</td>
<td>The list has an external data source.</td>
</tr>
</tbody>
</table>

### 3.1.5.27.1.1.41 TypeDisplayName

**Type:** CSOM String

**Accessibility:** Read Only

Gets the display name of the **TaxonomyField** (section 3.1.5.27).

It MUST NOT be NULL. It MUST NOT be empty.

### 3.1.5.27.1.1.42 TypeShortDescription

**Type:** CSOM String

**Accessibility:** Read Only

Gets the description of the **TaxonomyField** (section 3.1.5.27).

It MUST NOT be NULL. It MUST NOT be empty.

### 3.1.5.27.1.1.43 UserCreated

**Type:** CSOM Boolean

**Accessibility:** Read/Write

Specifies a **Boolean** value that specifies whether the **TaxonomyField** (section 3.1.5.27) object is linked to a customized **TermSet** (section 3.1.5.20) object.

**Remarks:**

A **TermSet** is customized if it was created by using the **Customize your TermSet** option selected at the time the field was created.

**Term** (section 3.1.5.16) objects in a customized **TermSet** are available to other users in the site collection, but they are not offered as suggestions in Enterprise **keywords** columns.
3.1.5.27.1.1.44 ValidationFormula

Type: CSOM String

Accessibility: Read/Write

Indicates the formula referenced by the TaxonomyField (section 3.1.5.27) and is evaluated when a list item is added or updated.

3.1.5.27.1.1.45 ValidationMessage

Type: CSOM String

Accessibility: Read/Write

Specifies a message to display to the user if validation fails for this TaxonomyField (section 3.1.5.27).

Its length MUST be equal to or less than 1024.

3.1.5.27.1.1.46 EntityPropertyName

Type: CSOM String

Accessibility: Read Only

Gets the name of the entity property for the list item entity which uses this field.

It MUST NOT be NULL. It MUST NOT be empty.

3.1.5.27.1.1.47 JSLink

Type: CSOM String

Accessibility: Read/Write

Gets the name of an external ECMAScript (ECMA-262-1999) file containing any logic for this field.

3.1.5.27.1.1.48 SchemaXmlWithResourceTokens

Type: CSOM String

Accessibility: Read Only

Gets the schema XML for the content type containing resource tokens.

3.1.5.27.1.1.49 AutoIndexed

Type: CSOM Boolean

Accessibility: Read Only

Gets a Boolean value that specifies whether the TaxonomyField (section 3.1.5.27) object is auto indexed.

3.1.5.27.1.1.50 ClientSideComponentId

Type: CSOM GUID

Accessibility: Read/Write
3.1.5.27.1.1.51  ClientSideComponentProperties

Type: CSOM String

Accessibility: Read/Write

See [MS-CSOMSPT] section 3.2.5.37.1.1.31

3.1.5.27.1.1.52  CustomFormatter

Type: CSOM String

Accessibility: Read/Write

See [MS-CSOMSPT] section 3.2.5.37.1.1.32

3.1.5.27.1.1.53  DefaultFormula

Type: CSOM String

Accessibility: Read/Write

See [MS-CSOMSPT] section 3.2.5.37.1.1.33

3.1.5.27.1.1.54  DependentLookupInternalNames

Type: CSOM array of CSOM String

Accessibility: Read Only

See [MS-CSOMSPT] section 3.2.5.49.1.1.42

3.1.5.27.1.1.55  IsDependentLookup

Type: CSOM Boolean

Accessibility: Read Only

See [MS-CSOMSPT] section 3.2.5.49.1.1.43

3.1.5.27.1.1.56  NoCrawl

Type: CSOM Boolean

Accessibility: Read/Write

See [MS-CSOMSPT] section 3.2.5.37.1.1.35

3.1.5.27.1.1.57  PinnedToFiltersPane

Type: CSOM Boolean

Accessibility: Read/Write

See [MS-CSOMSPT] section 3.2.5.37.1.1.36

3.1.5.27.1.1.58  ShowInFiltersPane
**Type:** Microsoft.SharePoint.Client.ShowInFiltersPaneStatus

**Accessibility:** Read/Write

See [MS-CSOMSPT] section 3.2.5.37.1.1.37

3.1.5.27.1.1.59 **UnlimitedLengthInDocumentLibrary**

**Type:** CSOM Boolean

**Accessibility:** Read/Write

See [MS-CSOMSPT] section 3.2.5.49.1.1.47

3.1.5.27.1.2 ObjectPath Properties

3.1.5.27.1.2.1 **DefaultValueTyped**

**Type:** CSOM Object

**Accessibility:** Read Only

Gets the default TaxonomyFieldValueCollection (section 3.1.5.28) or the TaxonomyFieldValue (section 3.1.5.13) object.

**Remarks:**

If the DefaultValue property is null or an empty string, this property returns null.

Otherwise, if the AllowMultipleValues (section 3.1.5.27.1.1) property is true, this property returns a TaxonomyFieldValueCollection object with the value set to the DefaultValue property. If the AllowMultipleValues property is false, this property returns a TaxonomyFieldValue object with the value set to the DefaultValue property.

3.1.5.27.1.2.2 **DescriptionResource**

**Type:** Microsoft.SharePoint.Client.UserResource

**Accessibility:** Read Only

Gets the resource object corresponding to the Description property for the TaxonomyField (section 3.1.5.27).

3.1.5.27.1.2.3 **TitleResource**

**Type:** Microsoft.SharePoint.Client.UserResource

**Accessibility:** Read Only

Gets the resource object corresponding to the Title property for the TaxonomyField (section 3.1.5.27).

3.1.5.27.2 Methods

3.1.5.27.2.1 Scalar Methods

3.1.5.27.2.1.1 **DeleteObject**

**Return Type:** None
Deletes the **TaxonomyField** (section 3.1.5.27).

**Parameters:** None

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>System.InvalidOperationException</td>
<td>The <strong>TaxonomyField</strong> cannot be deleted because it does not allow deletion.</td>
</tr>
</tbody>
</table>

3.1.5.27.2.1.2 **GetFieldValueAsHtml**

**Return Type:** CSOM String

Returns the **TaxonomyField** (section 3.1.5.27) value in Hypertext Markup Language (HTML) format to render the **TaxonomyField** value directly on a page.

**Parameters:**

- `value:` An object of type **TaxonomyFieldValue** (section 3.1.5.13) or **TaxonomyFieldValueCollection** (section 3.1.5.28) that represents the value to convert.

