

[MS-LISTSWS]: Lists Web Service Protocol Specification

Intellectual Property Rights Notice for Open Specifications Documentation

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

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

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

Preliminary Documentation. This Open Specification provides documentation for past and current releases and/or for the pre-release (beta) version of this technology. This Open Specification is final

documentation for past or current releases as specifically noted in the document, as applicable; it is preliminary documentation for the pre-release (beta) versions. Microsoft will release final documentation in connection with the commercial release of the updated or new version of this technology. As the documentation may change between this preliminary version and the final version of this technology, there are risks in relying on preliminary documentation. To the extent that you incur additional development obligations or any other costs as a result of relying on this preliminary documentation, you do so at your own risk.

Revision Summary

Date	Revision History	Revision Class	Comments
04/04/2008	0.1		Initial Availability
06/27/2008	1.0	Major	Revised and edited the technical content
12/12/2008	1.01	Editorial	Revised and edited the technical content
07/13/2009	1.02	Major	Revised and edited the technical content
08/28/2009	1.03	Editorial	Revised and edited the technical content
11/06/2009	1.04	Editorial	Revised and edited the technical content
02/19/2010	2.0	Major	Updated and revised the technical content
03/31/2010	2.01	Editorial	Revised and edited the technical content
04/30/2010	2.02	Editorial	Revised and edited the technical content
06/07/2010	2.03	Editorial	Revised and edited the technical content
06/29/2010	2.04	Major	Significantly changed the technical content.
07/23/2010	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
09/27/2010	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	2.05	Major	Significantly changed the technical content.
12/17/2010	2.05	No change	No changes to the meaning, language, or formatting of the technical content.
03/18/2011	2.05	No change	No changes to the meaning, language, or formatting of the technical content.
06/10/2011	2.05	No change	No changes to the meaning, language, or formatting of the technical content.
01/20/2012	3.0	Major	Significantly changed the technical content.
04/11/2012	3.0	No change	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1 Introduction	9
1.1 Glossary	9
1.2 References	11
1.2.1 Normative References	11
1.2.2 Informative References	12
1.3 Protocol Overview (Synopsis)	12
1.3.1 Roles	12
1.3.1.1 Protocol Server	12
1.3.1.2 Protocol Clients	13
1.3.2 Scenarios	13
1.3.2.1 List Schema Management	13
1.3.2.2 Content Type Management	13
1.3.2.3 List Data Management	14
1.3.2.3.1 List Items	14
1.3.2.3.2 Attachments	15
1.3.2.3.3 Documents	15
1.4 Relationship to Other Protocols	16
1.5 Prerequisites/Preconditions	16
1.6 Applicability Statement	16
1.7 Versioning and Capability Negotiation	17
1.8 Vendor-Extensible Fields	17
1.9 Standards Assignments	17
2 Messages	18
2.1 Transport	18
2.2 Common Message Syntax	18
2.2.1 Namespaces	18
2.2.2 Messages	19
2.2.3 Elements	19
2.2.4 Complex Types	19
2.2.4.1 AddOrUpdateFieldDefinition	19
2.2.4.2 AddOrUpdateFieldsDefinition	20
2.2.4.3 CamlContains	21
2.2.4.4 CamlQueryOptions	22
2.2.4.5 CamlViewFields	26
2.2.4.6 ContentTypePropertyDefinition	26
2.2.4.7 DataDefinition	27
2.2.4.8 DeleteFieldDefinition	27
2.2.4.9 DeleteFieldsDefinition	27
2.2.4.10 FieldReferenceDefinitionCT	28
2.2.4.11 ListDefinitionCT	29
2.2.4.12 ListDefinitionSchema	34
2.2.4.13 ListItemChangeDefinition	36
2.2.4.14 UpdateListFieldResults	37
2.2.4.15 UpdateListFieldsRequest	37
2.2.4.16 ViewChangeDefinition	38
2.2.5 Simple Types	38
2.2.5.1 ChangeTypeEnum	39
2.2.5.2 EnumOptimizeFor	40
2.2.5.3 EnumRecurrencePatternXMLVersion	40

2.2.5.4 EnumViewAttributes.....	41
2.2.5.5 MethodCmdEnum	41
2.2.5.6 OnErrorEnum	42
2.2.5.7 TRUEONLY	42
2.2.6 Attributes.....	42
2.2.7 Groups.....	42
2.2.8 Attribute Groups	43
2.2.9 Common Data Structures	43

3 Protocol Details..... 44

3.1 Lists Soap Server Details	44
3.1.1 Abstract Data Model	44
3.1.2 Timers	44
3.1.3 Initialization	44
3.1.4 Message Processing Events and Sequencing Rules.....	44
3.1.4.1 AddAttachment	46
3.1.4.1.1 Messages	47
3.1.4.1.1.1 AddAttachmentSoapIn	47
3.1.4.1.1.2 AddAttachmentSoapOut	47
3.1.4.1.2 Elements.....	47
3.1.4.1.2.1 AddAttachment	48
3.1.4.1.2.2 AddAttachmentResponse.....	48
3.1.4.2 AddDiscussionBoardItem	48
3.1.4.2.1 Messages	49
3.1.4.2.1.1 AddDiscussionBoardItemSoapIn	49
3.1.4.2.1.2 AddDiscussionBoardItemSoapOut	49
3.1.4.2.2 Elements.....	49
3.1.4.2.2.1 AddDiscussionBoardItem.....	50
3.1.4.2.2.2 AddDiscussionBoardItemResponse.....	50
3.1.4.3 AddList.....	51
3.1.4.3.1 Messages	52
3.1.4.3.1.1 AddListSoapIn.....	52
3.1.4.3.1.2 AddListSoapOut.....	52
3.1.4.3.2 Elements.....	52
3.1.4.3.2.1 AddList.....	53
3.1.4.3.2.2 AddListResponse	53
3.1.4.4 AddListFromFeature	53
3.1.4.4.1 Messages	54
3.1.4.4.1.1 AddListFromFeatureSoapIn	54
3.1.4.4.1.2 AddListFromFeatureSoapOut.....	55
3.1.4.4.2 Elements.....	55
3.1.4.4.2.1 AddListFromFeature.....	55
3.1.4.4.2.2 AddListFromFeatureResponse.....	56
3.1.4.5 AddWikiPage.....	56
3.1.4.5.1 Messages	56
3.1.4.5.1.1 AddWikiPageSoapIn	56
3.1.4.5.1.2 AddWikiPageSoapOut.....	57
3.1.4.5.2 Elements.....	57
3.1.4.5.2.1 AddWikiPage.....	57
3.1.4.5.2.2 AddWikiPageResponse.....	57
3.1.4.6 ApplyContentTypeToList	58
3.1.4.6.1 Messages	58
3.1.4.6.1.1 ApplyContentTypeToListSoapIn	58

3.1.4.6.1.2	ApplyContentTypeToListSoapOut	59
3.1.4.6.2	Elements	59
3.1.4.6.2.1	ApplyContentTypeToList	59
3.1.4.6.2.2	ApplyContentTypeToListResponse	60
3.1.4.7	CheckInFile	60
3.1.4.7.1	Messages	61
3.1.4.7.1.1	CheckInFileSoapIn	61
3.1.4.7.1.2	CheckInFileSoapOut	61
3.1.4.7.2	Elements	61
3.1.4.7.2.1	CheckInFile	62
3.1.4.7.2.2	CheckInFileResponse	62
3.1.4.8	CheckOutFile	62
3.1.4.8.1	Messages	63
3.1.4.8.1.1	CheckOutFileSoapIn	63
3.1.4.8.1.2	CheckOutFileSoapOut	63
3.1.4.8.2	Elements	64
3.1.4.8.2.1	CheckOutFile	64
3.1.4.8.2.2	CheckOutFileResponse	64
3.1.4.9	CreateContentType	65
3.1.4.9.1	Messages	65
3.1.4.9.1.1	CreateContentTypeSoapIn	66
3.1.4.9.1.2	CreateContentTypeSoapOut	66
3.1.4.9.2	Elements	66
3.1.4.9.2.1	CreateContentType	66
3.1.4.9.2.2	CreateContentTypeResponse	67
3.1.4.10	DeleteAttachment	67
3.1.4.10.1	Messages	68
3.1.4.10.1.1	DeleteAttachmentSoapIn	68
3.1.4.10.1.2	DeleteAttachmentSoapOut	69
3.1.4.10.2	Elements	69
3.1.4.10.2.1	DeleteAttachment	69
3.1.4.10.2.2	DeleteAttachmentResponse	69
3.1.4.11	DeleteContentType	70
3.1.4.11.1	Messages	70
3.1.4.11.1.1	DeleteContentTypeSoapIn	70
3.1.4.11.1.2	DeleteContentTypeSoapOut	71
3.1.4.11.2	Elements	71
3.1.4.11.2.1	DeleteContentType	71
3.1.4.11.2.2	DeleteContentTypeResponse	71
3.1.4.12	DeleteContentTypeXmlDocument	72
3.1.4.12.1	Messages	72
3.1.4.12.1.1	DeleteContentTypeXmlDocumentSoapIn	73
3.1.4.12.1.2	DeleteContentTypeXmlDocumentSoapOut	73
3.1.4.12.2	Elements	73
3.1.4.12.2.1	DeleteContentTypeXmlDocument	73
3.1.4.12.2.2	DeleteContentTypeXmlDocumentResponse	74
3.1.4.13	DeleteList	74
3.1.4.13.1	Messages	75
3.1.4.13.1.1	DeleteListSoapIn	75
3.1.4.13.1.2	DeleteListSoapOut	75
3.1.4.13.2	Elements	75
3.1.4.13.2.1	DeleteList	76
3.1.4.13.2.2	DeleteListResponse	76

3.1.4.14	GetAttachmentCollection.....	76
3.1.4.14.1	Messages	77
3.1.4.14.1.1	GetAttachmentCollectionSoapIn.....	77
3.1.4.14.1.2	GetAttachmentCollectionSoapOut	77
3.1.4.14.2	Elements	77
3.1.4.14.2.1	GetAttachmentCollection	78
3.1.4.14.2.2	GetAttachmentCollectionResponse	78
3.1.4.15	GetList	79
3.1.4.15.1	Messages	79
3.1.4.15.1.1	GetListSoapIn	79
3.1.4.15.1.2	GetListSoapOut	79
3.1.4.15.2	Elements.....	80
3.1.4.15.2.1	GetList	80
3.1.4.15.2.2	GetListResponse	80
3.1.4.16	GetListAndView	81
3.1.4.16.1	Messages	81
3.1.4.16.1.1	GetListAndViewSoapIn	82
3.1.4.16.1.2	GetListAndViewSoapOut	82
3.1.4.16.2	Elements.....	82
3.1.4.16.2.1	GetListAndView	82
3.1.4.16.2.2	GetListAndViewResponse	83
3.1.4.17	GetListCollection	83
3.1.4.17.1	Messages	83
3.1.4.17.1.1	GetListCollectionSoapIn.....	84
3.1.4.17.1.2	GetListCollectionSoapOut.....	84
3.1.4.17.2	Elements	84
3.1.4.17.2.1	GetListCollection	84
3.1.4.17.2.2	GetListCollectionResponse	84
3.1.4.18	GetListContentType.....	85
3.1.4.18.1	Messages	86
3.1.4.18.1.1	GetListContentTypeSoapIn.....	86
3.1.4.18.1.2	GetListContentTypeSoapOut.....	86
3.1.4.18.2	Elements	86
3.1.4.18.2.1	GetListContentType.....	86
3.1.4.18.2.2	GetListContentTypeResponse	87
3.1.4.19	GetListContentTypes	87
3.1.4.19.1	Messages	87
3.1.4.19.1.1	GetListContentTypesSoapIn	88
3.1.4.19.1.2	GetListContentTypesSoapOut	88
3.1.4.19.2	Elements.....	88
3.1.4.19.2.1	GetListContentTypes	88
3.1.4.19.2.2	GetListContentTypesResponse	89
3.1.4.20	GetListContentTypesAndProperties.....	90
3.1.4.20.1	Messages	91
3.1.4.20.1.1	GetListContentTypesAndPropertiesSoapIn	91
3.1.4.20.1.2	GetListContentTypesAndPropertiesSoapOut.....	91
3.1.4.20.2	Elements.....	91
3.1.4.20.2.1	GetListContentTypesAndProperties Element	92
3.1.4.20.2.2	GetListContentTypesAndPropertiesResponse	93
3.1.4.21	GetListItemChanges.....	95
3.1.4.21.1	Messages	95
3.1.4.21.1.1	GetListItemChangesSoapIn	96
3.1.4.21.1.2	GetListItemChangesSoapOut.....	96

3.1.4.21.2	Elements	96
3.1.4.21.2.1	GetListItemChanges	96
3.1.4.21.2.2	GetListItemChangesResponse	97
3.1.4.22	GetListItemChangesSinceToken	100
3.1.4.22.1	Messages	100
3.1.4.22.1.1	GetListItemChangesSinceTokenSoapIn	101
3.1.4.22.1.2	GetListItemChangesSinceTokenSoapOut	101
3.1.4.22.2	Elements	101
3.1.4.22.2.1	GetListItemChangesSinceToken	101
3.1.4.22.2.2	GetListItemChangesSinceTokenResponse	104
3.1.4.23	GetListItemChangesWithKnowledge	106
3.1.4.23.1	Messages	106
3.1.4.23.1.1	GetListItemChangeswithKnowledgeSoapIn	107
3.1.4.23.1.2	GetListItemChangeswithKnowledgeSoapOut	107
3.1.4.23.2	Elements	107
3.1.4.23.2.1	GetListItemChangesWithKnowledge	107
3.1.4.23.2.2	GetListItemChangesWithKnowledgeResponse	109
3.1.4.24	GetListItems	112
3.1.4.24.1	Messages	113
3.1.4.24.1.1	GetListItemsSoapIn	113
3.1.4.24.1.2	GetListItemsSoapOut	113
3.1.4.24.2	Elements	113
3.1.4.24.2.1	GetListItems	114
3.1.4.24.2.2	GetListItemsResponse	115
3.1.4.25	GetVersionCollection	116
3.1.4.25.1	Messages	117
3.1.4.25.1.1	GetVersionCollectionSoapIn	117
3.1.4.25.1.2	GetVersionCollectionSoapOut	118
3.1.4.25.2	Elements	118
3.1.4.25.2.1	GetVersionCollection	118
3.1.4.25.2.2	GetVersionCollectionResponse	119
3.1.4.26	UndoCheckOut	119
3.1.4.26.1	Messages	120
3.1.4.26.1.1	UndoCheckOutSoapIn	120
3.1.4.26.1.2	UndoCheckOutSoapOut	120
3.1.4.26.2	Elements	120
3.1.4.26.2.1	UndoCheckOut	121
3.1.4.26.2.2	UndoCheckOutResponse	121
3.1.4.27	UpdateContentType	121
3.1.4.27.1	Messages	122
3.1.4.27.1.1	UpdateContentTypeSoapIn	122
3.1.4.27.1.2	UpdateContentTypeSoapOut	123
3.1.4.27.2	Elements	123
3.1.4.27.2.1	UpdateContentType	123
3.1.4.27.2.2	UpdateContentTypeResponse	124
3.1.4.28	UpdateContentTypesXmlDocument	126
3.1.4.28.1	Messages	126
3.1.4.28.1.1	UpdateContentTypesXmlDocumentSoapIn	127
3.1.4.28.1.2	UpdateContentTypesXmlDocumentSoapOut	127
3.1.4.28.2	Elements	127
3.1.4.28.2.1	UpdateContentTypesXmlDocument	127
3.1.4.28.2.2	UpdateContentTypesXmlDocumentResponse	128
3.1.4.29	UpdateContentTypeXmlDocument	129

3.1.4.29.1	Messages	130
3.1.4.29.1.1	UpdateContentTypeXmlDocumentSoapIn	130
3.1.4.29.1.2	UpdateContentTypeXmlDocumentSoapOut	130
3.1.4.29.2	Elements	130
3.1.4.29.2.1	UpdateContentTypeXmlDocument	130
3.1.4.29.2.2	UpdateContentTypeXmlDocumentResponse	131
3.1.4.30	UpdateList	132
3.1.4.30.1	Messages	132
3.1.4.30.1.1	UpdateListSoapIn	132
3.1.4.30.1.2	UpdateListSoapOut	133
3.1.4.30.2	Elements	133
3.1.4.30.2.1	UpdateList	133
3.1.4.30.2.2	UpdateListResponse	135
3.1.4.31	UpdateListItems	136
3.1.4.31.1	Messages	137
3.1.4.31.1.1	UpdateListItemsSoapIn	137
3.1.4.31.1.2	UpdateListItemsSoapOut	137
3.1.4.31.2	Elements	137
3.1.4.31.2.1	UpdateListItems	138
3.1.4.31.2.2	UpdateListItemsResponse	144
3.1.4.32	UpdateListItemsWithKnowledge	145
3.1.4.32.1	Messages	146
3.1.4.32.1.1	UpdateListItemsWithKnowledgeSoapIn	146
3.1.4.32.1.2	UpdateListItemsWithKnowledgeSoapOut	147
3.1.4.32.2	Elements	147
3.1.4.32.2.1	UpdateListItemsWithKnowledge	147
3.1.4.32.2.2	UpdateListItemsWithKnowledgeResponse	148
3.1.5	Timer Events	149
3.1.6	Other Local Events	149
4	Protocol Examples	150
4.1	New Content Type	150
4.2	Create and Update a List	160
5	Security	166
5.1	Security Considerations for Implementers	166
5.2	Index of Security Parameters	166
6	Appendix A: Full WSDL	167
7	Appendix B: Product Behavior	200
8	Change Tracking	205
9	Index	206

1 Introduction

This document specifies the List Web Service Protocol that is used for the manipulation of list schemas and list data.