**Type:** CSOM Object

**Remarks:**

If the **AllowMultipleValues** (section 3.1.5.27.1.1.1) property of the **TaxonomyField** object is false, the value parameter is expected to be of type **TaxonomyFieldValue** or **SPFieldLookupValue** (see Microsoft.SharePoint.Client.FieldLookupValue as specified in [MS-CSOMSPT] section 3.2.5.50), or have an accurate string representation returned by the **ToString** method.

If the **AllowMultipleValues** property of the **TaxonomyField** object is true, the value parameter is expected to be of type **TaxonomyFieldValueCollection** or **SPFieldLookupValueCollection** (a CSOM array of **SPFieldLookupValue**) or have an accurate string representation returned by the **ToString** method. The HTML returned by this method will contain each value in the collection, separated by a semicolon character.

If the value parameter is a **SPFieldLookupValue** object or **SPFieldLookupValueCollection** object, the returned string will be HTML-encoded.

If the value parameter is **null**, an empty string is returned.

3.1.5.27.2.1.3 **GetFieldValueAsText**

**Return Type:** CSOM String

Returns a plain text representation of the **TaxonomyField** (section 3.1.5.27) value.

**Parameters:**

- `value:` An object of type **TaxonomyFieldValue** (section 3.1.5.13) or **TaxonomyFieldValueCollection** (section 3.1.5.28) that represents the value to convert.

**Type:** CSOM Object

**Remarks:**
If the value parameter is of type **TaxonomyFieldValue**, this method returns its label.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value parameter was of a type other than TaxonomyFieldValue or TaxonomyFieldValueCollection.</td>
</tr>
</tbody>
</table>

### 3.1.5.27.2.1.4 GetValidatedString

**Return Type:** **CSOM String**

Validates the **TaxonomyField** (section 3.1.5.27) value object using logic specific to the **TaxonomyField** object, and returns a serialized string representation of the validated field value object.

**Parameters:**

- **value**: An object of type **TaxonomyFieldValue** (section 3.1.5.13) or **TaxonomyFieldValueCollection** (section 3.1.5.28) that represents the value to validate and serialize.

**Type:** **CSOM Object**

**Remarks:**

If the **AllowMultipleValues** (section 3.1.5.27.1.1.1) property of the **TaxonomyField** object is **false**, the value parameter is expected to be of type **TaxonomyFieldValue**.

If the **AllowMultipleValues** property of the **TaxonomyField** object is **true**, the value parameter is expected to be of type **TaxonomyFieldValueCollection** or **SPFieldLookupValueCollection**.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>SPFieldValidationException</td>
<td>The value parameter contains a TaxonomyFieldValue object with an invalid Term (section 3.1.5.16) GUID.</td>
</tr>
</tbody>
</table>

### 3.1.5.27.2.1.5 Update

**Return Type:** None

Updates the database with changes that are made to the **TaxonomyField** (section 3.1.5.27) object.

**Parameters:** None

### 3.1.5.27.2.1.6 UpdateAndPushChanges

**Return Type:** None

Updates the database with changes that are made to the **TaxonomyField** (section 3.1.5.27) and optionally propagates the changes to lists that implement the **TaxonomyField**.

**Parameters:**
pushChangesToLists: true to propagate changes to lists that implement the TaxonomyField; otherwise, false.

Type: CSOM Boolean

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2130246331</td>
<td>Microsoft.SharePoint.SPException</td>
<td>Trying to change the type of a field, when the field is changing to be indexed or has already been indexed.</td>
</tr>
<tr>
<td>-2130246238</td>
<td>Microsoft.SharePoint.SPException</td>
<td>Trying to remove a field from the index, when the field does not allow duplicate values.</td>
</tr>
<tr>
<td>-2130246236</td>
<td>Microsoft.SharePoint.SPException</td>
<td>Trying to change the type of a field that does not allow duplicate values.</td>
</tr>
</tbody>
</table>

3.1.5.27.2.1.7 ValidateSetValue

Return Type: None

Validates a TaxonomyField (section 3.1.5.27) value.

Parameters:

item: A list item that contains a TaxonomyField whose value is to be validated.

Type: Microsoft.SharePoint.Client.ListItem

It MUST NOT be NULL.

value: The input value that is to be validated.

Type: CSOM String

3.1.5.27.2.1.8 GetFieldValueAsTaxonomyFieldValue

Return Type: Microsoft.SharePoint.Client.Taxonomy.TaxonomyFieldValue

Converts the specified string value into a TaxonomyFieldValue (section 3.1.5.13) object.

Remarks:

The value parameter string MUST consist of an internal identifier, a label, and a path: [ID;#LABEL|PATH].

[ID] is the internal identifier of the list item.

[Label] is the default Label of the corresponding Term (section 3.1.5.16) in the TermStore (section 3.1.5.23).

[PATH] is a string of GUIDs delimited by a pipe (|) character and terminating in the GUID that identifies the Term: [ROOT_TERM_GUID|...|PARENT_TERM_GUID|TERM_GUID].

For example, [id;#label|path] represents a single value.

Exceptions:
**Error Code** | **Error Type Name** | **Condition**
---|---|---
-1 | SPFieldValidationException | The given GUID does not exist in the TermStore.

**Parameters:**

*value*: A formatted string containing identifier, label, and path information to convert into a TaxonomyFieldValue.

*Type*: CSOM String

### 3.1.5.27.2.1.9 SetFieldValueByCollection

**Return Type**: None

Sets the value of the corresponding multi-value field in the list item to the properties of the Term (section 3.1.5.16) objects in the specified collection, in the specified language.

**Remarks:**

Each Term has a maximum of one default Label (section 3.1.5.10) for every language in the TermStore (section 3.1.5.23) object. For each Term, the default Label in the default language of the TermStore object is used.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
</table>
|-2147024809 | ArgumentException | One or more arguments to this method are null, or the TaxonomyField (section 3.1.5.27) object does not allow multiple values.

**Parameters:**

*item*: The ListItem ([MS-CSOMSPT] section 3.2.5.87) object whose field is to be updated.

*Type*: Microsoft.SharePoint.Client.ListItem

*terms*: The collection of Term objects whose properties are to be used to update the field value.

*Type*: CSOM array of Microsoft.SharePoint.Client.Taxonomy.Term

*lcid*: The language code identifier (LCID) of the language of the default Label objects to use.

*Type*: CSOM Int32

### 3.1.5.27.2.1.10 SetFieldValueByTerm

**Return Type**: None

Sets the value of the corresponding field in the list item to the properties of the specified Term (section 3.1.5.16) in the specified language.