Sections 1.8, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in RFC 2119. Sections 1.5 and 1.9 are also normative but cannot contain those terms. All other sections and examples in this specification are informative.

1.1 Glossary

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

anonymous user
Coordinated Universal Time (UTC)
GUID
Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)
language code identifier (LCID)

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

ActiveX object
assembly name
attachment
base type
check out
content type
content type group
content type identifier
context site
cookie
data type
data validation
default list view
default mobile list view
default view
discussion board
discussion item
display name
document
document library
document template
e-mail address
empty GUID
event receiver
external data
feature
feature identifier
field
folder
global assembly cache (GAC)
Information Rights Management (IRM)
item

list
list item
list schema
list template identifier
lookup field
major version
meeting instance
minor version
offline
permission
presence
property bag
query
Recycle Bin
root folder
security scope
server-relative URL
Simple Object Access Protocol (SOAP)
site
site collection
site content type
site template
SOAP action
SOAP body
SOAP fault
subsite
thumbnail
time zone
top-level site
Uniform Resource Identifier (URI)
Uniform Resource Locator (URL)
user identifier
user information list
version
view
View
Web application
Web Services Description Language (WSDL)
workflow
XML document
XML namespace
XML namespace prefix
XML Path Language (XPath)
XML schema

The following terms are specific to this document:

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

1.2 References

References to Microsoft Open Specifications documentation do not include a publishing year because links are to the latest version of the documents, which are updated frequently. References to other documents include a publishing year when one is available.

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. Please check the archive site, <http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624>, as an additional source.

[ISO-8601] International Organization for Standardization, "Data Elements and Interchange Formats - Information Interchange - Representation of Dates and Times", ISO/IEC 8601:2004, December 2004, <http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=40874&ICS1=1&ICS2=140&ICS3=30>

Note There is a charge to download the specification.

[MS-LCID] Microsoft Corporation, "[Windows Language Code Identifier \(LCID\) Reference](#)".

[MS-OUTSPS] Microsoft Corporation, "[Lists Client Sync Protocol Specification](#)".

[MS-PRSTFR] Microsoft Corporation, "[ADO XML Persistence Format Protocol Specification](#)".

[MS-VIEWSS] Microsoft Corporation, "[Views Web Service Protocol Specification](#)".

[MS-WSSCAML] Microsoft Corporation, "[Collaborative Application Markup Language \(CAML\) Structure Specification](#)".

[MS-WSSFO2] Microsoft Corporation, "[Windows SharePoint Services \(WSS\): File Operations Database Communications Version 2 Protocol Specification](#)".

[MS-WSSTS] Microsoft Corporation, "[Windows SharePoint Services Technical Specification](#)".

[MS-XSSK] Microsoft Corporation, "[XML Serialization of Synchronization Knowledge Specification](#)".

[RFC1123] Braden, R., "Requirements for Internet Hosts - Application and Support", STD 3, RFC 1123, October 1989, <http://www.ietf.org/rfc/rfc1123.txt>

[RFC2045] Freed, N., and Borenstein, N., "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, November 1996, <http://ietf.org/rfc/rfc2045.txt>

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

[RFC2396] Berners-Lee, T., Fielding, R., and Masinter, L., "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998, <http://www.ietf.org/rfc/rfc2396.txt>

[RFC2616] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, <http://www.ietf.org/rfc/rfc2616.txt>

[RFC2822] Resnick, P., Ed., "Internet Message Format", STD 11, RFC 2822, April 2001, <http://www.ietf.org/rfc/rfc2822.txt>

[RFC4648] Josefsson, S., "The Base16, Base32, and Base64 Data Encodings", RFC 4648, October 2006, <http://www.ietf.org/rfc/rfc4648.txt>

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", May 2000, <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

[SOAP1.2/1] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 1: Messaging Framework", W3C Recommendation, June 2003, <http://www.w3.org/TR/2003/REC-soap12-part1-20030624>

[SOAP1.2/2] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 2: Adjuncts", W3C Recommendation, June 2003, <http://www.w3.org/TR/2003/REC-soap12-part2-20030624>

[WSDL] Christensen, E., Curbera, F., Meredith, G., and Weerawarana, S., "Web Services Description Language (WSDL) 1.1", W3C Note, March 2001, <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>

[XMLNS] Bray, T., Hollander, D., Layman, A., et al., Eds., "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation, December 2009, <http://www.w3.org/TR/2009/REC-xml-names-20091208/>

[XMLSCHEMA1] Thompson, H.S., Ed., Beech, D., Ed., Maloney, M., Ed., and Mendelsohn, N., Ed., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, <http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/>

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

1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "[Windows Protocols Master Glossary](#)".

[MS-OFCGLOS] Microsoft Corporation, "[Microsoft Office Master Glossary](#)".

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000, <http://www.ietf.org/rfc/rfc2818.txt>

1.3 Protocol Overview (Synopsis)

This protocol enables protocol clients to manage lists of data maintained by a protocol server. The protocol allows the client to control both the **list schemas** as well as the data stored within the lists. A typical scenario for using this protocol is a data tracking application that allows users to access data remotely. Such an application could use this protocol to provide users with a way to maintain data on a central protocol server and access it through multiple protocol clients at different locations.

1.3.1 Roles

Two roles are always being played whenever this protocol is used. There is always a protocol client issuing requests to a protocol server, and there is always a protocol server to receive, process, and respond to the requests of protocol clients.

1.3.1.1 Protocol Server

The protocol server implements the Web service described by this protocol. It also maintains lists of data that are manipulated through that Web service.

1.3.1.2 Protocol Clients

Protocol clients issue commands to the protocol server using the Web service methods described in this protocol specification.

1.3.2 Scenarios

The methods described by this service enable several types of operations around lists, their data, and their metadata.

1.3.2.1 List Schema Management

Protocol clients can add, modify, delete, and retrieve information about all lists in a site. A common usage is to update the schema of a particular list, as follows:

1. The protocol client asks the protocol server for a summary of all lists on the protocol server's site.
2. The protocol server responds with a collection of information about lists.
3. The protocol client asks for detailed information about a particular list.
4. The protocol server responds with detailed information about the specified list, which includes the list's schema.
5. The protocol client sends the desired updates to the list's schema to the protocol server.
6. The protocol server responds with the updated schema.

The following diagram shows this exchange.

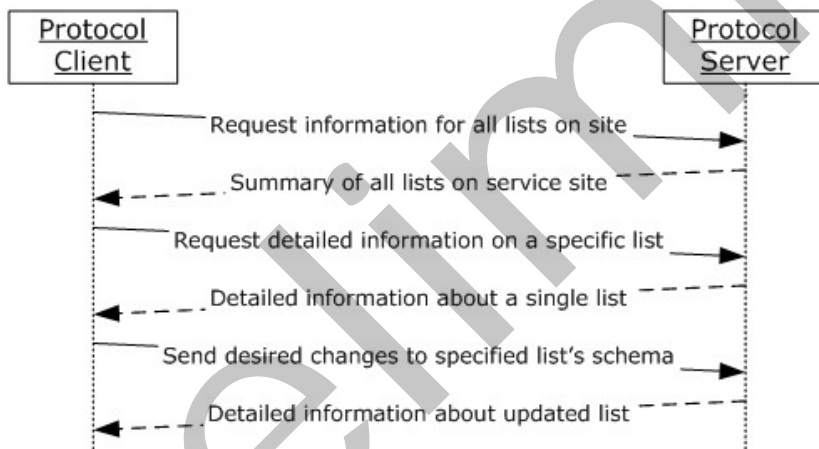


Figure 1: Path of information about list schemas

1.3.2.2 Content Type Management

Protocol clients can add, modify, delete, and retrieve information about content type metadata for a given list. A common usage is to remove an applied content type from a particular list, as follows:

1. The protocol client requests information about all the content types applied to a particular list.

2. The protocol server responds with a collection containing summary information about all content types on the specified list.
3. The protocol client requests the deletion of the content.
4. The protocol server confirms the deletion.

The following diagram shows this exchange.

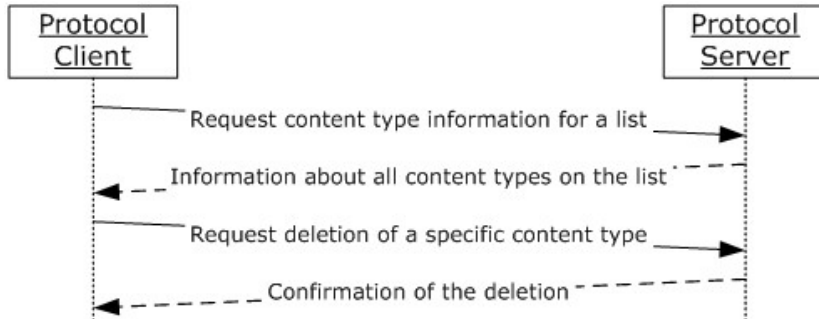


Figure 2: Path of information about content type metadata

1.3.2.3 List Data Management

Protocol clients can add, modify, delete, and retrieve information about all data stored within a list. List data can be further broken down into three types: list items, attachments, and documents.

1.3.2.3.1 List Items

List items refer to the records within most types of lists. A common usage of this protocol is to update data in an existing list. This can be accomplished through two requests to the protocol server, as follows:

1. The protocol client requests the current values of the list items.
2. The protocol server responds with the current data for the requested list items.
3. The protocol client sends the desired edits to the values of one or more list items.
4. The protocol server responds with updated list item values that show the desired edits applied to the values.

The following diagram shows this exchange.

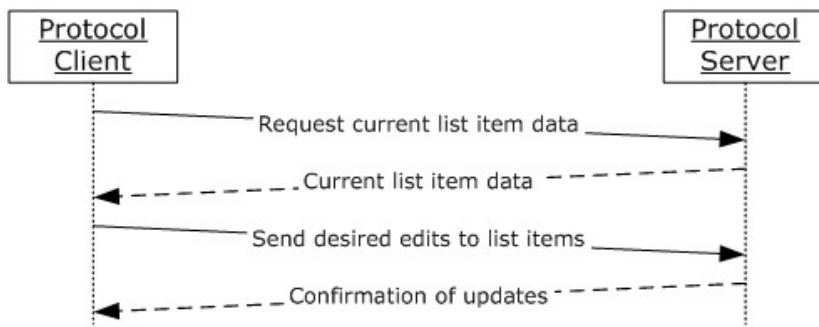


Figure 3: Path of information about list items

1.3.2.3.2 Attachments

List item attachments refer to the collection of files that can be associated with any list item. A common usage is to check for the presence of a particular attachment on a particular list item, and then add it if it is not already present, as follows:

1. The protocol client requests information about the attachments associated with a particular list item.
2. The protocol server responds with a collection of items that contains summary information about all the attachments associated with the specified list item.
3. Because the attachment on the protocol client is not already associated with the list item, the protocol client adds the attachment to the list item.
4. The protocol server responds with confirmation that the attachment was added successfully.

The following diagram shows this exchange.

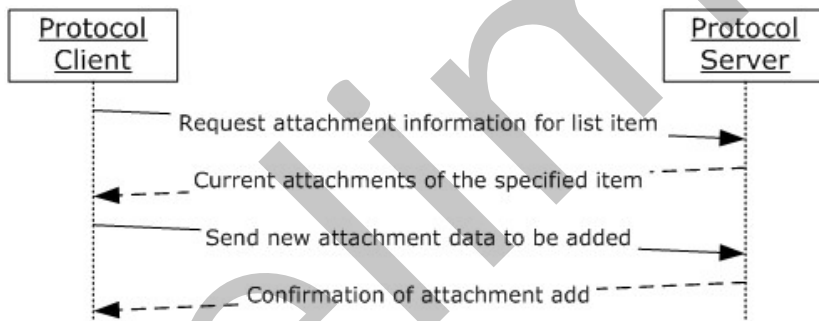


Figure 4: Path of information about list item attachments

1.3.2.3.3 Documents

Documents refer to the type of items stored in specific types of lists known as document libraries. A common usage is to check out a file from a **document library**, update it, and then check it back in, as follows:

1. The protocol client checks out the desired file.

2. The protocol server responds with confirmation that the file was checked out successfully.
3. The protocol client makes updates to the file and then checks in the file.
4. The protocol server responds with confirmation that the file was checked in successfully.

The following diagram shows this exchange.

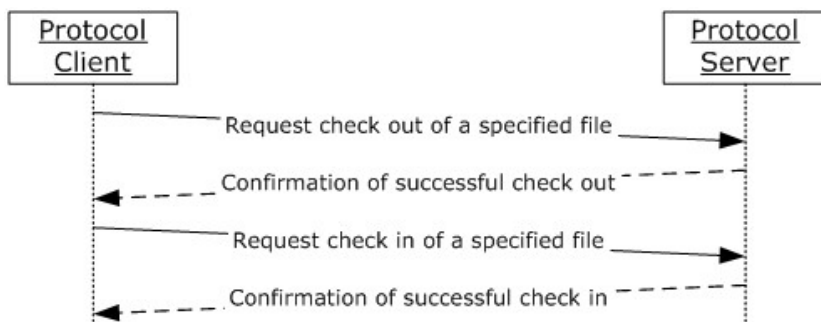


Figure 5: Path of information about documents

1.4 Relationship to Other Protocols

This protocol uses the **SOAP** message protocol for formatting request and response messages, as described in [\[SOAP1.1\]](#), [\[SOAP1.2/1\]](#) and [\[SOAP1.2/2\]](#). It transmits those messages by using **HTTP**, as described in [\[RFC2616\]](#), or **Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)**, as described in [\[RFC2818\]](#).

The following diagram shows the underlying messaging and transport stack used by the protocol:

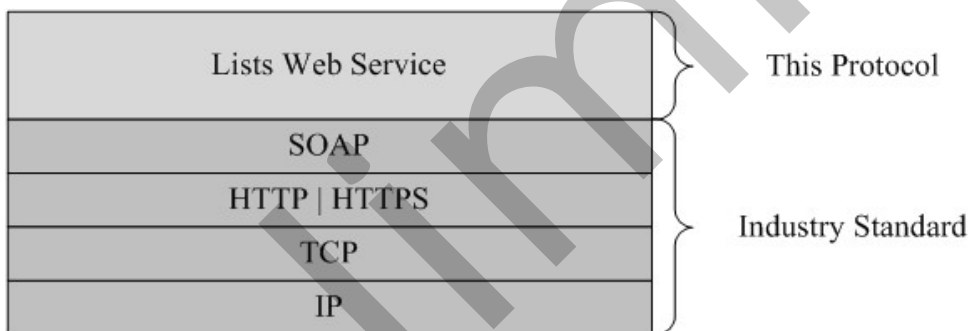


Figure 6: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a **site** that is identified by a **URL** that is known by protocol clients. The protocol server endpoint is formed by appending "_vti_bin/lists.asmx" to the URL of the site, for example http://www.contoso.com/Repository/_vti_bin/lists.asmx.

1.6 Applicability Statement

This protocol can be used for the transport and storage of any data.

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- **Supported Transports:** This protocol uses multiple transports with SOAP as described in section [2.1](#).

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

Protocol servers **MUST** support SOAP over HTTP. Protocol servers **SHOULD** additionally support SOAP over HTTPS for securing communication with protocol clients.

Protocol messages **MUST** be formatted as specified either in [\[SOAP1.1\]](#), "SOAP Envelope", or in [\[SOAP1.2/1\]](#), "SOAP Message Construct". Protocol server faults **MUST** be returned either by using HTTP Status Codes as specified in [\[RFC2616\]](#), section 10, "Status Code Definitions", or by using **SOAP faults** as specified either in [\[SOAP1.1\]](#), section 4.4, "SOAP Fault", or in [\[SOAP1.2/1\]](#), section 5.4, "SOAP Fault".

2.2 Common Message Syntax

This section contains common definitions that are used by this protocol. The syntax of the definitions uses **XML schema**, as specified in [\[XMLSCHEMA1\]](#) and [\[XMLSCHEMA2\]](#), and **WSDL**, as specified in [\[WSDL\]](#).

2.2.1 Namespaces

This protocol specifies and references **XML namespaces** by using the mechanisms specified in [\[XMLNS\]](#). Although this document associates an **XML namespace prefix** for each XML namespace that is used, the choice of any particular XML namespace prefix is implementation-specific and not significant for interoperability.

Prefix	Namespace URI	Reference
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
tns	http://schemas.microsoft.com/sharepoint/soap/	
s	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1]
soap12	http://schemas.xmlsoap.org/wsdl/soap12/	[SOAP1.2/1] [SOAP1.2/2]
(none)	http://schemas.microsoft.com/sharepoint/soap/	
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
mime	http://schemas.xmlsoap.org/wsdl/mime/	
http	http://schemas.xmlsoap.org/wsdl/http/	
rs	urn:schemas-microsoft-com:rowset	[MS-PRSTFR]
tm	http://microsoft.com/wsdl/mime/textMatching/	
soapenc	http://schemas.xmlsoap.org/soap/encoding/	
core	http://schemas.microsoft.com/sharepoint/soap/	[MS-WSSFO2] [MS-WSSCAML]
sync	http://schemas.microsoft.com/2008/03/sync/	[MS-XSSK]

2.2.2 Messages

This specification does not define any common WSDL message definitions.

2.2.3 Elements

This specification does not define any common XML schema element definitions.

2.2.4 Complex Types

The following table summarizes the set of common XML schema complex type definitions defined by this specification. XML schema complex type definitions that are specific to a particular operation are described with the operation.

Complex type	Description
AddOrUpdateFieldDefinition	Specifies data on a field to be added or updated on a content type .
AddOrUpdateFieldsDefinition	Specifies data on operations to add or update fields on a content type.
CamlContains	Specifies a filter based on a single field to reduce the data returned from a data retrieval request.
CamlQueryOptions	Specifies parameters to alter the data and format of the data returned from GetListItems , GetListItemChangesSinceToken and GetListItemChangesWithKnowledge .
CamlViewFields	Specifies which additional fields data SHOULD be returned from a data retrieval request.
ContentTypePropertyDefinition	Specifies properties to be set on the content type.
DataDefinition	Specifies items contained within a list.
DeleteFieldDefinition	Specifies data on a field to be removed from a content type.
DeleteFieldsDefinition	Specifies data on operations to remove fields from a content type.
FieldReferenceDefinitionCT	Specifies data on a field included in a content type.
ListDefinitionCT	Specifies information about a particular list .
ListDefinitionSchema	Specifies the results from retrieving a specified list.
ListItemChangeDefinition	Specifies a change to a list item.
UpdateListFieldResults	Specifies the results from an Add, Update, or Delete operation on a list's fields.
UpdateListFieldRequest	Specifies the field information to Add, Update, or Delete operation of fields of a list.
ViewChangeDefinition	Specifies a change to a list view.

2.2.4.1 AddOrUpdateFieldDefinition

Specifies data on a field to be added or updated on a content type.

```

<s:complexType name="AddOrUpdateFieldDefinition">
  <s:attribute name="Aggregation" type="s:string" />
  <s:attribute name="DisplayName" type="s:string" />
  <s:attribute name="Hidden" type="core:TRUE_Case_Sensitive_Else_Anything" />
  <s:attribute name="ID" type="core:UniqueIdentifierWithOrWithoutBraces" />
  <s:attribute name="Name" type="s:string" />
  <s:attribute name="Node" type="s:string" />
  <s:attribute name="PIAttribute" type="s:string" />
  <s:attribute name="PITarget" type="s:string" />
  <s:attribute name="PrimaryPIAttribute" type="s:string" />
  <s:attribute name="PrimaryPITarget" type="s:string" />
  <s:attribute name="ReadOnly" type="core:TRUE_Case_Sensitive_Else_Anything" />
  <s:anyAttribute namespace="##other" processContents="lax" />
</s:complexType>

```

AddOrUpdateFieldDefinition.Aggregation: For fields with the **Node** or **node** attribute as specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2, a protocol server MUST use this attribute to control promotion from XML or **site template** files.

AddOrUpdateFieldDefinition.DisplayName: Specifies the text to be displayed in the user interface when referring to the field.

AddOrUpdateFieldDefinition.Hidden: Specifies that a field SHOULD not be rendered in views or forms. "TRUE" MUST correspond to **true**, and all other values to false.

AddOrUpdateFieldDefinition.ID: Specifies a **GUID** for the field.

AddOrUpdateFieldDefinition.Name: Specifies a string that identifies the field within its list. Used to find the field to be added or updated if the protocol server cannot find a field for the value specified by the **ID** attribute of the **AddOrUpdateFieldDefinition** element, or if the **ID** attribute is not specified.

AddOrUpdateFieldDefinition.Node: When present, specifies an **XPath** to be used to read or write the value of the field into an XML document.

AddOrUpdateFieldDefinition.PIAttribute: As specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2.

AddOrUpdateFieldDefinition.PITarget: As specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2.

AddOrUpdateFieldDefinition.PrimaryPIAttribute: As specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2.

AddOrUpdateFieldDefinition.PrimaryPITarget: As specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2.

AddOrUpdateFieldDefinition.ReadOnly: Specifies whether the user is allowed to change the field through the user interface. If true, only programmatic changes are allowed. "TRUE" MUST correspond to **true**, and all other values to **false**.

2.2.4.2 AddOrUpdateFieldsDefinition

Specifies data on operations to add or update fields on a content type.

```

<s:complexType name="AddOrUpdateFieldsDefinition">
  <s:sequence>
    <s:element name="Fields">

```

```

<s:complexType>
  <s:sequence>
    <s:element name="Method" minOccurs="0" maxOccurs="unbounded">
      <s:complexType>
        <s:sequence>
          <s:element name="Field" type="tns:AddOrUpdateFieldDefinition" />
        </s:sequence>
        <s:attribute name="ID" type="s:string" use="required" />
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>

```

Fields: Specifies a set of field add or update operations.

Method: Specifies data for the add or update operation.

Method.ID: Identifies the add or update operation.

Field: Specifies data on a field to be added or updated on a content type.

2.2.4.3 CamlContains

Specifies a filter based on a single field to restrict the data returned from a data-retrieval request.

```

<s:complexType name="CamlContains">
  <s:sequence>
    <s:element name="Contains">
      <s:complexType>
        <s:sequence>
          <s:element name="FieldRef" >
            <s:complexType>
              <s:attribute name="Name" type="s:string" />
            </s:complexType>
          </s:element>
          <s:element name="Value" >
            <s:complexType mixed="true">
              <s:attribute name="Type" type="core:FieldInternalType" />
            </s:complexType>
          </s:element>
        </s:sequence>
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>

```

FieldRef: Specifies which field to apply the filter to. The **Name** attribute MUST be set to the **Name** attribute of the field as specified in the response to a **GetList** request.

Value: Specifies the **data type** of the field to which to apply the filter, and its **Type** attribute matches the field type specified in the response from a **GetList** request. The filter to apply is a simple string that can be converted into the field type by the protocol server. For example, filtering the identification field for 200 would be specified as follows:

```

<Contains>
  <FieldRef Name="ID">
    <Value Type="Counter">200</Value>
  </FieldRef>
</Contains>

```

In a different example, filtering a text field named TextCol for the text "FindMe" would be specified as follows:

```

<Contains>
  <FieldRef Name="TextCol">
    <Value Type="Text">FindMe</Value>
  </FieldRef>
</Contains>

```

2.2.4.4 CamlQueryOptions

Specifies parameters to alter the data and format of the data returned from **GetListItems**, **GetListItemChangesSinceToken** and **GetListItemChangesWithKnowledge**.

```

<s:complexType name="CamlQueryOptions">
  <s:sequence>
    <s:element name="QueryOptions" minOccurs="0" >
      <s:complexType>
        <s:all>
          <s:element name="DateInUtc" type="core:TRUEFALSE" minOccurs="0" />
          <s:element name="Folder" type="s:string" minOccurs="0" />
          <s:element name="Paging" minOccurs="0">
            <s:complexType>
              <s:attribute name="ListItemCollectionPositionNext"
                type="s:string" />
            </s:complexType>
          </s:element>
          <s:element name="IncludeMandatoryColumns"
            type="core:TRUEFALSE" minOccurs="0" />
          <s:element name="MeetingInstanceID"
            type="s:integer" minOccurs="0" />
          <s:element name="ViewAttributes" minOccurs="0" >
            <s:complexType>
              <s:attribute name="Scope" type="tns:EnumViewAttributes"/>
            </s:complexType>
          </s:element>
          <s:element name="RecurrencePatternXMLVersion"
            type="tns:EnumRecurrencePatternXMLVersion"
            minOccurs="0" />
          <s:element name="IncludePermissions" type="core:TRUEFALSE"
            minOccurs="0" />
          <s:element name="ExpandUserField"
            type="core:TRUEFALSE" minOccurs="0" />
          <s:element name="RecurrenceOrderBy"
            type="core:TRUEFALSE" minOccurs="0"/>
          <s:element name="IncludeAttachmentUrls"
            type="core:TRUEFALSE" minOccurs="0" />
          <s:element name="RemoveInvalidXmlCharacters"
            type="core:TRUEFALSE" minOccurs="0" />
          <s:element name="OptimizeFor" type="tns:EnumOptimizeFor" minOccurs="0" />
        </s:all>
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>

```

```

    <s:element name="ExtraIds" type="s:string" minOccurs="0" />
    <s:element name="OptimizeLookups" type="core:TRUEFALSE" minOccurs="0" />
    <s:element name="IncludeFragmentChanges" type="core:TRUEFALSE" minOccurs="0" />
  </s:all>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>

```

The elements of **CAMLQueryOptions** are specified in the following table.

Element name	Description
DateInUtc	<p>Specifies the format in which dates are returned. If True, dates returned are in Coordinated Universal Time (UTC) format. If False, dates returned are in [ISO-8601] format. This element is optional, and its default value is False.</p> <p>An example of setting this option is as follows:</p> <pre><DatesInUtc>True</DatesInUtc></pre> <p>An example of a UTC date is: 2006-10-04T10:00:00Z. Non-UTC dates would be in the local (protocol server) time zone. For a time zone that is 8-hours behind Greenwich Mean Time, the UTC date: 2006-10-04T10:00:00Z in non-UTC would be 2006-10-04 02:00:00.</p>
Folder	<p>Specifies a protocol server-relative URL used to filter document library items for the specified folder. This element is optional, and its default value is an empty string.</p> <p>An example of setting this option is as follows:</p> <pre><Folder>Shared Documents/foldername</Folder></pre> <p>This example retrieves only list items from the <i>foldername</i> folder.</p>
Paging	<p>Specifies an element with ListItemCollectionPositionNext attribute. If this element is left empty or excluded, the list items returned start from the beginning of the list. Protocol clients can supply the value of a protocol server-supplied bookmark, which is opaque to the protocol client: these MUST be obtained from a previous call to GetListItems. The protocol server MUST return the next page of data based on the bookmark supplied. Protocol clients recognize that all the data has been retrieved from the list when the count of elements retrieved is less than the rowLimit specified.</p> <p>An example snippet from the response to a GetListItems call is as follows.</p> <pre><GetListItemsResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/"><GetListItemsResult>< listitems xmlns:s='uuid:BDC6E3F0-6DA3-11d1-A2A3-00AA00C14882' xmlns:dt='uuid:C2F41010-65B3-11d1-A29F-00AA00C14882' xmlns:rs='urn:schemas-microsoft-com:rowset' xmlns:z='#RowsetSchema'> <rs:data itemCount="1" ListItemCollectionPositionNext="Paged=TRUE&amp;p_FSObjType=1&amp;p_ID=11">... </GetListItemsResponse></pre> <p>The ListItemCollectionPositionNext value is extracted from this response and then passed back to the protocol server to request the next page by including queryOptions with the following Paging element specified:</p> <pre><Paging ListItemCollectionPositionNext="Paged=TRUE&amp;p_FSObjType=1&amp;p_ID=11" /></pre>

Element name	Description
IncludeMandatoryColumns	<p>Specifies that required fields and fields used by specified calculated fields be returned in addition to the fields specified by the viewFields parameter if set to True. This element is optional, and its default value is True.</p> <p>It is important to note that there MAY be additional fields that are returned by the protocol server independent of this option.</p>
MeetingInstanceID	<p>Specifies the meeting instances to return. Each occurrence within a meeting workspace with a recurring event and single events that are linked have a positive integer value. Negative integer values have the following meaning: -3 = UnSpecified, -2 = AllWithSeries, -1 = AllButSeries, 0 = Series. This element is optional, and its default value is -3, unless the request does not support cookies, in which case the default is -1.</p> <p>AllButSeries Indicates all meeting occurrences that have been activated in the workspace, excluding the meeting entry representing the meeting series.</p> <p>AllWithSeries Indicates all meeting occurrences that have been activated in the workspace, including the meeting entry representing the meeting series, provided a series is linked to the workspace.</p> <p>Series Indicates the meeting entry corresponding to the meeting series, provided a series is linked to the workspace.</p> <p>UnSpecified Indicates the meeting instance is unspecified. The first upcoming meeting occurrence is used in this case.</p>
ViewAttributes	<p>When this element is supplied, it overrides any view attributes that can be retrieved from the persisted view specified by the viewName parameter. The default value is an empty string.</p>
RecurrencePatternXMLVersion	<p>Specifies the version supported for recurrence patterns. The default is an empty string or NULL.</p>
RecurrenceOrderBy	<p>If set to True, this element specifies that for each recurring series, the master item is returned first and then all of the exceptions to this recurrence follow. This is a special internal ordering that is applied ahead of any other ordering and if the list or view has a field of type Recurrence, the list is ordered by fields of reference type UID, EventType and StartDate in the definition of the recurrence field. The default value is False.</p>
IncludePermissions	<p>If it is True, the EffectivePermMask is returned when the permissions of a given row differ from the base permissions of the list. See [MS-WSSFO2] section 2.2.2.14 for more details about permissions. The default value is False.</p>
ExpandUserField	<p>If set to True, specifies that fields in list items that are lookup fields to the user information list are returned as if they were multi-value lookups, including "Name", "EMail", "SipAddress", and "Title" fields from the user information List for the looked-up item. These values are separated by ";" and any commas in the lookup field name are encoded as ",".</p> <p>The default for this element is False.</p> <p>Passing <code><ExpandUserField>False</ExpandUserField></code> in the query options causes the "Author" field to be returned in the following format:</p> <pre>ows_Author="1;#Admin AdminName"</pre> <p>Passing <code><ExpandUserField>True</ExpandUserField></code> returns the preceding result in the</p>

Element name	Description
	<p>following different format:</p> <pre>ows_Author="1;#Admin AdminName,#login\name,#email@address,#sip@address,#Admin AdminName</pre>
IncludeAttachmentUrls	<p>If set to True, specifies that the value returned for the Attachments field is a list of full URLs separated by the delimiter ";#". The default value is False.</p> <p>If this option is True, then all list items returned that have no attachments show the Attachments field as follows, where ... specifies that attributes have been omitted:</p> <pre><z:row ... ows_Attachments='0' ... /></pre> <p>If there are attachments on this list item, then the Attachment field is returned such as follows, where ... specifies that attributes have been omitted:</p> <pre><z:row ... ows_Attachments=';#http://siteservice/Lists/TestListItem/Attachments/249/Attachment1.txt;#http://siteservice/Lists/TestListItem/Attachments/249/Attachment2.txt;#' ... /></pre>
IncludeAttachmentVersion	<p>This MUST be used only in conjunction with IncludeAttachmentUrls. Specifying True causes the GUID and version number to be returned for attachments, which can be used for conflict detection on update. The default value is False.</p> <p>An example snippet from the protocol server response for a list item with two attachments is as follows, where ... specifies that attributes have been omitted:</p> <pre><z:row ... ows_Attachments=';#http://siteservice/Lists/TestListItem/Attachments/249/Attachment1.txt;#{72c714ab-32b2-46e0-8a49-8c9d2ee6ee8d},1;#http://siteservice/Lists/TestListItem/Attachments/249/Attachment2.txt;#{f196ee67-b9e8-475b-9308-a6e941052321},1;#' ... /></pre>
RemoveInvalidXmlCharacters	<p>If set to True, the server MUST remove all characters that have an ASCII value in the range 0-31 other than 9, 10 or 13 in the field values of the list items returned.</p>
OptimizeFor	<p>Specifies how the internal query SHOULD be optimized: list items or list folders. The default is to optimize for list items.</p>
ExtraIds	<p>Specifies extra list items to be included in the returned set regardless of whether they changed or not. The common use of ExtraIds is to specify the IDs of the folders being synchronized. This MUST only be used with a change token.</p>
OptimizeLookups	<p>Specifies how to return lookup values when a list item that is the target of a lookup is in an incomplete or draft state. When False, the data returned will include the current lookup field values. When True, the query returns published lookup field values, when present, regardless of whether the list item that is the target of the lookup is in an incomplete or draft state. The default value is False.</p>
IncludeFragmentChanges	<p>Specifies whether to include file fragment changes in the returned data. When True, file fragment changes that satisfy the criteria specified in the query will be included in the result set. When False, file fragment changes will not be included in the result set. The default value is False.</p>

2.2.4.5 CamlViewFields

Specifies which additional fields SHOULD be returned from a data retrieval request in addition to the fields the protocol server already returns.

```
<s:complexType name="CamlViewFields">
  <s:sequence>
    <s:element name="ViewFields" minOccurs="0" >
      <s:complexType mixed="true" >
        <s:sequence>
          <s:element name="FieldRef" minOccurs="0" maxOccurs="unbounded">
            <s:complexType>
              <s:attribute name="Name" type="s:string" />
            </s:complexType>
          </s:element>
        </s:sequence>
        <s:attribute name="Properties" type="core:TRUEFALSE" />
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>
```

ViewFields: Specifies the fields of the list item that are requested by the client. An empty element `<ViewFields />` specifies that all fields of the list item MUST be returned in the protocol server response.