**Remarks:**


A **Term** has a maximum of one default **Label** (section 3.1.5.10) for every new language in the **TermStore** (section 3.1.5.23) object. The default **Label** used by this method is the default **Label** of the **Term** in the language specified by the lcid parameter, if one exists; otherwise, the default language of the **TermStore** is used.

If the **AllowMultipleValues** (section 3.1.5.27.1.1.1) property is true, the field will be assigned a **TaxonomyFieldValueCollection** (section 3.1.5.28) object. Otherwise, it will be assigned a **TaxonomyFieldValue** (section 3.1.5.13) object.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>One or more arguments to this method are null.</td>
</tr>
</tbody>
</table>

**Parameters:**

- **item:** The **ListItem** ([MS-CSOMSPT] section 3.2.5.87) object whose field is to be updated.
  Type: Microsoft.SharePoint.Client.ListItem

- **term:** The **Term** object whose properties are used to update the field value.
  Type: Microsoft.SharePoint.Client.Taxonomy.Term

- **lcid:** The **language code identifier (LCID)** of the language of the default **Label** to use.
  Type: **CSOM Int32**

### 3.1.5.27.2.1.11 SetFieldValueByTermCollection

**Return Type:** None

Sets the value of the corresponding multi-value field in the **list item** to the properties of the **Term** (section 3.1.5.16) objects in the specified collection, in the specified language.

**Remarks:**

Each **Term** has a maximum of one default **Label** (section 3.1.5.10) for every language in the **TermStore** (section 3.1.5.23) object. For each **Term**, the default **Label** in the default language of the **TermStore** is used.

The field will be assigned a **TaxonomyFieldValueCollection** (section 3.1.5.28) object.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>One or more arguments to this method are null, or the <strong>TaxonomyField</strong> (section 3.1.5.27) object does not allow multiple values.</td>
</tr>
</tbody>
</table>

**Parameters:**

- **item:** The **ListItem** ([MS-CSOMSPT] section 3.2.5.87) object whose field is to be updated.
  Type: Microsoft.SharePoint.Client.ListItem
**terms**: The collection of **Term** objects whose properties are to be used to update the field value.

Type: Microsoft.SharePoint.Client.Taxonomy.TermCollection

**lcid**: The **language code identifier (LCID)** of the language of the default **Label** objects to use.

Type: **CSOM Int32**

### 3.1.5.27.2.1.12 SetFieldValueByValue

**Return Type**: None

Sets the value of the corresponding field in the **list item** to the value of the specified **TaxonomyFieldValue** (section 3.1.5.13) object.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>One or more arguments to this method are null.</td>
</tr>
</tbody>
</table>

**Parameters:**

*item*: The **ListItem** ([MS-CSOMSPT] section 3.2.5.87) object whose field is to be updated.

Type: Microsoft.SharePoint.Client.ListItem

Microsoft.SharePoint.Client.ListItem is specified in [MS-CSOMSPT] section 3.2.5.87.

*taxValue*: The **TaxonomyFieldValue** object whose value is to be used to update this field.

Type: Microsoft.SharePoint.Client.Taxonomy.TaxonomyFieldValue

### 3.1.5.27.2.1.13 SetFieldValueByValueCollection

**Return Type**: None

Sets the value of the corresponding multi-valued field in the **list item** to the value of the specified **TaxonomyFieldValueCollection** (section 3.1.5.28) object.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>One or more arguments to this method are null.</td>
</tr>
<tr>
<td>-1</td>
<td>NotSupportedException</td>
<td>The field is not multi-valued.</td>
</tr>
</tbody>
</table>

**Parameters:**

*item*: The **ListItem** ([MS-CSOMSPT] section 3.2.5.87) object whose field is to be updated.

Type: Microsoft.SharePoint.Client.ListItem

*taxValueCollection*: The **TaxonomyFieldValueCollection** object whose value is to be used to update this field.
Type: Microsoft.SharePoint.Client.Taxonomy.TaxonomyFieldValueCollection

3.1.5.27.2.1.14  **SetShowInDisplayForm**

**Return Type:** None

Sets the corresponding field on the **list item** such that the field will be displayed in the **display form**.

**Parameters:**

*value:* A Boolean value that, when set to **true**, enables the field to be displayed in the display form. A value of **false** causes the field to not be displayed in the display form.

Type: **CSOM Boolean**

3.1.5.27.2.1.15  **SetShowInEditForm**

**Return Type:** None

Sets the corresponding field on the **list item** such that the field will be displayed in the **edit form**.

**Parameters:**

*value:* A Boolean value that, when set to **true**, enables the field to be displayed in the edit form. A value of **false** causes the field to not be displayed in the edit form.

Type: **CSOM Boolean**

3.1.5.27.2.1.16  **SetShowInNewForm**

**Return Type:** None

Sets the corresponding field on the **list item** such that the field will be displayed in the **new form**.

**Parameters:**

*value:* A Boolean that, when set to **true**, enables the field to be displayed in the new form. A value of **false** causes the field to not be displayed in the new form.

Type: **CSOM Boolean**

3.1.5.27.2.2  **ObjectPath Methods**

3.1.5.27.2.2.1  **GetFieldValueAsTaxonomyFieldValueCollection**

**Return Type:** Microsoft.SharePoint.Client.Taxonomy.TaxonomyFieldValueCollection

Converts the specified string value into a **TaxonomyFieldValueCollection** (section 3.1.5.28) object.

**Remarks:**

The value parameter string MUST be formatted as a series of one or more string tokens separated by the ;# delimiter. Each token consists of an internal identifier, a label, and a path; for example, [ID];#LABEL|PATH].

[ID] is the internal identifier of the **list item**.

[LABEL] is the default **Label** of the corresponding **Term** (section 3.1.5.16) in the **TermStore** (section 3.1.5.23).
[PATH] is a string of GUIDs delimited by a pipe (|) character and terminating in the GUID that identifies the Term: [ROOT_TERM_GUID]|...|PARENT_TERM_GUID|TERM_GUID.

For example, the string [id1;#label1|path1|id2;#label2|path2;#id3;#label3|path3] represents three TaxonomyField (section 3.1.5.27) values that can be parsed into a TaxonomyFieldValueCollection object.

The AllowMultipleValues (section 3.1.5.27.1.1.1) property determines whether the TaxonomyField object allows multiple values. If the TaxonomyField object does not allow multiple values, please use the GetFieldValueAsTaxonomyFieldValue (section 3.1.5.27.2.1.8) method instead.