FieldRef: Specifies the field to be included in the response.

FieldRef .Name: Can match the **Name** attribute of the field as specified in the response from a **GetList** request, but if it matches no fields, it MUST be ignored by the protocol server.

ViewFields.Properties: If **True** and the **MetaInfo** field is referenced in a **FieldRef** element, then metadata properties related to the list item are returned in the protocol server response with the prefix "ows_MetaInfo_".

2.2.4.6 ContentTypePropertyDefinition

Specifies properties to be set on the content type.

```
<s:complexType name="ContentTypePropertyDefinition">
  <s:sequence />
  <s:attribute name="Description" type="s:string" use="optional" />
  <s:attribute name="DocumentTemplate" type="s:string" use="optional" />
  <s:attribute name="Group" type="s:string" use="optional" />
  <s:attribute name="Title" type="s:string" use="optional" />
  <s:attribute name="NewDocumentControl" type="s:string" use="optional" />
  <s:attribute name="RequireClientRenderingOnNew"
    type="s:string" use="optional" />
</s:complexType>
```

ContentTypePropertyDefinition.Description: The description of the content type.

ContentTypePropertyDefinition.DocumentTemplate: Specifies the content type's **document template**.

ContentTypePropertyDefinition.Group: The name of the **content type group** in which this content type is contained.

ContentTypePropertyDefinition.NewDocumentControl: Identifies the **ActiveX object** used for creating a new item.

ContentTypePropertyDefinition.RequireClientRenderingOnNew: Specifies whether the browser uses the ActiveX Object for creating a new item. "TRUE" MUST correspond to **True**, and all other values to **False**.

ContentTypePropertyDefinition.Title: The name of the content type.

2.2.4.7 DataDefinition

Specifies items contained within a list.

```
<s:complexType name="DataDefinition" mixed="true">
  <s:sequence>
    <s:any />
  </s:sequence>
</s:complexType>
```

The **DataDefinition** element contains a required **ItemCount** attribute and an optional **ListItemCollectionPositionNext** attribute. The **ItemCount** attribute is an unsigned 32-bit integer that specifies the number of list items that are included in the response.

ListItemCollectionPositionNext is used by protocol server methods that support paging of results and it is an opaque string returned by the protocol server that allows the protocol client to pass in a subsequent call to get the next page of data.

The **DataDefinition** element further contains a number of **z:row** elements equal to the value of the **ItemCount** attribute. **z** is equal to #RowsetSchema in the Microsoft ADO 2.6 Persistence format (as specified in [\[MS-PRSTFR\]](#)). Each **z:row** element describes a single list item. The names of the attributes of this element correspond to the names of fields in the list.

2.2.4.8 DeleteFieldDefinition

Specifies data on a field to be removed from a content type.

```
<s:complexType name="DeleteFieldDefinition">
  <s:attribute name="ID" type="core:UniqueIdentifierWithOrWithoutBraces" />
  <s:attribute name="Name" type="s:string" />
  <s:anyAttribute namespace="##other" processContents="lax" />
</s:complexType>
```

DeleteFieldDefinition.ID: Specifies a GUID for the field.

DeleteFieldDefinition.Name: Specifies a string that identifies the field within its list. Used to find the field to be removed if the protocol server cannot find a field for the value specified by the **ID** attribute of the **DeleteFieldDefinition** element, or if the **ID** attribute is not specified.

2.2.4.9 DeleteFieldsDefinition

Specifies data on operations to remove fields from a content type.

```

<s:complexType name="DeleteFieldsDefinition">
  <s:sequence>
    <s:element name="Fields" minOccurs="0">
      <s:complexType>
        <s:sequence>
          <s:element name="Method" minOccurs="0" maxOccurs="unbounded">
            <s:complexType>
              <s:sequence>
                <s:element name="Field" type="tns>DeleteFieldDefinition" minOccurs="0" />
              </s:sequence>
              <s:attribute name="ID" type="s:string" use="required" />
            </s:complexType>
          </s:element>
        </s:sequence>
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>

```

Fields: Specifies a set of field remove operations.

Method: Specifies data for the remove operation.

Method.ID: Identifies the field remove operation.

Field: Specifies data on a field to be removed from a content type.

2.2.4.10 FieldReferenceDefinitionCT

Specifies data on a field included in a content type.

```

<s:complexType name="FieldReferenceDefinitionCT">
  <s:attribute name="Aggregation" type="s:string"/>
  <s:attribute name="Customization" type="s:string"/>
  <s:attribute name="DisplayName" type="s:string"/>
  <s:attribute name="Format" type="s:string"/>
  <s:attribute name="Hidden" type="core:TRUEFALSE"/>
  <s:attribute name="ID" type="core:UniqueIdentifierWithOrWithoutBraces"
    use="required"/>
  <s:attribute name="Name" type="s:string" use="required"/>
  <s:attribute name="Node" type="s:string"/>
  <s:attribute name="PIAttribute" type="s:string"/>
  <s:attribute name="PITarget" type="s:string"/>
  <s:attribute name="PrimaryPIAttribute" type="s:string"/>
  <s:attribute name="PrimaryPITarget" type="s:string"/>
  <s:attribute name="ReadOnly" type="core:TRUEFALSE"/>
  <s:attribute name="Required" type="core:TRUEFALSE"/>
  <s:attribute name="ShowInEditForm" type="core:TRUEFALSE"/>
  <s:attribute name="ShowInNewForm" type="core:TRUEFALSE"/>
</s:complexType>

```

FieldReferenceDefinitionCT.Aggregation: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.Customization: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.DisplayName: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.Format: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.Hidden: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.ID: The identifier specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for the referenced field.

FieldReferenceDefinitionCT.Name: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.Node: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.PIAttribute: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.PITarget: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.PrimaryPIAttribute: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.PrimaryPITarget: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.ReadOnly: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.Required: Overrides the field value specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

FieldReferenceDefinitionCT.ShowInNewForm: Overrides the field value described in [\[MS-WSSFO2\]](#) section 2.2.8.3.2.2 for this reference to the field.

FieldReferenceDefinitionCT.ShowInEditForm: Overrides the field value described in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.2 for this reference to the field.

2.2.4.11 ListDefinitionCT

Specifies information about a particular list.

```
<s:complexType name="ListDefinitionCT">
  <s:attribute name="DocTemplateUrl" type="s:string" />
  <s:attribute name="DefaultViewUrl" type="s:string" use="required" />
  <s:attribute name="MobileDefaultViewUrl" type="s:string" />
  <s:attribute name="ID" type="s:string" use="required" />
  <s:attribute name="Title" type="s:string" use="required" />
  <s:attribute name="Description" type="s:string" />
  <s:attribute name="ImageUrl" type="s:string" use="required" />
  <s:attribute name="Name" type="s:string" use="required" />
  <s:attribute name="BaseType" type="s:string" use="required" />
  <s:attribute name="FeatureId" type="s:string" use="required" />
</s:complexType>
```

```
<s:attribute name="ServerTemplate" type="s:string" use="required" />
<s:attribute name="Created" type="s:string" use="required" />
<s:attribute name="Modified" type="s:string" use="required" />
<s:attribute name="LastDeleted" type="s:string" />
<s:attribute name="Version" type="s:int" use="required" />
<s:attribute name="Direction" type="s:string" use="required" />
<s:attribute name="ThumbnailSize" type="s:int" />
<s:attribute name="WebImageWidth" type="s:int" />
<s:attribute name="WebImageHeight" type="s:int" />
<s:attribute name="Flags" type="s:int" />
<s:attribute name="ItemCount" type="s:int" use="required" />
<s:attribute name="AnonymousPermMask" type="s:int64" />
<s:attribute name="RootFolder" type="s:string" />
<s:attribute name="ReadSecurity" type="s:int" />
<s:attribute name="WriteSecurity" type="s:int" />
<s:attribute name="Author" type="s:string" />
<s:attribute name="EventSinkAssembly" type="s:string" />
<s:attribute name="EventSinkClass" type="s:string" />
<s:attribute name="EventSinkData" type="s:string" />
<s:attribute name="EmailInsertsFolder" type="s:string" />
<s:attribute name="EmailAlias" type="s:string" />
<s:attribute name="WebFullUrl" type="s:string" />
<s:attribute name="WebId" type="s:string" />
<s:attribute name="SendToLocation" type="s:string" />
<s:attribute name="ScopeId" type="s:string" />
<s:attribute name="MajorVersionLimit" type="s:int" />
<s:attribute name="MajorWithMinorVersionsLimit" type="s:int" />
<s:attribute name="WorkFlowId" type="s:string" />
<s:attribute name="HasUniqueScopes" type="s:boolean" />
<s:attribute name="NoThrottleListOperations" type="s:boolean" />
<s:attribute name="HasRelatedLists" type="s:boolean" />
<s:attribute name="AllowDeletion" type="s:boolean" />
<s:attribute name="AllowMultiResponses" type="s:boolean" />
<s:attribute name="EnableAttachments" type="s:boolean" />
<s:attribute name="EnableModeration" type="s:boolean" />
<s:attribute name="EnableVersioning" type="s:boolean" />
<s:attribute name="HasExternalDataSource" type="s:boolean" />
<s:attribute name="Hidden" type="s:boolean" />
<s:attribute name="MultipleDataList" type="s:boolean" />
<s:attribute name="Ordered" type="s:boolean" />
<s:attribute name="ShowUser" type="s:boolean" />
<s:attribute name="EnablePeopleSelector" type="s:string" />
<s:attribute name="EnableResourceSelector" type="s:string" />
<s:attribute name="EnableMinorVersion" type="s:boolean" />
<s:attribute name="RequireCheckout" type="s:boolean" />
<s:attribute name="ThrottleListOperations" type="s:boolean" />
<s:attribute name="ExcludeFromOfflineClient" type="s:boolean" />
<s:attribute name="EnableFolderCreation" type="s:boolean" />
<s:attribute name="IrmEnabled" type="s:boolean" />
<s:attribute name="IsApplicationList" type="s:boolean" />
<s:attribute name="PreserveEmptyValues" type="s:boolean" />
<s:attribute name="StrictTypeCoercion" type="s:boolean" />
<s:attribute name="EnforceDataValidation" type="s:boolean" use="optional"/>
<s:attribute name="MaxItemsPerThrottledOperation" type="s:int" />
<s:attribute name="EnableAssignedToEmail" type="s:boolean" use="optional"/>
</s:complexType>
```

ListDefinitionCT.DocTemplateUrl: This is the **server-relative URL** to the location of the template document used for creating new items in a document library. This SHOULD only be used by document libraries.

ListDefinitionCT.DefaultViewUrl: Server-relative path to the page containing the **default mobile list view**.

ListDefinitionCT.MobileDefaultViewUrl: Server-relative path to the page containing the default mobile list view.

ListDefinitionCT.ID: The GUID for the list.

ListDefinitionCT.Title: The **display name** of the list.

ListDefinitionCT.Description: The description of the list.

ListDefinitionCT.ImageUrl: The server-relative path to the image that is used to represent this list.

ListDefinitionCT.Name: The GUID for the list.

ListDefinitionCT.BaseType: The **base type** of the list. See [\[MS-WSSFO2\]](#) section 2.2.3.11 for the possible values of the **BaseType**.

ListDefinitionCT.FeatureID: The GUID of the **feature** that contains the list schema for the list. It MUST be an **empty GUID** if the list schema is not contained within a feature.

ListDefinitionCT.ServerTemplate: The value corresponding to the template that the list is based on. See [\[MS-WSSFO2\]](#) section 2.2.3.12 for the possible values.

ListDefinitionCT.Created: Specifies the Coordinated Universal Time (UTC) date and time in the Gregorian calendar when the list was created in the format "yyyyMMdd hh:mm:ss" where "yyyy" represents the year, "MM" represents the month, "dd" represents the day of the month, "hh" represents the hour, "mm" represents the minute, and "ss" represents the second.

ListDefinitionCT.Modified: Specifies the Coordinated Universal Time (UTC) date and time in the Gregorian calendar when the list was last modified in the format "yyyyMMdd hh:mm:ss" where "yyyy" represents the year, "MM" represents the month, "dd" represents the day of the month, "hh" represents the hour, "mm" represents the minute, and "ss" represents the second.

ListDefinitionCT.LastDeleted: Specifies the Coordinated Universal Time (UTC) date and time in the Gregorian calendar when the list last had an element deleted in the format "yyyyMMdd hh:mm:ss" where "yyyy" represents the year, "MM" represents the month, "dd" represents the day of the month, "hh" represents the hour, "mm" represents the minute, and "ss" represents the second.

ListDefinitionCT.Version: The numeric major revision of the list.

ListDefinitionCT.Direction: Specifies the direction that items in the list are laid out in when displayed. MUST be one of the following values.

Value	Meaning
none	No direction is specified.
ltr	Items SHOULD be laid out from left to right.

Value	Meaning
rtl	Items SHOULD be laid out from right to left.

ListDefinitionCT.ThumbnailSize: The size of the **thumbnail** image in pixels. Used by picture libraries. A list with template type NOT equal to 109 MUST return an empty string.

ListDefinitionCT.WebImageWidth: The display width of an image in pixels. Used by picture libraries. A list with template type NOT equal to 109 MUST return an empty string.

ListDefinitionCT.WebImageHeight: The display height of an image in pixels. Used by picture libraries. A list with template type NOT equal to 109 MUST return an empty string.

ListDefinitionCT.Flags: Bit Flags for the list. See [\[MS-WSSFO2\]](#) section 2.2.2.5 for more details about **Flags**.

ListDefinitionCT.ItemCount: The number of items in the list.

ListDefinitionCT.AnonymousPermMask: Bit Flags for the list describing the permissions given to **anonymous users**. See [\[MS-WSSFO2\]](#) section 2.2.2.14 for details about permissions.

ListDefinitionCT.RootFolder: The **root folder** of the list.

ListDefinitionCT.ReadSecurity: The read permission setting for this list MUST be one of the following values.

Value	Description
1	All users have read permission to all list items.
2	Users have read permission only to list items they create.

ListDefinitionCT.WriteSecurity: The write permission setting for this list MUST be one of the following values.

Value	Description
1	All users can modify all list items.
2	Users can only modify list items they create.
4	Users cannot modify any list items.

ListDefinitionCT.Author: The **user identifier** of the user who created the list, which is contained in the user information list.

ListDefinitionCT.EventSinkAssembly: The **assembly name** of a file in the **global assembly cache (GAC)** that provides an event handler for document library events. Protocol clients MUST NOT use this attribute. [<1>](#)

ListDefinitionCT.EventSinkClass: The fully qualified, case-sensitive name of a class that defines an event handler for document library events. Protocol clients MUST NOT use this attribute. [<2>](#)

ListDefinitionCT.EventSinkData: An arbitrary string for use by an event handler for document library events. Protocol clients MUST NOT use this attribute. [<3>](#)

ListDefinitionCT.EmailInsertsFolder: The folder in a document library where documents uploaded by e-mail are inserted.

ListDefinitionCT.EmailAlias: The **e-mail address** used by a document library to receive incoming documents to be uploaded.

ListDefinitionCT.WebFullUrl: The server-relative path to the **subsite** that contains this list.

ListDefinitionCT.WebId: The GUID of the site that this list is associated with.

ListDefinitionCT.SendToLocation: The name and URL of a custom location to send list items from within a document library. These two items are returned as a string with a '|' character in between them.

ListDefinitionCT.ScopeId: The GUID of the site that contains this list, unless this list has been given unique **permissions**. In that case this is the GUID of those permissions.

ListDefinitionCT.MajorVersionLimit: The maximum number of **major versions** allowed for a document in a document library that uses version control with major versions only.

ListDefinitionCT.MajorWithMinorVersionsLimit: The maximum number of major versions that are allowed for a document in a document library that uses version control with both major versions and **minor versions**.

ListDefinitionCT.WorkFlowId: The GUID of a workflow association that is used to manage the Content Approval process for the list. Only available for document libraries that have content management enabled and an associated content approval **workflow**.

ListDefinitionCT.HasUniqueScopes: True if any list item within the list has a **security scope** different from the list itself.

ListDefinitionCT.NoThrottleListOperations: True if the list is excluded from the **query** size threshold.

ListDefinitionCT.HasRelatedLists: True if this list has any related lists.