Exceptions:

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>SPFieldValidationException</td>
<td>The given GUID does not exist in the TermStore.</td>
</tr>
<tr>
<td>-1</td>
<td>InvalidOperationException</td>
<td>Invalid operation. This TaxonomyField does not allow multiple values.</td>
</tr>
</tbody>
</table>

Parameters:

value: A formatted string containing identifier, label, and path information to convert into a TaxonomyFieldValueValueCollection.

Type: CSOM String

3.1.5.28    Microsoft.SharePoint.Client.Taxonomy.TaxonomyFieldValueValueCollection

TypeId: {C3DFAE10-F3BF-4894-9012-BB60665B6D91}

ShortName: SP.Taxonomy.TaxonomyFieldValueValueCollection

Represents the multi-value object for the taxonomy column.

3.1.5.28.1    Properties

3.1.5.28.1.1    Scalar Properties

None.

3.1.5.28.1.2    ObjectPath Properties

None.

3.1.5.28.2    Methods

3.1.5.28.2.1    Scalar Methods

3.1.5.28.2.1.1    PopulateFromLabelGuidPairs

Return Type: None

This method adds TaxonomyFieldValue (section 3.1.5.13) objects based on the parsing of the text.

Parameters:
Values to add as a string.

Type: **CSOM String**

**Remarks:**

Values MUST be added as label/Guid pairs. The format is as follows: label|GUID This can be repeated multiple times delimited by a ;. All **WssId** values will be set to -1.

### 3.1.5.28.2.2 ObjectPath Methods

#### 3.1.5.28.2.2.1 CSOM Constructor

Creates a new instance of a **TaxonomyFieldValueCollection** (section 3.1.5.28) that is bound to the **creatingField** parameter.

**Parameters:**

* **fieldValue**: Field that this value is bound to.
  
  Type: **CSOM String**

* **creatingField**: String to parse for the value.
  
  Type: **Microsoft.SharePoint.Client.Field**

The **fieldValue** MUST be in the format wss Id;#Label|Guid. This entry can be repeated for multiple values by repeating the ;# characters. If this value is not bound to a field, **TermGuid** (section 3.1.5.13.1.1.2) can throw a MemberAccessException.

### 3.1.5.29 **Microsoft.SharePoint.Client.Taxonomy.CustomPropertyMatchInformation**

**TypeId**: `{56747951-DF44-4BED-BF36-2B3BDF587F9}`

**ShortName**: SP.Taxonomy.CustomPropertyMatchInformation

This class is used as an input parameter for methods that retrieve objects based on specific custom property matching criteria.

**Remarks:**

For details about how to set each value of this class, please refer to the method that is using this class as an input parameter.

#### 3.1.5.29.1 Properties

#### 3.1.5.29.1.1 Scalar Properties

#### 3.1.5.29.1.1.1 CustomPropertyName

**Type**: **CSOM String**

**Accessibility**: Read/Write

The custom property name with which to match.

**Exceptions:**
### CustomPropertyValue

**Type:** CSOM String  
**Accessibility:** Read/Write  

The custom property value with which to match.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value cannot be set to null.</td>
</tr>
</tbody>
</table>

### ResultCollectionSize

**Type:** CSOM Int32  
**Accessibility:** Read/Write  

Indicates the maximum number of objects to include in the returned collection.

### StringMatchOption

**Type:** Microsoft.SharePoint.Client.Taxonomy.StringMatchOption  
**Accessibility:** Read/Write  

Indicates whether to search for custom property names and values that start with the specified strings, or search for those that exactly match the specified strings.

### TrimUnavailable

**Type:** CSOM Boolean  
**Accessibility:** Read/Write  

Indicates whether to trim out Term (section 3.1.5.16) objects that have the IsAvailableForTagging (section 3.1.5.16.1.1.5) property set to false.

### ObjectPath Properties

None.

### Methods

#### Scalar Methods

None.
3.1.5.29.2.2 ObjectPath Methods

3.1.5.29.2.2.1 CSOM Constructor

Constructs an instance of a CustomPropertyMatchInformation (section 3.1.5.29) object with no values set.

Parameters: None

3.1.5.30 Microsoft.SharePoint.Client.Taxonomy.LabelMatchInformation

TypeId: {61A1D689-2744-4EA3-A88B-C95BEE9803AA}

ShortName: SP.Taxonomy.LabelMatchInformation

This class is used as an input parameter for methods that retrieve objects based on specific label matching criteria.

For details about how to set each value of this class, please refer to the method that is using this class as an input parameter.

3.1.5.30.1 Properties

3.1.5.30.1.1 Scalar Properties

3.1.5.30.1.1.1 DefaultLabelOnly

Type: CSOM Boolean

Accessibility: Read/Write

Indicates whether only default labels need to be matched against in the search.

3.1.5.30.1.1.2 ExcludeKeyword

Type: CSOM Boolean

Accessibility: Read/Write

Indicates whether to exclude Term (section 3.1.5.16) objects that are in the keywords term set (section 3.1.5.20).

3.1.5.30.1.1.3 Lcid

Type: CSOM Int32

Accessibility: Read/Write

The language code identifier (LCID) of the Term.Labels (section 3.1.5.16.1.2.1) to be matched.

3.1.5.30.1.1.4 ResultCollectionSize

Type: CSOM Int32

Accessibility: Read/Write

Indicates the maximum number of objects to include in the returned collection.

3.1.5.30.1.1.5 StringMatchOption
**Type:** Microsoft.SharePoint.Client.Taxonomy.StringMatchOption

**Accessibility:** Read/Write

Indicates whether to search for Term.Labels (section 3.1.5.16.1.2.1) that start with the specified string, or those that have an exact match.

### 3.1.5.30.1.1.6 TermLabel

**Type:** CSOM String

**Accessibility:** Read/Write

The Label (section 3.1.5.10) of the Term (section 3.1.5.16) to get or set.

**Exceptions:**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Error Type Name</th>
<th>Condition</th>
</tr>
</thead>
<tbody>
<tr>
<td>-2147024809</td>
<td>ArgumentException</td>
<td>The value cannot be null.</td>
</tr>
</tbody>
</table>

### 3.1.5.30.1.1.7 TrimDeprecated

**Type:** CSOM Boolean

**Accessibility:** Read/Write

Indicates whether to trim out Term (section 3.1.5.16) objects that have Term.IsDeprecated (section 3.1.5.16.1.1.6) set to true.

### 3.1.5.30.1.1.8 TrimUnavailable

**Type:** CSOM Boolean

**Accessibility:** Read/Write

Indicates whether to trim out Term (section 3.1.5.16) objects that have the Term.IsAvailableForTagging (section 3.1.5.16.1.1.5) property set to false.