ListDefinitionCT.AllowDeletion: True if this list MAY be deleted.

ListDefinitionCT.AllowMultiResponses: True if this list is a survey list and users are allowed to respond to it multiple times.

ListDefinitionCT.EnableAttachments: True if attachments MAY be attached to items in this List.

ListDefinitionCT.EnableModeration: True if content approval is enabled for this list.

ListDefinitionCT.EnableVersioning: True if this list is a document library and version control is enabled.

ListDefinitionCT.HasExternalDataSource: True if this list has an **external data** source. <4>

ListDefinitionCT.Hidden: True if this list is hidden.

ListDefinitionCT.MultipleDataList: True if the list is in a meeting workspace site and it contains data for multiple meeting instances within the site.

ListDefinitionCT.Ordered: True if list items can be explicitly re-ordered.

ListDefinitionCT.ShowUser: True if this list is a survey list and user names are included in responses.

ListDefinitionCT.EnablePeopleSelector: True if a UI for adding users is available on the list. <5>

ListDefinitionCT.EnableResourceSelector: True if a UI for adding resources is available on the list. <6>

ListDefinitionCT.EnableMinorVersion: True if this list is a document library and minor versions are enabled.

ListDefinitionCT.RequireCheckout: True if this list is a document library and documents MUST be checked out before they can be modified.

ListDefinitionCT.ThrottleListOperations: False if the list is excluded from the query size threshold.

ListDefinitionCT.ExcludeFromOfflineClient: True if the content of the list is not recommended to be taken **offline**. <7>

ListDefinitionCT.EnableFolderCreation: True if folder creation is enabled on the list. <8>

ListDefinitionCT.IrmEnabled: True if **Information Rights Management (IRM)** is enabled on the list. <9>

ListDefinitionCT.IsApplicationList: True if the list is an application list. <10>

ListDefinitionCT.PreserveEmptyValues: True if Null values are preserved for expression evaluation. False if there are no special requirements on how the protocol server treats Null values. <11>

ListDefinitionCT.StrictTypeCoercion: True if expressions anywhere in the list that include invalid type casts such as casting a date to a string SHOULD return an error instead of attempting to perform the cast. <12>

ListDefinitionCT.MaxItemsPerThrottledOperation: Specifies the value of the query size threshold for the **Web application**. <13>

ListDefinitionCT.EnableAssignedToEmail: MUST be set to True if the list is an issues list and the list is configured to send e-mail when an issue's "Assigned To" field value is changed. This attribute is only present if the current list is an issues list.

ListDefinitionCT.EnforceDataValidation: If TRUE then the protocol server SHOULD enforce the **field** properties that limit the data values that can be stored in that field. <14>

2.2.4.12 ListDefinitionSchema

Specifies the results from retrieving a specified list.

```
<s:complexType name="ListDefinitionSchema">
  <s:sequence>
    <s:extension base="tns:ListDefinitionCT">
      <s:sequence>
        <s:element name="Validation" minOccurs="0">
          <s:complexType>
            <s:attribute name="Message" type="s:string" use="optional" />
          </s:complexType>
        </s:element>
      </s:sequence>
    </s:extension>
  </s:sequence>
</s:complexType>
```

```

    </s:element>
    <s:element name="ValidationDisplayNames" minOccurs="0"
      type="s:string" />
    <s:element name="Fields">
      <s:complexType mixed="true">
        <s:sequence>
          <s:element name="Field"
            type="core:FieldDefinition"
            minOccurs="0" maxOccurs="unbounded" />
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element name="RegionalSettings" >
      <s:complexType mixed="true">
        <s:sequence>
          <s:element name="Language" type="s:string" />
          <s:element name="Locale" type="s:string" />
          <s:element name="AdvanceHijri" type="s:string" />
          <s:element name="CalendarType" type="s:string" />
          <s:element name="Time24" type="s:string" />
          <s:element name="TimeZone" type="s:string" />
          <s:element name="SortOrder" type="s:string" />
          <s:element name="Presence" type="s:string" />
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element name="ServerSettings" >
      <s:complexType mixed="true">
        <s:sequence>
          <s:element name="ServerVersion" type="s:string" />
          <s:element name="RecycleBinEnabled" type="core:TRUEFALSE" />
          <s:element name="ServerRelativeUrl" type="s:string" />
        </s:sequence>
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>

```

ListDefinitionSchema.Validation: An expression as specified by [\[MS-WSSTS\]](#) section 2.9. Each data-reference in the expression MUST correspond to a valid Name attribute of a field (2) in the current list (1). If present, it MUST not be longer than 1024 characters. [<15>](#)

ListDefinitionSchema.Validation.Message: A message used to indicate that **data validation** was not satisfied. If present, it MUST not be longer than 1024 characters. [<16>](#)

ListDefinitionSchema.ValidationDisplayNames: An expression as specified by [\[MS-WSSTS\]](#) section 2.9. Each data-reference in the expression MUST correspond to a valid **DisplayName** attribute of a field (2) in the current list (1). **Constants** MUST be formatted according to the **locale** attribute of the **RegionalSettings** element as specified by section [2.2.4.12](#). **Commas** MUST be replaced with the appropriate separator character according to the **locale** attribute of the **RegionalSettings** element as specified by section [2.2.4.12](#). [<17>](#)

ListDefinitionSchema.FieldDefinition: As specified in [\[MS-WSSFO2\]](#) section 2.2.8.3.3.

ListDefinitionSchema.Language: A valid **language code identifier (LCID)** as defined in [\[MS-LCID\]](#).

ListDefinitionSchema.Locale: A valid language code identifier (LCID) as defined in [MS-LCID].

ListDefinitionSchema.AdvanceHijri: An integer between -2 and 2, based on how many days the Hijri calendar needs to be adjusted.

ListDefinitionSchema.CalendarType: Specifies the type of calendar. See [\[MS-WSSFO2\]](#) section 2.2.3.3 for the different types of calendar supported.

ListDefinitionSchema.Time24: Specifies that the list uses 24 hour time if set to True; otherwise, the list does not use 24 hour time.

ListDefinitionSchema.TimeZone: An unsigned short integer that specifies the identifier of the **time zone** that is stored on the server. The value of the integer represents the number of minutes that the time zone differs from Coordinates Universal Time (UTC).

ListDefinitionSchema.SortOrder: The Regional Settings Collation language code identifier (LCID).

ListDefinitionSchema.Presence: Specifies that **presence** is enabled if set to True; otherwise Presence is not enabled.

ListDefinitionSchema.ServerVersion: The current version of protocol server.

ListDefinitionSchema.RecycleBinEnabled: Specifies that the **Recycle Bin** is enabled if set to True; otherwise, the Recycle Bin is not enabled.

ListDefinitionSchema.ServerRelativeUrl: The protocol server-relative URL of the site to which this list belongs. This value is relative to the URL of the **top-level site** of the **site collection** to which the site belongs.

2.2.4.13 ListItemChangeDefinition

Specifies the change to a list item.

```
<s:complexType name="ListItemChangeDefinition">
  <s:attribute name="ChangeType" type="tns:ChangeTypeEnum" />
  <s:attribute name="AfterListId" type="core:UniqueIdentifierWithOrWithoutBraces" />
  <s:attribute name="AfterItemId" type="s:unsignedInt" />
  <s:attribute name="UniqueId" type="core:UniqueIdentifierWithOrWithoutBraces" />
  <s:attribute name="MetaInfo_vti_clientid" type="s:string" />
  <s:attribute name="ServerChangeUnit" type="s:string" />
</s:complexType>
```

The value of the element specifies the identifier of the item in the list.

ChangeType: Specifies the type of change as specified in section [2.2.5.1](#).

AfterListId: Specifies the identifier of the list into which the list item was moved. This MUST be set only for a change type of **ChangeTypeEnum.MoveAway**.

AfterItemId: Specifies the identifier of the **item** after the move change. This MUST be set only for a change type of **ChangeTypeEnum.MoveAway**.

UniqueId: Specifies the unique identifier of the list item. [<18>](#)

MetaInfo_vti_clientid: Specifies the MetaInfo property **vti_clientid** for the item. This MUST be set only if the metainfo property was set for the item and the value is not empty. <19>

ServerChangeUnit: Specifies the synchronization version, as described in [MS-XSSK] section 1.3, for the change. <20> This attribute is not returned when the element is contained in the **GetListItemChangesSinceTokenResult** element.

2.2.4.14 UpdateListFieldResults

Specifies the results from an **Add**, **Update**, or **Delete** operation on a list's fields.

```
<s:complexType name="UpdateListFieldResults">
  <s:sequence>
    <s:element name="Method" minOccurs="0" maxOccurs="unbounded">
      <s:complexType mixed="true">
        <s:sequence>
          <s:element name="ErrorCode" type="s:string" />
          <s:element name="ErrorText" type="s:string" minOccurs="0" />
          <s:element name="Field" type="core:FieldDefinition" minOccurs="0"/>
        </s:sequence>
        <s:attribute name="ID" type="s:string" />
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>
```

Method.ID: The method identification matching the incoming method identification for which the results are to be displayed.

ErrorCode: The string representation of a hexadecimal number indicating whether the operation succeeded or failed. Success is indicated by 0x00000000 and failure is indicated by any other value.

ErrorString: The string description of error if the operation failed.

Field: As specified in [MS-WSSFO2] section 2.2.8.3.3.

2.2.4.15 UpdateListFieldsRequest

Specifies the field information to **Add**, **Update**, or **Delete** operation of fields of a list.

```
<s:complexType name="UpdateListFieldsRequest" mixed="true">
  <s:sequence>
    <s:element name="Fields" minOccurs="0">
      <s:complexType mixed="true">
        <s:sequence>
          <s:element name="Method" maxOccurs="unbounded">
            <s:complexType mixed="true">
              <s:sequence>
                <s:element name="Field" type="core:FieldDefinition"/>
              </s:sequence>
              <s:attribute name="ID" type="s:string"/>
              <s:attribute name="AddToView" type="s:string"/>
            </s:complexType>
          </s:element>
        </s:sequence>
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>
```

```

        </s:complexType>
    </s:element>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>

```

Fields: Specifies data on operations to be performed on a list.

Method: Specifies data on an operation to be performed on the list.

Method.ID: A unique identifier for the operation that allows the protocol client to match with the identification in the protocol server response.

Method.AddToView: The name of the **view** to which to add the fields. Only used for adding new fields. This is an optional parameter and the protocol server MUST assume the **default view** if the value of the parameter is not a GUID or null.

Field: Specifies the **FieldDefinition** ([\[MS-WSSFO2\]](#) section 2.2.8.3.3) of the field to be operated on.

2.2.4.16 ViewChangeDefinition

Specifies a change to a view.

```

<s:complexType name="ViewChangeDefinition">
  <s:attribute name="ChangeType" type="tns:ChangeTypeEnum" />
  <s:attribute name="Id" type="core:UniqueIdentifierWithOrWithoutBraces" />
</s:complexType>

```

ChangeType: Specifies the type of change as specified in section [2.2.5.1](#)

UniqueId: Specifies the unique identifier of the view.

2.2.5 Simple Types

The following table summarizes the set of common XML schema simple type definitions defined by this specification. XML schema simple type definitions that are specific to a particular operation are described with the operation.

Simple type	Description
ChangeTypeEnum	Specifies the type of changes returned when a protocol client requests changes to list items.
EnumOptimizeFor	Specifies whether the retrieval of the list items SHOULD be optimized for many items or for folders.
EnumRecurrencePatternXMLVersion	Specifies which format of recurrence pattern the protocol client supports so the protocol server can return recurrences in the supported format.
EnumViewAttributes	Specifies values to restrict the data returned from document libraries.

Simple type	Description
MethodCmdEnum	Specifies which operation to perform for a list item.
OnErrorEnum	Specifies how to handle errors that occur during a batched mode operation.
TRUEONLY	Specifies that a particular attribute is restricted to only the value "TRUE".

2.2.5.1 ChangeTypeEnum

Specifies the type of changes returned when a protocol client requests changes to list items.

```
<s:simpleType name="ChangeTypeEnum">
  <s:restriction base="s:string">
    <s:enumeration value="Delete" />
    <s:enumeration value="InvalidToken" />
    <s:enumeration value="Restore" />
    <s:enumeration value="MoveAway" />
    <s:enumeration value="SystemUpdate" />
    <s:enumeration value="Rename" />
  </s:restriction>
</s:simpleType>
```

A change token could be invalid, expired, or subject to other errors such as those following, which is indicated in the response.

Change	Description
<Id ChangeType="InvalidToken" />	Specifies that the changeToken parameter for the GetListItemChangesSinceToken operation is either invalid or expired or the knowledge parameter for the GetListItemChangesWithKnowledge operation is too old and cannot be used by the protocol server to enumerate changes in the list.
<Id ChangeType="Restore" />	Specifies that the list has been restored from a protocol server-specific backup such as a Recycle Bin.

In the prior cases, the protocol client MUST discard any copy of the list items that it has cached. Then the protocol client MUST do a full reconciliation with the protocol server by calling **GetListItemChangesSinceToken** with an empty **changeToken** parameter or **GetListItemChangesWithKnowledge** with a null **knowledge** parameter.

The **Changes** element could include other changes that have been made to a list item or file or folder that is not a list item or view or file fragment. A protocol client SHOULD recognize these and treat them as follows:

Change Type	Description
Delete	Specifies that the item with the specified identification is no longer present. Note that Delete changes are sent even if the list item was filtered out by the query.
MoveAway	Specifies that the item be treated in the same manner as a delete.
SystemUpdate	Specifies that changes have been made on the protocol server such that a protocol client MAY get the updated fields for the item in a subsequent GetListItemChangesSinceToken or GetListItemChangesWithKnowledge call by specifying the returned identification by using the ExtraIds query option for list items. If the change was made to a view, the updated view can be obtained from the server by calling GetListAndView method as specified in section 3.1.4.16.
Rename	Specifies that changes have been made on the protocol server such that a protocol client MAY get the updated fields for the list item in a subsequent GetListItemChangesSinceToken or GetListItemChangesWithKnowledge call by specifying the returned identification by using the ExtraIds query option for list items.
Restore	Specifies that the item with the specified identification and any items beneath it in the hierarchy (for example, items under an item that represents a folder) were restored.

2.2.5.2 EnumOptimizeFor

Specifies whether the retrieval of the list items SHOULD be optimized for many items or for folders.

```
<s:simpleType name="EnumOptimizeFor">
  <s:restriction base="s:string">
    <s:enumeration value="ItemIds" />
    <s:enumeration value="FolderUrls" />
  </s:restriction>
</s:simpleType>
```

ItemIds: Sorts list items by identification in ascending order and optimizes performance for many items in a list as long as a query or recurrence order is not requested.

FolderUrls: Optimizes for a request that is filtered to the flat contents of one or more folders.

2.2.5.3 EnumRecurrencePatternXMLVersion

Specifies which format of recurrence pattern the protocol client supports so the protocol server can return recurrences in the supported format.

```
<s:simpleType name="EnumRecurrencePatternXMLVersion">
  <s:restriction base="s:string">
    <s:enumeration value="v3" />
    <s:enumeration value="" />
  </s:restriction>
</s:simpleType>
```

To turn on the new recurrence patterning, specify the **v3** option. This is used by the protocol server to maintain backward compatibility. Passing up **v3** results in new recurrence patterns being sent back in the new format, while not passing this parameter surrounds all new recurrence patterns being sent back in **V3RecurrencePattern** tags.

Result data without the option set includes the **ows_RecurrenceData** attribute with an encoded outer **V3RecurrencePattern** tag. The following is an example of this:

```
<z:row ... ows_fRecurrence='1'  
ows_RecurrenceData='&lt;V3RecurrencePattern /&gt;'  
ows_TimeZone='13'  
ows_XMLTZone='&lt;timeZoneRule&gt;&lt;standardBias&gt;480&lt;/standardBias&gt;&lt;additionalD  
aylightBias&gt;-60&lt;/additionalDaylightBias&gt;&lt;standardDate&gt;&lt;transitionRule  
month=&#39;11&#39; day=&#39;su&#39; weekdayOfMonth=&#39;first&#39;  
/&gt;&lt;transitionTime&gt;2:0:0&lt;/transitionTime&gt;&lt;/standardDate&gt;&lt;daylightDate&  
gt;&lt;transitionRule month=&#39;3&#39; day=&#39;su&#39; weekdayOfMonth=&#39;second&#39;  
/&gt;&lt;transitionTime&gt;2:0:0&lt;/transitionTime&gt;&lt;/daylightDate&gt;&lt;/timeZoneRule  
&gt;'  
</z:row>
```

Result data with the v3 option set includes the **ows_RecurrenceData** attribute without an outer **V3RecurrencePattern** tag. The following is an example of this:

```
<z:row ows_fRecurrence='1'  
ows_RecurrenceData='&lt;recurrence&gt;&lt;rule&gt;&lt;firstDayOfWeek&gt;su&lt;/firstDayOfWeek  
&gt;&lt;repeat&gt;&lt;daily weekday=&quot;TRUE&quot;  
/&gt;&lt;/repeat&gt;&lt;repeatForever&gt;FALSE&lt;/repeatForever&gt;&lt;/rule&gt;&lt;/recurre  
nce&gt;'  
ows_TimeZone='13'  
ows_XMLTZone='&lt;timeZoneRule&gt;&lt;standardBias&gt;480&lt;/standardBias&gt;&lt;additionalD  
aylightBias&gt;-60&lt;/additionalDaylightBias&gt;&lt;standardDate&gt;&lt;transitionRule  
month=&#39;11&#39; day=&#39;su&#39; weekdayOfMonth=&#39;first&#39;  
/&gt;&lt;transitionTime&gt;2:0:0&lt;/transitionTime&gt;&lt;/standardDate&gt;&lt;daylightDate&  
gt;&lt;transitionRule month=&#39;3&#39; day=&#39;su&#39; weekdayOfMonth=&#39;second&#39;  
/&gt;&lt;transitionTime&gt;2:0:0&lt;/transitionTime&gt;&lt;/daylightDate&gt;&lt;/timeZoneRule  
&gt;'  
</z:row>
```

See [\[MS-OUTSPS\]](#) for details on how a calendaring application uses this information.

2.2.5.4 EnumViewAttributes

Specifies values to restrict the data returned from document libraries.

```
<s:simpleType name="EnumViewAttributes">  
  <s:restriction base="s:string">  
    <s:enumeration value="Recursive" />  
    <s:enumeration value="RecursiveAll" />  
    <s:enumeration value="FilesOnly" />  
    <s:enumeration value="" />  
  </s:restriction>  
</s:simpleType>
```

The values here **MUST** be used by the protocol server to restrict the data returned in document libraries. The **Recursive** option **MUST** return only those items one level deep, the **RecursiveAll** option **MUST** return a flat view of all documents and folders, and the **FilesOnly** option **MUST** return only the documents and not the folders.

2.2.5.5 MethodCmdEnum

Specifies which operation to perform for a list item.

```
<s:simpleType name="MethodCmdEnum">
  <s:restriction base="s:string">
    <s:enumeration value="New" />
    <s:enumeration value="Update" />
    <s:enumeration value="Delete" />
    <s:enumeration value="Moderate" />
    <s:enumeration value="Move" />
  </s:restriction>
</s:simpleType>
```

New: Add data for a new list item.

Update: Update data for an existing list item.

Delete: Delete a list item.

Moderate: Update the moderate status of an existing list item.

Move: Change the location of a list item.

2.2.5.6 OnErrorEnum

Specifies how to handle errors that occur during a batched mode operation.

```
<s:simpleType name="OnErrorEnum">
  <s:restriction base="s:string">
    <s:enumeration value="Return" />
    <s:enumeration value="Continue" />
  </s:restriction>
</s:simpleType>
```

Return: Stops any more methods from running after the first error is encountered.

Continue: Continues running subsequent methods after an error is encountered.

2.2.5.7 TRUEONLY

Specifies that a particular attribute is restricted to only the value "TRUE".

```
<s:simpleType name="TRUEONLY">
  <s:restriction base="s:string">
    <s:enumeration value="TRUE" />
  </s:restriction>
</s:simpleType>
```

2.2.6 Attributes

This specification does not define any common XML schema attribute definitions.

2.2.7 Groups

This specification does not define any common XML schema group definitions.

2.2.8 Attribute Groups

This specification does not define any common XML schema attribute group definitions.

2.2.9 Common Data Structures

This specification does not define any common XML schema data structures.

Preliminary

3 Protocol Details

The protocol client side of this protocol is simply a pass-through. That is, no additional timers or other state is required on the protocol client side of this protocol. Calls made by the higher-layer protocol or application are passed directly to the transport, and the results returned by the transport are passed directly back to the higher-layer protocol or application.

Except where specified, protocol clients SHOULD interpret HTTP Status Codes returned by the protocol server as specified in [\[RFC2616\]](#), section 10, "Status Code Definitions".

This protocol allows protocol servers to notify protocol clients of application-level faults by using SOAP faults. Except where specified, these SOAP faults are not significant for interoperability, and protocol clients can interpret them in an implementation-specific manner.

This protocol allows protocol servers to perform implementation-specific authorization checks and notify protocol clients of authorization faults either by using HTTP Status Codes or by using SOAP faults as specified previously in this section.

3.1 Lists Soap Server Details

3.1.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this document.

This protocol deals with lists and the items contained within them. For more information, see [\[MS-WSSTS\]](#) sections [2.1.2.7](#) and [2.1.2.10](#).

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

The following table summarizes the list of WSDL operations as defined by this specification:

Operation	Description
AddAttachment	Adds an attachment to the specified list item in the specified list.
AddDiscussionBoardItem	Adds discussion items to a specified discussion board .
AddList	Creates a list in the current site based on the specified name, description, and list template ID.
AddListFromFeature	Creates a list in the current site based on the specified list name, description, feature identifier , and list template

Operation	Description
	identifier .
AddWikiPage	Adds a Wiki page to a Wiki library.
ApplyContentTypeToList	Adds a specific content type to a specific list.
CheckInFile	Checks in a file to a document library.
CheckOutFile	Checks out the file from a document library.
CreateContentType	Creates a new content type on a list.
DeleteAttachment	Removes the attachment from the specified list item in a specified list.
DeleteContentType	Removes a content type from a specific list.
DeleteContentTypeXmlDocument	Deletes an XML Document from XML Document collection in a content type of a list.
DeleteList	Deletes a specified list.
GetAttachmentCollection	Returns the collection of URLs of attachments associated with a specified list item in a specified list.
GetList	Returns the properties and fields for the specified list.
GetListAndView	Returns the properties and fields for the specified list and the properties for the specified View .
GetListCollection	Returns properties for all lists within the site.
GetListContentType	Requests a specific content type from a specific list.
GetListContentTypes	Requests all content types on a specific list.
GetListContentTypesAndProperties	Requests all content types from a specific list and specified properties from the list and site property bags.
GetListItemChanges	Returns all list items satisfying specified criteria that have changed after the date/time specified for a given list. Protocol clients need to use GetListItemChangesSinceToken or GetListItemChangesWithKnowledge in place of this method.
GetListItemChangesSinceToken	Returns all list items satisfying specified criteria that have changed after the date/time implicit in the change token parameter for a given list.
GetListItemChangesWithKnowledge	Returns all list items satisfying specified criteria that have changed after the date/time implicit in the knowledge parameter for a given list.
GetListItems	Returns all list items that meet the specified criteria for a given list.
GetVersionCollection	Returns version information for the specified field of a list.
UndoCheckOut	Undoes the checkout of the specified file in a document library.

Operation	Description
UpdateContentType	Updates a content type on a list.
UpdateContentTypesXmlDocument	Updates the XML document property of the content type collection on the list.
UpdateContentTypeXmlDocument	Updates the XML document of a list content type.
UpdateList	Performs updates on a specified list's properties and adds, deletes, and updates fields for the same list.
UpdateListItems	Performs inserts, updates, or deletions on specified list items in a list.
UpdateListItemsWithKnowledge	Performs inserts, updates, or deletions on specified list items in a list.

3.1.4.1 AddAttachment

The **AddAttachment** operation adds an attachment to the specified list item in the specified list.

```
<wsdl:operation name="AddAttachment">
  <wsdl:input message="AddAttachmentSoapIn" />
  <wsdl:output message="AddAttachmentSoapOut" />
</wsdl:operation>
```

The protocol client sends an **AddAttachmentSoapIn** request message and the server responds with an **AddAttachmentSoapOut** response message, as follows:

- If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
- If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
- If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [<21>](#) return a SOAP fault with error code 0x82000006. This indicates that the List does not exist or it MAY have been deleted by another user.
- If the **listItemID** does not correspond to a list item in a specified list on the site, the protocol server MUST return a SOAP fault. There is no error code returned for this fault.
- If the specified **fileName** is not unique for the specified list item's attachment collection then the protocol server MUST return a SOAP fault with error code 0x81020067. This indicates that the specified **fileName** is already in use.
- If the **attachment** parameter value does not follow the **base64Binary** format, the protocol server MUST return a SOAP fault. There is no error code returned for this fault.
- If the protocol client requests to add an **attachment** of content length zero, the protocol server MUST return a SOAP fault with error code 0x8007000d.
- If the protocol client request excludes the **listItemID** parameter, the protocol server MUST return a SOAP fault with error code 0x82000001. Otherwise, if the **listName** or **attachment** parameters are missing, the protocol server MUST return a SOAP fault with no error code.

Otherwise, if the **fileName** parameter is missing, the protocol server MUST return a SOAP fault with error code 0x81020073.

- If the protocol client passes valid input parameters, the protocol server MUST add the attachment to the specified list item in the specified list.

3.1.4.1.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
AddAttachmentSoapIn	The request to add an attachment to the specified list item in the specified list.
AddAttachmentSoapOut	The response to a request to add an attachment to the specified list item in the specified list.

3.1.4.1.1.1 AddAttachmentSoapIn

This message is the request of the **AddAttachment** operation.

The SOAP Action value of the message is defined as follows:

`http://schemas.microsoft.com/sharepoint/soap/AddAttachment`

The SOAP Body contains an **AddAttachment** element.

3.1.4.1.1.2 AddAttachmentSoapOut

This message is the response of the **AddAttachment** operation.

The SOAP Action value of the message is defined as follows:

`http://schemas.microsoft.com/sharepoint/soap/AddAttachment`

The SOAP Body contains an **AddAttachmentResponse** element.

3.1.4.1.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
AddAttachment	The request to add an attachment to the specified list item in the specified list.
AddAttachmentResponse	Contains the response to a request to add an attachment to the specified list item in the specified list.

3.1.4.1.2.1 AddAttachment

This element specifies the details for adding an attachment to a list item.

```
<s:element name="AddAttachment">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
      <s:element name="listItemID" type="s:string" />
      <s:element name="fileName" type="s:string" />
      <s:element name="attachment" type="s:base64Binary" />
    </s:sequence>
  </s:complexType>
</s:element>
```

listName: The GUID or the list title of the list.

listItemID: The list item identifier.

fileName: The name of the file being added as an attachment.

attachment: A byte array that contains the contents of the file to attach in base-64 encoding.

3.1.4.1.2.2 AddAttachmentResponse

This element contains the response to the request to add an attachment to a list item. If the operation succeeds, an attachment **MUST** be added to the specified list item in the specified list.

```
<s:element name="AddAttachmentResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="AddAttachmentResult" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

AddAttachmentResult: The URL for the newly added attachment.

3.1.4.2 AddDiscussionBoardItem

This operation [<22>](#) is used to add new discussion items to a specified discussion board.

```
<wsdl:operation name="AddDiscussionBoardItem">
  <wsdl:input message="AddDiscussionBoardItemSoapIn" />
  <wsdl:output message="AddDiscussionBoardItemSoapOut" />
</wsdl:operation>
```

The protocol client sends an **AddDiscussionBoardItemSoapIn** request message and the protocol server responds with an **AddDiscussionBoardItemSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.

2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check whether the **listName** corresponds to the list title of a list on the site and if so, use that list.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [<23>](#) return a SOAP fault with error code 0x82000006. This indicates that a list corresponding to the specified **listName** does not exist on the server.
4. If the list is not a discussion board, the protocol server MUST return a SOAP fault. There is no error code returned for this fault.
5. If both parameters are valid, the protocol server MUST add a discussion item to the discussion board.

3.1.4.2.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
AddDiscussionBoardItemSoapIn	The request to add new discussion items to a specified discussion board.
AddDiscussionBoardItemSoapOut	The response to a request to add new discussion items to a specified discussion board.

3.1.4.2.1.1 AddDiscussionBoardItemSoapIn

This message is the request that contains information required by the **AddDiscussionBoardItem** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/AddDiscussionBoardItem
```

The SOAP Body contains an **AddDiscussionBoardItem** element.

3.1.4.2.1.2 AddDiscussionBoardItemSoapOut

This message is the response that contains the results returned by the **AddDiscussionBoardItem** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/AddDiscussionBoardItem
```

The SOAP Body contains an **AddDiscussionBoardItemResponse** element.

3.1.4.2.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
AddDiscussionBoardItem	The request to add new discussion items to a specified discussion board.
AddDiscussionBoardItemResponse	Contains the response to a request to add new discussion items to a specified discussion board.

3.1.4.2.2.1 AddDiscussionBoardItem

This element specifies details for the SOAP request that is sent from the protocol client to the protocol server.

```
<s:element name="AddDiscussionBoardItem">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
      <s:element name="message" type="s:base64Binary" />
    </s:sequence>
  </s:complexType>
</s:element>
```

listName : The GUID or the XML encoded list title of the list.

message: The message to be added to the discussion board. The message MUST be in MIME [\[RFC2045\]](#) format and then Base64 [\[RFC4648\]](#) encoded. Message headers [\[RFC2822\]](#) MUST contain the following text properties.

Field name	String value
Message-id	MUST be present and set to a text value that is unique among all the messages in the Discussion Board.
Thread-index	MUST be present and set to a Base64 [RFC4648] encoded string that is at least 22 bytes long when un-encoded. This value MUST be unique among all the messages in the discussion board.

3.1.4.2.2.2 AddDiscussionBoardItemResponse

This element contains the SOAP response that is returned from the protocol server to the protocol client.

```
<s:element name="AddDiscussionBoardItemResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="AddDiscussionBoardItemResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="listitems" >
              <s:complexType>
                <s:sequence>
                  <s:any />
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

```

        </s:complexType>
    </s:element>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>

```

AddDiscussionBoardItemResult: Contains an XML description of the new discussion board list item.

listitems: The container element for the details of the new discussion board item.

This protocol server response included in the **listitems** element is modeled on the Microsoft ADO 2.6 Persistence format [\[MS-PRSTFR\]](#), excluding the **<s:schema>** element. The **listitems** element includes attributes describing the namespaces for the ADO 2.6 Persistence format, which MAY be ignored by the protocol client, as well as the **TimeStamp** attribute. **listitems** contains an inner element named **rs:data**, which is of type **DataDefinition**.

3.1.4.3 AddList

The **AddList** operation is used to add a new list to the site that is based on the list template that has an identification matching the specified **templateID**.

```

<wsdl:operation name="AddList">
  <wsdl:input message="AddListSoapIn" />
  <wsdl:output message="AddListSoapOut" />
</wsdl:operation>

```

The protocol client sends an **AddListSoapIn** request message and the server responds with an **AddListSoapOut** response message, as follows:

1. If the **templateID** provided is less than 0, the protocol server MUST return a SOAP fault. There is no error code for this fault. This indicates that the list template is invalid.
2. If the **templateID** provided is not one of the known list template identifiers, or no **templateID** parameter is provided, the protocol server MUST return a SOAP fault with error code 0x8102007b. This indicates that the list template is invalid.
3. If the **templateID** provided matches a known template, but a list based on that template already exists and the template is marked as unique, the protocol server MUST return a SOAP fault with error code 0x8102003c. This indicates that **templateID** provided is marked unique and that a list with the specified **templateID** already exists.
4. If the feature identifier that corresponds with the provided **templateID** is not installed on the site, the protocol server MUST return a SOAP fault with error code 0x8107058a. This indicates that the feature for the provided **templateID** is not installed on the site.
5. If the **listName** is already used by another list then the protocol server MUST return a SOAP fault with error code 0x81020012. This indicates that another list has the specified **listName**.

6. Otherwise, a new list named **listName** MUST be created on the site by using the template with an identification matching the provided **templateID** and the list description MUST be the description passed in.

3.1.4.3.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
AddListSoapIn	The request to add a new list to the site that is based on the list template that has an identification matching the specified templateID .
AddListSoapOut	The response to a request to add a new list to the site.

3.1.4.3.1.1 AddListSoapIn

This message is the request to create a new list named **listName** on the site.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/AddList
```

The SOAP Body contains an **AddList** element.

3.1.4.3.1.2 AddListSoapOut

This message is the response that confirms the addition of a new list after a call to the **AddList** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/AddList
```

The SOAP Body contains an **AddListResponse** element.

3.1.4.3.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
AddList	The request to add a new list to the site that is based on the list template that has an identification matching the specified templateID .
AddListResponse	Contains the response to a request to add a new list to the site.

3.1.4.3.2.1 AddList

This element specifies the request to add a new list to the current site with the provided **listName** and **description**, based on the template that has an identification matching the provided **templateID**.

```
<s:element name="AddList">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string"/>
      <s:element name="description" type="s:string" minOccurs="0"/>
      <s:element name="templateID" type="s:int"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

listName : The name for the list to be created on the site.

description : A string that is the description of the new list.

templateID: The list template identifier of a template that is already installed on the site. See section [2.2.4.11](#).

3.1.4.3.2.2 AddListResponse

This element contains the response that is sent to the protocol client after a successful **AddList** call.

```
<s:element name="AddListResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="AddListResult" >
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="List" type="tns:ListDefinitionSchema" />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

AddListResult: Contains information about the properties and schema of the list created by the **AddList** operation. See section [2.2.4.12](#) for more details.