### 3.1.5.30.1.2 ObjectPath Properties

None.

### 3.1.5.30.2 Methods

#### 3.1.5.30.2.1 Scalar Methods

None.

#### 3.1.5.30.2.2 ObjectPath Methods

#### 3.1.5.30.2.2.1 CSOM Constructor

Constructs an instance of a LabelMatchInformation (section 3.1.5.30) object with no values set.

**Parameters:** None
3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.
4 Protocol Examples

The examples in this section demonstrate common usage scenarios around creating, adding, updating, and deleting terms in a term set. All of the examples use a group and term set that contains a hierarchical collection of conference room terms. The complete hierarchy is listed in the following table.

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Id</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conferences</td>
<td>Group</td>
<td>{6F3CA467-B915-4901-A38E-907D4A048712}</td>
</tr>
<tr>
<td>Conference Rooms</td>
<td>Term Set</td>
<td>{E143C875-5A3C-4C38-B13F-ABE2CEDC1573}</td>
</tr>
<tr>
<td>Auditoriums</td>
<td>Term</td>
<td>{63161CAE-A38C-4C3B-976B-71F76506FDSB}</td>
</tr>
<tr>
<td>Auditorium C</td>
<td>Term</td>
<td>{B72D8AE6-E6A6-47FE-82D8-099F070FD37}</td>
</tr>
<tr>
<td>Auditorium D</td>
<td>Term</td>
<td>{674E29CE-739C-491B-A119-C79A5AA7FA0C}</td>
</tr>
<tr>
<td>Auditorium E</td>
<td>Term</td>
<td>{984DC45A-139F-426F-B0BF-AA1CD275E024}</td>
</tr>
<tr>
<td>Halls</td>
<td>Term</td>
<td>{AA770CFB-EB06-4091-9B29-9F8933D5C2FF}</td>
</tr>
<tr>
<td>Hall A</td>
<td>Term</td>
<td>{80938290-52B2-4713-B32D-6E55D194E2EA}</td>
</tr>
<tr>
<td>Hall B</td>
<td>Term</td>
<td>{6CB787F9-5339-4024-9EF6-D88BC24C51D}</td>
</tr>
<tr>
<td>Hall C</td>
<td>Term</td>
<td>{28E647DB-BECF-4CAA-83EC-1A5293CE127A}</td>
</tr>
</tbody>
</table>

The example in section 4.1 creates the entire taxonomy using the ids as described in the previous table as needed. The remaining examples demonstrate adding, modifying, or removing terms from the term set. All of the examples use a TermStore (section 3.1.5.23) named "Managed Metadata Service Application Proxy".

4.1 Create a New Group and Term Set

In this example, a new group and term set is created. The term set represents a collection of conference rooms as described in section 4.

The operation is summarized in the following steps.

1. The "Managed Metadata Service Application Proxy" TermStore object is retrieved by calling the GetByName method on the TermStoreCollection object.
2. The CreateGroup method is called on the TermStore object to create the "Conferences" group.
3. The CreateTermSet method is called on the TermGroup object to create the "Conference Rooms" term set.
4. The CreateTerm method is called on the TermSet object and subsequent Term objects to construct the hierarchy of conference room terms.

Request:

```xml
<Request AddExpandoFieldTypeSuffix="true" SchemaVersion="15.0.0.0" LibraryVersion="15.0.0.0"
ApplicationName=".NET Library"
xmns="http://schemas.microsoft.com/sharepoint/clientquery/2009">
<Actions>
  <ObjectPath Id="2" ObjectPathId="1" />
  <ObjectPath Id="4" ObjectPathId="3" />
</Actions>
</Request>
```
<ObjectPath Id="6" ObjectPathId="5" />
<ObjectPath Id="8" ObjectPathId="7" />
<ObjectPath Id="10" ObjectPathId="9" />
<ObjectIdentityQuery Id="11" ObjectPathId="9" />
<ObjectPath Id="13" ObjectPathId="12" />
<ObjectIdentityQuery Id="16" ObjectPathId="14" />
<ObjectPath Id="18" ObjectPathId="17" />
<ObjectIdentityQuery Id="19" ObjectPathId="17" />
<ObjectPath Id="21" ObjectPathId="20" />
<ObjectIdentityQuery Id="22" ObjectPathId="20" />
<ObjectPath Id="24" ObjectPathId="23" />
<ObjectIdentityQuery Id="25" ObjectPathId="23" />
<ObjectPath Id="27" ObjectPathId="26" />
<ObjectIdentityQuery Id="28" ObjectPathId="26" />
<ObjectPath Id="30" ObjectPathId="29" />
<ObjectIdentityQuery Id="31" ObjectPathId="29" />
<ObjectPath Id="33" ObjectPathId="32" />
<ObjectIdentityQuery Id="34" ObjectPathId="32" />
<ObjectPath Id="36" ObjectPathId="35" />
<ObjectIdentityQuery Id="37" ObjectPathId="35" />
<ObjectPath Id="40" ObjectPathId="38" />
<ObjectIdentityQuery Id="43" ObjectPathId="41" />
</Actions>
<ObjectPaths>
<StaticProperty Id="1" TypeId="{3747adcd-a3c3-41b9-bfab-4a64dd2f1e0a}" Name="Current" />
<Property Id="3" ParentId="1" Name="Site" />
<Constructor Id="5" TypeId="{981cbc68-9edc-4f8d-872f-71146fcbdb84f}" />
<Parameters>
  <Parameter ObjectPathId="3" />
  <Parameter Type="Boolean">true</Parameter>
</Parameters>
<Constructor>
<Property Id="7" ParentId="5" Name="TermStores" />
<Method Id="9" ParentId="7" Name="GetByName">
<Parameters>
  <Parameter Type="String">Managed Metadata Service Application Proxy</Parameter>
</Parameters>
</Method>
<Constructor Id="12" TypeId="{981cbc68-9edc-4f8d-872f-71146fcbdb84f}" />
<Parameters>
  <Parameter ObjectPathId="3" />
  <Parameter Type="Boolean">true</Parameter>
</Parameters>
<Method Id="14" ParentId="9" Name="CreateGroup">
<Parameters>
  <Parameter Type="String">Conferences</Parameter>
</Parameters>
</Method>
<Method Id="17" ParentId="14" Name="CreateTermSet">
<Parameters>
  <Parameter Type="String">Conference Rooms</Parameter>
  <Parameter Type="Guid">{e143c875-5a3c-4c38-b13f-abe2cedc1573}</Parameter>
</Parameters>
</Method>
<Method Id="20" ParentId="17" Name="CreateTerm">
<Parameters>
  <Parameter Type="String">Auditoriums</Parameter>
  <Parameter Type="Int32">1033</Parameter>
  <Parameter Type="Guid">{63161cae-a38c-4c3b-976b-71f76506fd5b}</Parameter>
</Parameters>
</Method>
<Method Id="23" ParentId="20" Name="CreateTerm"
<Method Id="26" ParentId="20" Name="CreateTerm">
  <Parameters>
    <Parameter Type="String">Auditorium C</Parameter>
    <Parameter Type="Int32">1033</Parameter>
    <Parameter Type="Guid">{b72d8ae6-e6a6-47fe-82d8-099f070fdd37}</Parameter>
  </Parameters>
</Method>