3.1.4.4 AddListFromFeature

The **AddListFromFeature** operation [<24>](#) is used to add a new list to the specified site based on the specified template and feature.

```
<wsdl:operation name="AddListFromFeature">
  <wsdl:input message="AddListFromFeatureSoapIn" />
  <wsdl:output message="AddListFromFeatureSoapOut" />
</wsdl:operation>
```

The protocol client sends an **AddListFromFeatureSoapIn** request message. The server responds with an **AddListFromFeatureSoapOut** response message, as follows:

1. If the **templateID** provided is less than 0, the protocol server MUST return a SOAP fault. There is no error code returned for this fault.
2. If the **templateID** provided is not one of the known list template identifiers, or no **templateID** parameter is provided, the protocol server MUST return a SOAP fault with error string "Parameter {0} is missing or invalid". This indicates that the list template is missing or invalid.
3. If the **templateID** provided matches a known template, but a list based on that template already exists and the template is marked as unique, the protocol server MUST return a SOAP fault with error code 0x8102003c. This indicates that **templateID** provided is marked unique and that a list with the specified **templateID** already exists.
4. If the **featureID** tag is specified and the value is not a GUID or empty, the protocol server MUST return a SOAP fault. There is no error code returned for this fault.
5. If the provided **featureID** is {00000000-0000-0000-0000-000000000000}, or no **featureID** is specified, the protocol server MUST do a lookup on the legacy types (see: list template identifier) to find the correct feature identifier. If it cannot find a **featureID** for the provided **templateID**, the protocol server MUST return a SOAP fault. There is no error code returned for this fault.
6. If the provided **templateID** cannot be used with the provided **featureID**, the protocol server MUST return a SOAP fault with error code 0x81072101. This indicates that the SOAP protocol failed to add a list.
7. If the specified **featureID** is not installed on the site for the specified **templateID**, the protocol server MUST return a SOAP fault with error code 0x8107058a. This indicates that the feature for the provided **templateID** is not installed on the site.
8. If the **listName** is already used by another list then the protocol server MUST return a SOAP fault with error code 0x81020012. This indicates that another list has the specified **listName**.
9. If there are no other errors, a new list MUST be created on the site by using the **listName**, **description**, **featureID**, and **templateID** specified in the **AddListSoapIn** request message.

3.1.4.4.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
AddListFromFeatureSoapIn	The request to add a new list to the specified site based on the specified template and feature.
AddListFromFeatureSoapOut	The response to a request to add a new list to the specified site based on the specified template and feature.

3.1.4.4.1.1 AddListFromFeatureSoapIn

This message is the request to create a new list, named "listName", on the site.

The SOAP Action value of the message is defined as follows:

<http://schemas.microsoft.com/sharepoint/soap/AddListFromFeature>

The SOAP Body contains an **AddListFromFeature** element.

3.1.4.4.1.2 AddListFromFeatureSoapOut

This message is the response that confirms the addition of a new list after a call to **AddList**.

The SOAP Action value of the message is defined as follows:

<http://schemas.microsoft.com/sharepoint/soap/AddListFromFeature>

The SOAP Body contains an **AddListFromFeatureResponse** element.

3.1.4.4.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
AddListFromFeature	The request to add a new list to the specified site based on the specified template and feature.
AddListFromFeatureResponse	Contains the response to a request to add a new list to the specified site based on the specified template and feature.

3.1.4.4.2.1 AddListFromFeature

This element specifies that a list with the specified name and description be added on the specified site by using the predefined template specified by the **templateID** parameter. This request differs from the **AddList** operation in that it utilizes the additional **featureID** parameter to specify which feature to use.

```
<s:element name="AddListFromFeature">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string"/>
      <s:element name="description" type="s:string" minOccurs="0"/>
      <s:element name="featureID" minOccurs="0" >
        <s:simpleType>
          <s:restriction base="s:string">
            <s:pattern value="[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{12}" />
          </s:restriction>
        </s:simpleType>
      </s:element>
      <s:element name="templateID" type="s:int"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

listName : The name of the list to create on the site.

description : A string that is the description of the list to be created.

featureID : The GUID for an installed feature to use for creating the new list.

templateID: The list template identifier of a template that is already installed on the site. See section [2.2.4.11](#).

3.1.4.4.2.2 AddListFromFeatureResponse

This element contains the response that is sent to the protocol client after a successful **AddListFromFeature** call.

```
<s:element name="AddListFromFeatureResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="AddListFromFeatureResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="List" type="tns:ListDefinitionSchema" />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

AddListFromFeatureResult: Contains information about the properties and schema of the list created by the **AddListFromFeature** operation. See section [2.2.4.12](#) for more details

3.1.4.4.5 AddWikiPage

The **AddWikiPage** operation MUST NOT be used.

```
<wsdl:operation name="AddWikiPage">
  <wsdl:input message="AddWikiPageSoapIn" />
  <wsdl:output message="AddWikiPageSoapOut" />
</wsdl:operation>
```

3.1.4.5.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
AddWikiPageSoapIn	The request to add a wiki page to a wiki library.
AddWikiPageSoapOut	The response to a request to add a wiki page to a wiki library.

3.1.4.5.1.1 AddWikiPageSoapIn

This message MUST NOT be used.

3.1.4.5.1.2 AddWikiPageSoapOut

This message MUST NOT be used.

3.1.4.5.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
AddWikiPage	The request to add a wiki page to a wiki library.
AddWikiPageResponse	Contains the response to a request to add a wiki page to a wiki library.

3.1.4.5.2.1 AddWikiPage

This element MUST NOT be used.

```
<s:element name="AddWikiPage">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="strListName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="listRelPageUrl"
        type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="wikiContent" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

strListName: MUST NOT be used.

listRelPageUrl: MUST NOT be used.

wikiContent: MUST NOT be used.

3.1.4.5.2.2 AddWikiPageResponse

This element MUST NOT be used.

```
<s:element name="AddWikiPageResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="AddWikiPageResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="listitems" >
              <s:complexType>
                <s:sequence>
                  <s:any />
                </s:sequence>
                <s:anyAttribute />
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
```

```
</s:element>
</s:sequence>
</s:complexType>
</s:element>
```

3.1.4.6 ApplyContentTypeToList

This operation [<25>](#) is used to apply an existing **site content type** to the requested list.

```
<wsdl:operation name="ApplyContentTypeToList">
  <wsdl:input message="ApplyContentTypeToListSoapIn" />
  <wsdl:output message="ApplyContentTypeToListSoapOut" />
</wsdl:operation>
```

The protocol client sends an **ApplyContentTypeToListSoapIn** request message and the protocol server responds with an **ApplyContentTypeToListSoapOut** response message, as follows:

1. If the content type cannot be found, the protocol server MUST return a SOAP fault. There is no error code returned for this fault.
2. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
3. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
4. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [<26>](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
5. Otherwise, after the method runs, the requested list will have the content type added to its list of valid content types.

3.1.4.6.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
ApplyContentTypeToListSoapIn	The request to apply an existing site content type to the requested list.
ApplyContentTypeToListSoapOut	The response to a request to apply an existing site content type to the requested list.

3.1.4.6.1.1 ApplyContentTypeToListSoapIn

This message is the request of the **ApplyContentTypeToList** operation. The SOAP Action value of the message is defined as follows:

<http://schemas.microsoft.com/sharepoint/soap/ApplyContentTypeToList>

The SOAP Body contains an **ApplyContentTypeToList** element.

3.1.4.6.1.2 ApplyContentTypeToListSoapOut

This message is the response of the **ApplyContentTypeToList** operation.

The SOAP Action value of the message is defined as follows:

<http://schemas.microsoft.com/sharepoint/soap/ApplyContentTypeToList>

The SOAP Body contains an **ApplyContentTypeToListResponse** element.

3.1.4.6.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
ApplyContentTypeToList	The request to apply an existing site content type to the requested list.
ApplyContentTypeToListResponse	Contains the response to a request to apply an existing site content type to the requested list.

3.1.4.6.2.1 ApplyContentTypeToList

This element specifies details for the request to apply a content type to a list.

```
<s:element name="ApplyContentTypeToList">
  <s:complexType>
    <s:sequence>
      <s:element name="webUrl" type="s:string" minOccurs="0" />
      <s:element name="contentTypeId" type="core:ContentTypeId" />
      <s:element name="listName" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

webUrl: This parameter is reserved and MUST be ignored. The value MUST be an empty string if it is present.

contentTypeId: The **content type identifier** of the content type that is to be associated with the specified list. The structure of the content type identifier is described in [MS-WSSTS] section 2.1.2.8.1. Some predefined content type identifiers are described in [MS-WSSTS] section 2.6.

listName: The GUID or the list title of the list in the **context site** that is to have the content type applied to it.

3.1.4.6.2 ApplyContentTypeToListResponse

This element contains the response to the request to add a content type to a list.

```
<s:element name="ApplyContentTypeToListResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="ApplyContentTypeToListResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="Success">
              <s:complexType/>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

ApplyContentTypeToListResult: If the operation succeeds, an **ApplyContentTypeToListResult** MUST be returned. If the operation fails, a SOAP fault MUST be returned. The possible error codes are described in section [3.1.4.6](#).

Success: An empty element, the presence of which indicates that the request was handled successfully.

3.1.4.7 CheckInFile

This operation [<27>](#) is used to check in documents to a document library.

```
<wsdl:operation name="CheckInFile">
  <wsdl:input message="CheckInFileSoapIn" />
  <wsdl:output message="CheckInFileSoapOut" />
</wsdl:operation>
```

The protocol client sends a **CheckInFileSoapIn** request message and the protocol server responds with a **CheckInFileSoapOut** response message, as follows:

- If the **pageUrl** is null or an empty string the protocol server MUST return a SOAP fault with error code 0x82000001. This indicates that the parameter **pageUrl** is missing.
- If the **pageUrl** is an invalid URL, the protocol server MUST return a SOAP fault with error string "Invalid URI: The format of the URI could not be determined". This indicates that the parameter **pageUrl** is invalid.
- If the **pageUrl** does not refer to a document library, the protocol server MUST return a SOAP fault. There is no error code returned for this fault.
- If the **checkInType** parameter is an empty string, the protocol server MUST return a SOAP fault. There is no error code for this fault.

- If the **checkInType** is 2 for a file that has no checked-in copy, the protocol server MUST return a SOAP fault. Such a file could be generated by creating or uploading to a checked out state and never checking in.
- If there are no other errors, the document located at **pageUrl** MUST be checked-in by using **comments** and **checkInType** specified in the **CheckInFileSoapIn** request message.

3.1.4.7.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
CheckInFileSoapIn	The request to check in documents to a document library.
CheckInFileSoapOut	The response to a request to check in documents to a document library.

3.1.4.7.1.1 CheckInFileSoapIn

This message is the request to check in the document giving its location, comments and type of check-in.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/CheckInFile
```

The SOAP Body contains a **CheckInFile** element.

3.1.4.7.1.2 CheckInFileSoapOut

This message is the response that confirms checking in the document.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/CheckInFile
```

The SOAP Body contains a **CheckInFileResponse** element.

3.1.4.7.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
CheckInFile	The request to check in documents to a document library.
CheckInFileResponse	Contains the response to a request to check in documents to a document library.

3.1.4.7.2.1 CheckInFile

This element specifies details for the request to check in a document with the provided **pageUrl**, **comment**, and **CheckinType**.

```
<s:element name="CheckInFile">
  <s:complexType>
    <s:sequence>
      <s:element name="pageUrl" type="s:string" />
      <s:element name="comment" type="s:string" minOccurs="0"/>
      <s:element name="CheckinType" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

pageUrl : The URL of the document to check in.

comment : A string containing check-in comments.

CheckinType: A string representation of the values 0, 1 or 2. These values MUST map to one of the enumerated values in the following table.

Value	Meaning
0	MinorCheckIn
1	MajorCheckIn
2	OverwriteCheckIn

3.1.4.7.2.2 CheckInFileResponse

This element contains the response that is sent to the protocol client after a successful **CheckInFile** operation.

```
<s:element name="CheckInFileResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="CheckInFileResult" type="s:boolean"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

CheckInFileResult: The value of the **CheckInFileResult** specifies whether the call is successful or not. The value is True if the operation is successful; otherwise, False is returned.

3.1.4.8 CheckOutFile

This operation [<28>](#) is used to check out a document in a document library.

```
<wsdl:operation name="CheckOutFile">
  <wsdl:input message="CheckOutFileSoapIn" />
  <wsdl:output message="CheckOutFileSoapOut" />
</wsdl:operation>
```

</wsdl:operation>

The protocol client sends a **CheckOutFileSoapIn** request message and the protocol server responds with a **CheckOutFileSoapOut** response message, as follows:

1. If the **pageUrl** is null or an empty string, the protocol server MUST return a SOAP fault with error code 0x82000001. This indicates that the **pageUrl** is an empty string.
2. If the **pageUrl** is an invalid URL, the protocol server MUST return a SOAP fault. There is no error code for this fault.
3. If the **pageUrl** is not a valid document library, the protocol server MUST return a SOAP fault. There is no error code for this fault.
4. If the **checkoutToLocal** parameter does not resolve to a valid Boolean string (case-insensitive equality to True or False, ignoring leading and trailing white space), the protocol server MUST return a SOAP fault. There is no error code for this fault.
5. If there are no other errors, the document located at **pageUrl** MUST be checked out by using **checkoutToLocal** and **lastmodified** as specified in the **CheckOutFileSoapIn** request message.

3.1.4.8.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
CheckOutFileSoapIn	The request to check out a document in a document library.
CheckOutFileSoapOut	The response to a request to check out a document in a document library.

3.1.4.8.1.1 CheckOutFileSoapIn

This message is the request to check out the document given its location, **checkoutToLocal** flag and **lastmodified** date.

The SOAP Action value of the message is defined as follows:

`http://schemas.microsoft.com/sharepoint/soap/CheckOutFile`

The SOAP Body contains a **CheckOutFile** element.

3.1.4.8.1.2 CheckOutFileSoapOut

This message is the response that confirms checking out the document.

The SOAP Action value of the message is defined as follows:

`http://schemas.microsoft.com/sharepoint/soap/CheckOutFile`

The SOAP Body contains a **CheckOutFileResponse** element.

3.1.4.8.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
CheckOutFile	The request to check out a document in a document library.
CheckOutFileResponse	Contains the response to a request to check out a document in a document library.

3.1.4.8.2.1 CheckOutFile

This element specifies details for the request to check out a document with the provided **pageUrl**, **checkoutToLocal** flag, and **lastmodified** date.

```
<s:element name="CheckOutFile">
  <s:complexType>
    <s:sequence>
      <s:element name="pageUrl" type="s:string" />
      <s:element name="checkoutToLocal" type="s:string" />
      <s:element name="lastmodified" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

pageUrl : The URL of the document to check out.

checkoutToLocal : MUST be true if the caller wants to keep a local version for offline editing.

lastmodified: A string in date format [RFC1123](#) section 5.2.14 that represents the date and time of the last modification by the site (2) to the file; for example, "20 Jun 1982 12:00:00 GMT". If a valid date is passed, it needs to be equal to the **lastmodified** time stamp on the document that is being checked out. If an invalid format of date is passed, this parameter is ignored.

3.1.4.8.2.2 CheckOutFileResponse

This element contains the response that is sent to the protocol client after a successful **CheckOutFile** call.

```
<s:element name="CheckOutFileResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="CheckOutFileResult" type="s:boolean"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

CheckOutFileResult: The value of **CheckOutFileResult** specifies whether the call is successful or not. The value is **True** if the operation is successful; otherwise, **False** is returned.

3.1.4.9 CreateContentType

This operation [<29>](#) is used to create a new content type on a list.

```
<wsdl:operation name="CreateContentType">
  <wsdl:input message="CreateContentTypeSoapIn" />
  <wsdl:output message="CreateContentTypeSoapOut" />
</wsdl:operation>
```

The protocol client sends a **CreateContentTypeSoapIn** request message and the protocol server responds with a **CreateContentTypeSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [<30>](#) return a SOAP fault with error code 0x82000006. This indicates that the **listName** specified does not exist or it MAY have been deleted by another user.
4. If the **parentType** is not a valid content type identifier or is not the content type identifier of a content type available on the list, the protocol server MUST return a SOAP fault. There is no error code for this fault.
5. The protocol server attempts to create a new content type on the list with the given parent content type and display name. If an error specified in section [3.1.4.9.2.1](#) is encountered during this operation, the protocol server MUST return a SOAP fault. Refer section [3.1.4.9.2.1](#) for error code.
6. The protocol server adds the given fields to the newly created content type. If an error regarding the naming of fields, similar to the error specified in section [3.1.4.9.2.1](#), is encountered during this operation, the protocol server MUST return a SOAP fault and the content type will still exist as a newly created content type on the protocol server and all fields added to the content type before the error was encountered MUST stay added to the content type.
7. The protocol server updates the content type with the given properties. If an error regarding the naming of properties, similar to the error specified in section [3.1.4.9.2.1](#), is encountered during this operation, the protocol server MUST return a SOAP fault, and the content type will still exist as a newly created content type on the protocol server. All fields that were added MUST stay added to the content type, and all properties updated before the error was encountered MUST stay updated.

3.1.4.9.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
CreateContentTypeSoapIn	The request to create a new content type on a list.
CreateContentTypeSoapOut	The response to a request to create a new content type on a list.

3.1.4.9.1.1 CreateContentTypeSoapIn

This message is the request of the **CreateContentType** operation.

The **SOAP action** value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/eCreateContentType
```

The **SOAP body** contains a **CreateContentType** element.