<Method Id="29" ParentId="20" Name="CreateTerm">
  <Parameters>
    <Parameter Type="String">Auditorium D</Parameter>
    <Parameter Type="Int32">1033</Parameter>
    <Parameter Type="Guid">{03ae29ce-739c-491b-a119-c79a5aa7fa0c}</Parameter>
  </Parameters>
</Method>

<Method Id="32" ParentId="17" Name="CreateTerm">
  <Parameters>
    <Parameter Type="String">Halls</Parameter>
    <Parameter Type="Int32">1033</Parameter>
    <Parameter Type="Guid">{aa770cfb-eb06-4091-9860-9fb6913a4fcf}</Parameter>
  </Parameters>
</Method>

<Method Id="35" ParentId="32" Name="CreateTerm">
  <Parameters>
    <Parameter Type="String">Hall A</Parameter>
    <Parameter Type="Int32">1033</Parameter>
    <Parameter Type="Guid">{80938290-52b2-4713-b32d-6e55d194e2ea}</Parameter>
  </Parameters>
</Method>

<Method Id="38" ParentId="32" Name="CreateTerm">
  <Parameters>
    <Parameter Type="String">Hall B</Parameter>
    <Parameter Type="Int32">1033</Parameter>
    <Parameter Type="Guid">{6cb787f9-5339-4024-9efc-6d8bbc24c51d}</Parameter>
  </Parameters>
</Method>

<Method Id="41" ParentId="32" Name="CreateTerm">
  <Parameters>
    <Parameter Type="String">Hall C</Parameter>
    <Parameter Type="Int32">1033</Parameter>
    <Parameter Type="Guid">{28e647db-becf-4caa-83ec-1a5293ce127a}</Parameter>
  </Parameters>
</Method>

Response:

[
  {
    "SchemaVersion":"15.0.0.0","LibraryVersion":"15.0.3410.1000","ErrorInfo":null
  },2, {
    "IsNull":false
  },4, {
    "IsNull":false
  },6, {
    "IsNull":false
  },8, {
    "IsNull":false
  },10, {
    "IsNull":false
  },11, {
    "ObjectIdentity_":"fec14c62-7c3b-481b-851b-c80d7802b224:st:TKE6KUAnlkq7BYswGJADmQ=="
]
4.2 Retrieve a Term

In this example, the protocol client retrieves the term "Auditorium C" from the protocol server. The following operation retrieves the Term object (section 3.1.5.16) named "Auditorium C" by calling the GetTerms method (section 3.1.5.20.2.2.6) on the TermSet object (section 3.1.5.20).

Request:

```xml
<Request AddExpandoFieldTypeSuffix="true" SchemaVersion="15.0.0.0" LibraryVersion="15.0.0.0"
ApplicationName=".NET Library"
xmnl="http://schemas.microsoft.com/sharepoint/clientquery/2009">
<Actions>
<ObjectPath Id="2" ObjectPathId="1" />
<ObjectPath Id="4" ObjectPathId="3" />
<ObjectPath Id="6" ObjectPathId="5" />
<ObjectPath Id="8" ObjectPathId="7" />
<ObjectPath Id="10" ObjectPathId="9" />
<ObjectIdentityQuery Id="11" ObjectPathId="9" />
<ObjectPath Id="13" ObjectPathId="12" />
<Query Id="14" ObjectPathId="12">
<Query SelectAllProperties="false">
<Properties />
</Query>
<ChildItemQuery SelectAllProperties="false">
<Properties>
<Property Name="Name" ScalarProperty="true" />
</Properties>
</ChildItemQuery>
</Query>
</Actions>
<ObjectPaths>
<StaticProperty Id="1" TypeId="{3747adcd-a3c3-41b9-bfab-4a64dd2f1e0a}" Name="Current" />
<Property Id="3" ParentId="1" Name="Site" />
<Constructor Id="5" TypeId="{981cbe68-9edc-4f8d-872f-71146fcb84f}">
<Parameters>
<Parameter ObjectPathId="3" />
<Parameter Type="Boolean">true</Parameter>
</Parameters>
</Constructor>
<Property Id="7" ParentId="5" Name="TermStores" />
<Method Id="9" ParentId="7" Name="GetByName">
<Parameters>
<Parameter Type="String">Managed Metadata Service Application Proxy</Parameter>
</Parameters>
</Method>
<Method Id="12" ParentId="9" Name="GetTerms">
<Parameters>
<Parameter Type="String">Auditorium C</Parameter>
<Parameter Type="Boolean">true</Parameter>
</Parameters>
</Method>
</ObjectPaths>
</Request>
```

The protocol server returns a TermCollection object (section 3.1.5.17) that contains the Term object for "Auditorium C" and the requested Name property (section 3.1.5.16.1.1.12).

Response:

```json
[
{
"SchemaVersion": "15.0.0.0", "LibraryVersion": "15.0.3410.1000", "ErrorInfo": null
}, 2, {
```
4.3 Add a New Term

In this example, the protocol client adds a new term named "Hall D" to the conference rooms term set on the protocol server. The term will be added under the "Halls" term.

In the following operation, the protocol client retrieves the Term object (section 3.1.5.16) for "Halls" by passing the correct id to the GetTerm method (section 3.1.5.23.2.2.4) on the TermStore object (section 3.1.5.23).

Request:

```xml
<Request AddExpandoFieldTypeSuffix="true" SchemaVersion="15.0.0.0" LibraryVersion="15.0.0.0" ApplicationName=".NET Library" xmlns="http://schemas.microsoft.com/sharepoint/clientquery/2009">
  <Actions>
    <ObjectPath Id="91" ObjectPathId="90" />
  </Actions>
  <ObjectPaths>
    <Method Id="90" ParentId="9" Name="GetTerm">
      <Parameters>
        <Parameter Type="Guid">{aa770cfb-eb06-4091-9860-9fb6913a4fcf}</Parameter>
      </Parameters>
    </Method>
    <Identity Id="9" Name="fec14c62-7c3b-481b-c80d7802b224:st:TKE6KUAnlkq7BYswGJADmQ==" />
  </ObjectPaths>
</Request>
```

The protocol server returns the Term object for "Halls."

Response:

```
[
  {
    "SchemaVersion":"15.0.0.0","LibraryVersion":"15.0.3410.1000","ErrorInfo":null
  },91,
  {
    "IsNull":false
  },92,
```

"IsNull":false,
},4,
"IsNull":false,
},6,
"IsNull":false,
},8,
"IsNull":false,
},10,
"IsNull":false,
},11,
"ObjectIdentity_":"fec14c62-7c3b-481b-c80d7802b224:st:TKE6KUAnlkq7BYswGJADmQ=="
},13,
"IsNull":false,
},14,
"ObjectType_":"SP.Taxonomy.TermCollection","_Child_Items_":[
  {
    "ObjectType_":"SP.Taxonomy.Term","ObjectIdentity_":"fec14c62-7c3b-481b-c80d7802b224:te:TKE6KUAnlkq7BYswGJADmQGpI3ewiWBMcXAXIXG0JxHR1yEPPhF04TLE\"u002fqqLO3BVz5oott6b\m\u002fkeC2AmfBw\u002f\dNw==","Name":"Auditorium C"
  }
]
]
The protocol client then calls the CreateTerm method (section 3.1.5.16.2.2.3) on the Term object to create the "Hall D" term.

**Request:**

```
<Request AddExpandoFieldTypeSuffix="true" SchemaVersion="15.0.0.0" ApplicationName=".NET Library" xmlns="http://schemas.microsoft.com/sharepoint/clientquery/2009">
  <Actions>
    <ObjectPath Id="95" ObjectPathId="94" />
    <ObjectIdentityQuery Id="96" ObjectPathId="94" />
    <Method Name="CommitAll" Id="97" ObjectPathId="9" />
  </Actions>
  <ObjectPaths>
    <Method Id="94" ParentId="90" Name="CreateTerm">
      <Parameters>
        <Parameter Type="String">Hall D</Parameter>
        <Parameter Type="Int32">1033</Parameter>
        <Parameter Type="Guid">{32d50796-39ed-44bb-9fb1-b991a532b128}</Parameter>
      </Parameters>
    </Method>
    <Identity Id="9" Name="fec14c62-7c3b-481b-851b-c80d7802b224:st:TKE6KUAnlkq7BYswGJAdmQGpI3ewiWBMvXAXIgOJxHR1yEFhPFo4TLE\u002fq+LO3BVz+wx3qgbrkUCYYJ+2kTpPzw==" />
    <Identity Id="90" Name="fec14c62-7c3b-481b-851b-c80d7802b224:te:TKE6KUAnlkq7BYswGJAdmQGpI3ewiWBMvXAXIgOJxHR1yEFhPFo4TLE/q+LO3BVz+wx3qgbrkUCYYJ+2kTpPzw=="/>
  </ObjectPaths>
</Request>
```

The protocol server returns the Term object of the new term.

**Response:**

```
[
  {
    "SchemaVersion":"15.0.0.0","LibraryVersion":"15.0.3410.1000","ErrorInfo":null,
    "IsNull":false,
    "ObjectIdentity":"
```

4.4 Delete an Existing Term

In this example, the protocol client deletes the term "Auditorium C" from the conference rooms term set on the protocol server.

In the following operation, the protocol client retrieves the Term object (section 3.1.5.16) for "Auditorium C" by passing the correct id to the GetTerm method (section 3.1.5.23.2.2.4) on the TermStore object (section 3.1.5.23).

**Request:**

```
<Request AddExpandoFieldTypeSuffix="true" SchemaVersion="15.0.0.0" LibraryVersion="15.0.3410.1000" ApplicationName=".NET Library" xmlns="http://schemas.microsoft.com/sharepoint/clientquery/2009">
  <Actions>
    <ObjectPath Id="95" ObjectPathId="94" />
    <ObjectIdentityQuery Id="96" ObjectPathId="94" />
    <Method Name="CommitAll" Id="97" ObjectPathId="9" />
  </Actions>
  <ObjectPaths>
    <Method Id="94" ParentId="90" Name="CreateTerm">
      <Parameters>
        <Parameter Type="String">Hall D</Parameter>
        <Parameter Type="Int32">1033</Parameter>
        <Parameter Type="Guid">{32d50796-39ed-44bb-9fb1-b991a532b128}</Parameter>
      </Parameters>
    </Method>
    <Identity Id="9" Name="fec14c62-7c3b-481b-851b-c80d7802b224:st:TKE6KUAnlkq7BYswGJAdmQGpI3ewiWBMvXAXIgOJxHR1yEFhPFo4TLE\u002fq+LO3BVz+wx3qgbrkUCYYJ+2kTpPzw==" />
    <Identity Id="90" Name="fec14c62-7c3b-481b-851b-c80d7802b224:te:TKE6KUAnlkq7BYswGJAdmQGpI3ewiWBMvXAXIgOJxHR1yEFhPFo4TLE/q+LO3BVz+wx3qgbrkUCYYJ+2kTpPzw=="/>
  </ObjectPaths>
</Request>
```
The protocol server returns the Term object for "Auditorium C".

Response:

```xml
[  
  
  
  {  
    "SchemaVersion": "15.0.0.0",  
    "LibraryVersion": "15.0.3410.1000",  
    "ErrorInfo": null  
  },  
  356,  
  {  
    "IsNull": false  
  },  
  357,  
  {  
    "ObjectIdentity": "fec14c62-7c3b-481b-851b-c80d7802b224:te:TKE6KUAnlkq7BYswGJADmQGpI3ewiWBMvXAXIg0JxK8yEPhPFo4TLE\u002fq+LO3VZ6DCp8eJ3dU1ojJ6nQaWEnw="  
  }  
]
```

The protocol client then calls the DeleteObject method (section 3.1.5.16.2.1.2) on the Term object. It also calls the CommitAll method (section 3.1.5.23.2.1.1) on the TermStore object to commit the change.

Request:

```xml
<Request AddExpandoFieldTypeSuffix="true" SchemaVersion="15.0.0.0" LibraryVersion="15.0.0.0" ApplicationName=".NET Library" xmlns="http://schemas.microsoft.com/sharepoint/clientquery/2009">
  <Actions>
    <Method Name="DeleteObject" Id="359" ObjectPathId="355" />
    <Method Name="CommitAll" Id="360" ObjectPathId="9" />
  </Actions>
  <ObjectPaths>
    <Identity Id="355" Name="fec14c62-7c3b-481b-851b-c80d7802b224:te:TKE6KUAnlkq7BYswGJADmQGpI3ewiWBMvXAXIg0JxK8yEPhPFo4TLE\u002fq+LO3VZ6DCp8eJ3dU1ojJ6nQaWEnw=" />
    <Identity Id="9" Name="fec14c62-7c3b-481b-851b-c80d7802b224:st:TKE6KUAnlkq7BYswGJADmQGpI3ewiWBMvXAXIg0JxK8yEPhPFo4TLE\u002fq+LO3VZ6DCp8eJ3dU1ojJ6nQaWEnw=" />
  </ObjectPaths>
</Request>
```

The protocol server returns no errors.

Response:

```
[
]
```
4.5  Add a New Label

In this example, the protocol client modifies the term "Hall A" on the protocol server by adding a new label named "Baker."

In the following operation, the protocol client retrieves the Term object (section 3.1.5.16) for "Hall A" by passing the correct id to the GetTerm method (section 3.1.5.23.2.2.4) on the TermStore object (section 3.1.5.23).

Request:

```
<Request AddExpandoFieldTypeSuffix="true" SchemaVersion="15.0.0.0" LibraryVersion="15.0.0.0"
ApplicationName=".NET Library"
xmlns="http://schemas.microsoft.com/sharepoint/clientquery/2009">
  <Actions>
    <ObjectPath Id="439" ObjectPathId="438" />
    <ObjectIdentityQuery Id="440" ObjectPathId="438" />
  </Actions>
  <ObjectPaths>
    <Method Id="438" ParentId="9" Name="GetTerm">
      <Parameters>
        <Parameter Type="Guid">{80938290-52b2-4713-b32d-6e55d194e2ea}</Parameter>
      </Parameters>
    </Method>
    <Identity Id="9" Name="fec14c62-7c3b-481b-851b-c80d7802b224:st:TKE6KUAnlkq7BYswGJADmQGpI3ewiWBMvXAXIgOJxHR1yEPhF0o4TLE\u002fq+LO3BVzkIKTgLJSE0ezLW5V82T16g==" />
  </ObjectPaths>
</Request>
```

The protocol server returns the Term object for "Hall A."

Response:

```
[
  {
    "SchemaVersion": "15.0.0.0", "LibraryVersion": "15.0.3410.1000", "ErrorInfo": "null",
  }, 439,
  {"IsNull": "false",
   "ObjectClass": "term",
   "ObjectIdentity": "fec14c62-7c3b-481b-851b-c80d7802b224:st:TKE6KUAnlkq7BYswGJADmQGpI3ewiWBMvXAXIgOJxHR1yEPhF0o4TLE\u002fq+LO3BVzkIKTgLJSE0ezLW5V82T16g=="
  }, 440,
]
```

The protocol client then calls the CreateLabel method (section 3.1.5.16.2.2.2) on the Term object to create the "Baker" label.

Request:

```
<Request AddExpandoFieldTypeSuffix="true" SchemaVersion="15.0.0.0" LibraryVersion="15.0.0.0"
ApplicationName=".NET Library"
xmlns="http://schemas.microsoft.com/sharepoint/clientquery/2009">
  <Actions>
    <ObjectPath Id="443" ObjectPathId="442" />
  </Actions>
</Request>
```
The protocol server returns the new **Label** object (section 3.1.5.10).

**Response:**

```xml
<ObjectIdentityQuery Id="444" ObjectPathId="442" />
<Method Name="CommitAll" Id="445" ObjectPathId="9" />
</Actions>
<ObjectPaths>
  <Method Id="442" ParentId="438" Name="CreateLabel">
    <Parameters>
      <Parameter Type="String">Baker</Parameter>
      <Parameter Type="Int32">1033</Parameter>
      <Parameter Type="Boolean">false</Parameter>
    </Parameters>
  </Method>
  <Identity Id="9" Name="fec14c62-7c3b-481b-851b-c80d7802b224:st:TKE6KUAnlkq7ByswGJAdmQ==" />
  <Identity Id="438" Name="fec14c62-7c3b-481b-851b-c80d7802b224:te:TKE6KUAnlkq7ByswGJAdmQGpI3ewiWBMvXAXIgOJxHR1yEPhFFo4TLE/q+LO3BVzkIKTgLJSE0ezLW5V0ZT16g==" />
</ObjectPaths>
</Request>
```
5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.
6 Appendix A: 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 SharePoint Server 2013
- Microsoft SharePoint Server 2016
- Microsoft SharePoint Server 2019

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.

<1> Section 3.1.5.23.1.1.7: Content type syndication hub is a feature of SharePoint Server 2013, which provisions a site to be an Enterprise Metadata hub site.
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.

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Revision class</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.5.16.1.1.13 Owner</td>
<td>Removed two error codes.</td>
<td>Minor</td>
</tr>
<tr>
<td>3.1.5.20.1.1.13 Names</td>
<td>Clarified description of this read-only property.</td>
<td>Minor</td>
</tr>
</tbody>
</table>
8 Index

A
Abstract data model
  server 20
Add a new label example 150
Applicability 18

C
Capability negotiation 18
Change tracking 154
Create a new group and term set example 142

D
Data model - abstract
  server 20
Delete an existing term example 148

E
Examples
  add a new label 150
  add a new term 147
  create a new group and term set 142
  delete an existing term 148
  overview 142
  retrieve a term 146

F
Fields - vendor-extensible 18

G
Glossary 14

H
Higher-layer triggered events
  server 20

I
Implementer - security considerations 152
Index of security parameters 152
Informative references 17
Initialization
  server 20
  Introduction 14

M
Message processing - server 20
Messages
  transport 19

N
Normative references 17

O
Other local events
  server 141
  Overview (synopsis) 17

P
Parameters - security index 152
Preconditions 18
Prerequisites 18
Product behavior 153

R
References 17
  informative 17
  normative 17
Relationship to other protocols 17
Retrieve a term example 146

S
Security
  implementer considerations 152
  parameter index 152
Sequencing rules - server 20
Server
  abstract data model 20
  higher-layer triggered events 20
  initialization 20
  message processing 20
  other local events 141
  sequencing rules 20
  timer events 141
  timers 20
Standards assignments 18

T
Timer events
  server 141
Timers
  server 20
Tracking changes 154
Transport 19
Triggered events - higher-layer
  server 20

V
Vendor-extensible fields 18
Versioning 18