3.1.4.9.1.2 CreateContentTypeSoapOut

This message is the response of the **CreateContentType** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/CreateContentType
```

The SOAP body contains a **CreateContentTypeResponse** element.

3.1.4.9.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
CreateContentType	The request to create a new content type on a list.
CreateContentTypeResponse	Contains the response to a request to create a new content type on a list.

3.1.4.9.2.1 CreateContentType

This element specifies details for the request to create a content type on the list.

```
<s:element name="CreateContentType">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
      <s:element name="displayName" type="s:string" />
      <s:element name="parentType" type="core:ContentTypeId" />
      <s:element name="fields" type="tns:AddOrUpdateFieldsDefinition"
        minOccurs="0"/>
      <s:element name="contentTypeProperties" >
        <s:complexType>
          <s:sequence>
            <s:element name="ContentType" type="tns:ContentTypePropertyDefinition"
              minOccurs="0" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="addToView" type="core:TRUE_Case_Sensitive_Else_Anything"
        minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

```
</s:sequence>
</s:complexType></s:element>
```

listName : The GUID or the list title of the list.

displayName : The XML-encoded name of the content type to be created. A SOAP fault with error code 0x80131600 MUST be returned if any of the following conditions applies to the un-encoded version of this element:

- It contains the name of an existing content type on the list, ignoring case.
- It is longer than 128 characters.
- It ends with a period, or begins or ends with whitespace.
- It contains a tab, two consecutive periods, or any of the following characters: \/: * ? " # % < > { } | ~ &

parentType: The identification of a content type from which the content type to be created will inherit.

fields: The container for a list of existing fields to be included in the content type as defined in [\[MS-WSSTS\]](#) section 2.6.

contentTypeProperties: The container for properties to set on the content type as defined in [\[MS-WSSTS\]](#) section 2.6. MUST NOT contain only whitespace.

addToView: Specifies whether the fields will be added to the **default list view**, where "TRUE" MUST correspond to true, and all other values to false.

3.1.4.9.2.2 CreateContentTypeResponse

This element contains the response for a request to create a new content type on the list.

```
<s:element name="CreateContentTypeResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="CreateContentTypeResult" type="core:ContentTypeId" />
    </s:sequence>
  </s:complexType>
</s:element>
```

CreateContentTypeResult: If no error condition, as specified in the preceding section, causes the protocol server to return a SOAP fault, a **CreateContentTypeResult** MUST be returned.

3.1.4.10 DeleteAttachment

This operation is used to remove the attachment from the specified list item in the specified list.

```
<wsdl:operation name="DeleteAttachment">
  <wsdl:input message="DeleteAttachmentSoapIn" />
  <wsdl:output message="DeleteAttachmentSoapOut" />
</wsdl:operation>
```

</wsdl:operation>

The protocol client sends a **DeleteAttachmentSoapIn** request message and the protocol server responds with a **DeleteAttachmentSoapOut** response message, as follows:

1. If the **listItemID** does not correspond to a list item in a specified list on the site, the protocol server MUST return a SOAP fault with error code 0x82000001.
2. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
3. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
4. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [<31>](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
5. If the specified URL does not correspond to the absolute URL path of an attachment in a specified list item in a specified list on the site, the protocol server MUST return a SOAP fault with error code 0x82000001. This indicates that the URL is missing or is invalid.
6. If the protocol client excludes the **listItemID** parameter, the protocol server MUST return a SOAP fault with error code 0x82000001. Otherwise, if the protocol client excludes any of the other parameters, the protocol server MUST return a SOAP fault with no error code.
7. If the protocol client calls the operation by passing valid input parameters, the protocol server MUST delete the attachment associated with the specified list item in the specified list.

3.1.4.10.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
DeleteAttachmentSoapIn	The request to remove the attachment from the specified list item in the specified list.
DeleteAttachmentSoapOut	The response to a request to remove the attachment from the specified list item in the specified list.

3.1.4.10.1.1 DeleteAttachmentSoapIn

This message is the request of the **DeleteAttachment** operation.

The SOAP Action value of the message is defined as follows:

<http://schemas.microsoft.com/sharepoint/soap/DeleteAttachment>

The SOAP Body contains a **DeleteAttachment** element.

3.1.4.10.1.2 DeleteAttachmentSoapOut

This message is the response of the **DeleteAttachment** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/DeleteAttachment
```

The SOAP Body contains a **DeleteAttachmentResponse** element.

3.1.4.10.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
DeleteAttachment	The request to remove the attachment from the specified list item in the specified list.
DeleteAttachmentResponse	Contains the response to a request to remove the attachment from the specified list item in the specified list.

3.1.4.10.2.1 DeleteAttachment

This element contains the request to remove the attachment from the specified list item in a specified list.

```
<s:element name="DeleteAttachment">  
  <s:complexType>  
    <s:sequence>  
      <s:element name="listName" type="s:string" />  
      <s:element name="listItemID" type="s:string" />  
      <s:element name="url" type="s:string" />  
    </s:sequence>  
  </s:complexType>  
</s:element>
```

listName : The GUID or the list title of the list.

listItemID : The identification for the list item.

url : The absolute URL for the attachment. This can be determined by making a **GetAttachmentCollection** request to the protocol server with the same **listName** and **listItemID**.

3.1.4.10.2.2 DeleteAttachmentResponse

This element contains the response to the request to remove the attachment from the specified list item. If the operation succeeds, the attachment **MUST** be deleted; otherwise, the protocol server **MUST** return a SOAP fault with error codes as described in section [3.1.4.10](#).

```
<s:element name="DeleteAttachmentResponse">  
  <s:complexType/>
```

</s:element>

3.1.4.11 DeleteContentType

This operation [32](#) is used to remove the association between the specified list and the specified content type.

```
<wsdl:operation name="DeleteContentType">
  <wsdl:input message="DeleteContentTypeSoapIn" />
  <wsdl:output message="DeleteContentTypeSoapOut" />
</wsdl:operation>
```

The protocol client sends a **DeleteContentTypeSoapIn** request message and the protocol server responds with a **DeleteContentTypeSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [33](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
4. If the content type cannot be found on the list, the protocol server MUST return a SOAP fault. There is no error code for this fault.
5. If there are no other errors, the content type is removed from the list.

3.1.4.11.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
DeleteContentTypeSoapIn	The request to remove the association between the specified list and the specified content type.
DeleteContentTypeSoapOut	The response to a request to remove the association between the specified list and the specified content type.

3.1.4.11.1.1 DeleteContentTypeSoapIn

This message is the request of the **DeleteContentType** operation. The SOAP Action value of the message is defined as:

```
http://schemas.microsoft.com/sharepoint/soap/DeleteContentType
```

The SOAP Body contains a **DeleteContentType** element.

3.1.4.11.1.2 DeleteContentTypeSoapOut

This message is the response of the **DeleteContentType** operation. The SOAP Action value of the message is defined as:

```
http://schemas.microsoft.com/sharepoint/soap/DeleteContentType
```

The SOAP Body contains a **DeleteContentTypeResponse** element.

3.1.4.11.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
DeleteContentType	The request to remove the association between the specified list and the specified content type.
DeleteContentTypeResponse	Contains the response to a request to remove the association between the specified list and the specified content type.

3.1.4.11.2.1 DeleteContentType

This element specifies details for the request to delete the content type from the list.

```
<s:element name="DeleteContentType"> <s:complexType> <s:sequence>  
  <s:element name="listName" type="s:string" />  
  <s:element name="contentTypeId" type="core:ContentTypeId" />  
</s:sequence>  
</s:complexType></s:element>
```

listName : The GUID or the list title of the list in the context site from which to remove the content type.

contentTypeId: The content type identifier, as specified in [\[MS-WSSCAML\]](#) of the content type that is to be removed from the specified list.

3.1.4.11.2.2 DeleteContentTypeResponse

This element contains the response to the request to delete the content type.

```
<s:element name="DeleteContentTypeResponse">  
  <s:complexType>  
    <s:sequence>  
      <s:element name="DeleteContentTypeResult" minOccurs="0">  
        <s:complexType mixed="true">  
          <s:sequence>  
            <s:element name="Success">  
              <s:complexType />  
            </s:element>  
          </s:sequence>  
        </s:complexType>  
      </s:element>  
    </s:sequence>  
  </s:complexType>
```

```

    </s:element>
  </s:sequence>
</s:complexType>
</s:element>

```

DeleteContentTypeResult: If the operation succeeds, a **DeleteContentTypeResult** MUST be returned. If the operation fails, a SOAP fault MUST be returned.

Success: An empty element, the presence of which indicates that the request was handled successfully.

3.1.4.12 DeleteContentTypeXmlDocument

This operation [<34>](#) is used to delete an XML document Property from XML document collection in a content type of a list.

```

<wsdl:operation name="DeleteContentTypeXmlDocument">
  <wsdl:input message="DeleteContentTypeXmlDocumentSoapIn" />
  <wsdl:output message="DeleteContentTypeXmlDocumentSoapOut" />
</wsdl:operation>

```

The protocol client sends a **DeleteContentTypeXmlDocumentSoapIn** request message and the protocol server responds with a **DeleteContentTypeXmlDocumentSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [<35>](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
4. If the content type specified by the content **TypeId** element is not found, a SOAP fault MUST be thrown. There is no error code with this fault.
5. If an XML document with the same namespace **URI** [\[RFC2396\]](#) specified by **documentUri** exists, it MUST be deleted.
6. If no SOAP fault has been thrown, the protocol server MUST respond with **DeleteContentTypeXmlDocumentResult**.

3.1.4.12.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
DeleteContentTypeXmlDocumentSoapIn	The request to delete an XML document Property from

Message	Description
	XML document collection in a content type of a list.
DeleteContentTypeXmlDocumentSoapOut	The response to a request to delete an XML document Property from XML document collection in a content type of a list.

3.1.4.12.1.1 DeleteContentTypeXmlDocumentSoapIn

This message is the request of the **DeleteContentTypeXmlDocument** operation. The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/DeleteContentTypeXmlDocument
```

The SOAP Body contains a **DeleteContentTypeXmlDocument** element.

3.1.4.12.1.2 DeleteContentTypeXmlDocumentSoapOut

This message is the response of the **DeleteContentTypeXmlDocument** operation. The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/DeleteContentTypeXmlDocument
```

The SOAP Body contains a **DeleteContentTypeXmlDocumentResponse** element.

3.1.4.12.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
DeleteContentTypeXmlDocument	The request to delete an XML document Property from XML document collection in a content type of a list.
DeleteContentTypeXmlDocumentResponse	Contains the response to a request to delete an XML document Property from XML document collection in a content type of a list.

3.1.4.12.2.1 DeleteContentTypeXmlDocument

This element specifies details for the request to delete an XML Document from an XML Document collection in a content type of a list.

```
<s:element name="DeleteContentTypeXmlDocument">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
      <s:element name="contentTypeId" type="core:ContentTypeId" />
      <s:element name="documentUri" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

listName : The GUID or the list title of the list in the context site where the content type is to be modified.

contentTypeId : The content type identifier of the content type that is to be modified.

documentUri: The namespace URI [\[RFC2396\]](#) of the XML document to remove.

3.1.4.12.2.2 DeleteContentTypeXmlDocumentResponse

This element contains the response for deleting an XML document from an XML document collection in a content type of a list.

```
<s:element name="DeleteContentTypeXmlDocumentResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="DeleteContentTypeXmlDocumentResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="Success" minOccurs="0">
              <s:complexType />
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

DeleteContentTypeXmlDocumentResult: If the operation succeeds, a **DeleteContentTypeXmlDocumentResult** MUST be returned. If the operation fails, a SOAP fault MUST be returned.

Success: An empty element, the presence of which indicates that the request was handled successfully.

3.1.4.13 DeleteList

The **DeleteList** operation deletes the specified list from the specified site.

```
<wsdl:operation name="DeleteList">
  <wsdl:input message="DeleteListSoapIn" />
  <wsdl:output message="DeleteListSoapOut" />
</wsdl:operation>
```

The protocol client sends a **DeleteListSoapIn** request message. The protocol server MUST respond with a **DeleteListSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, that list MUST be deleted and the protocol server MUST return a **DeleteListResponse** element.

2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, that list MUST be deleted and the protocol server MUST return a **DeleteListResponse** element.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [36](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or it might have been deleted by another user.
4. If there are no other errors, the list MUST be deleted.

3.1.4.13.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
DeleteListSoapIn	The request to delete the specified list from the specified site.
DeleteListSoapOut	The response to a request to delete the specified list from the specified site.

3.1.4.13.1.1 DeleteListSoapIn

This message is the request to delete the list that has a GUID or name corresponding to the specified **listName**.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/DeleteList
```

The SOAP Body contains a **DeleteList** element.

3.1.4.13.1.2 DeleteListSoapOut

This message is the response that confirms the deletion of a list by the **DeleteList** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/DeleteList
```

The SOAP Body contains a **DeleteListResponse** element.

3.1.4.13.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
DeleteList	The request to delete the specified list from the specified site.
DeleteListResponse	Contains the response to a request to delete the specified list from the specified site.

3.1.4.13.2.1 DeleteList

This element specifies details for the request to delete a list from the site.

```
<s:element name="DeleteList">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

listName: The GUID or the list title of the list.

3.1.4.13.2.2 DeleteListResponse

This element contains a single element that MUST be sent by the site if the **DeleteList** operation succeeds.

```
<s:element name="DeleteListResponse"> <s:complexType/></s:element>
```

3.1.4.14 GetAttachmentCollection

This operation is used to return the list of URLs of attachments associated with a specified list item in a specified list.

```
<wsdl:operation name="GetAttachmentCollection">
  <wsdl:input message="GetAttachmentCollectionSoapIn" />
  <wsdl:output message="GetAttachmentCollectionSoapOut" />
</wsdl:operation>
```

The protocol client sends a **GetAttachmentCollectionSoapIn** request message and the protocol server responds with a **GetAttachmentCollectionSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [<37>](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or it might have been deleted by another user.
4. If the **listItemID** is a negative value, the server MUST return a SOAP fault with error code 0x81020014.
5. If the **listItemID** does not correspond to a list item in a specified list on the site, the protocol server MUST return a SOAP fault. There is no error code for this fault.
6. If the protocol client excludes the **listItemID** parameter, the protocol server MUST return a SOAP fault with error code 0x82000001.

7. If the protocol client excludes the **listName** parameter, the protocol server MUST return a SOAP fault with no error code.
8. If the protocol client passes valid input parameters, the protocol server MUST return the collection of attachments associated with the list item.
9. If the protocol client calls the operation on a **listItemID** containing zero attachments, the protocol server MUST return an empty attachment collection.

3.1.4.14.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetAttachmentCollectionSoapIn	The request to return the list of URLs of attachments associated with a specified list item in a specified list.
GetAttachmentCollectionSoapOut	The response to a request to return the list of URLs of attachments associated with a specified list item in a specified list.

3.1.4.14.1.1 GetAttachmentCollectionSoapIn

This message is the request of the **GetAttachmentCollection** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetAttachmentCollection
```

The SOAP Body contains a **GetAttachmentCollection** element.

3.1.4.14.1.2 GetAttachmentCollectionSoapOut

This message is the response of the **GetAttachmentCollection** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetAttachmentCollection
```

The SOAP Body contains a **GetAttachmentCollectionResponse** element.

3.1.4.14.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetAttachmentCollection	The request to return the list of URLs of attachments associated with a specified list item in a specified list.
GetAttachmentCollectionResponse	Contains the response to a request to return the list of URLs of attachments associated with a specified list item in a specified

Element	Description
	list.

3.1.4.14.2.1 GetAttachmentCollection

This element specifies details for the request to return the list of URLs of attachments associated with a specified list item.

```
<s:element name="GetAttachmentCollection">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
      <s:element name="listItemID" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

listName : The GUID or the list title of the list.

listItemID: The identification for the list item. This value does not correspond to the index of the list item within the collection of list items.

3.1.4.14.2.2 GetAttachmentCollectionResponse

This element contains the response to the request to return the list of URLs of attachments associated with a specified list item. If the operation succeeds, the protocol server MUST return the collection of attachment URLs for the specified list item; otherwise, the protocol server MUST return a SOAP fault with error codes as described in section [3.1.4.14](#).

```
<s:element name="GetAttachmentCollectionResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="GetAttachmentCollectionResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="Attachments">
              <s:complexType>
                <s:sequence>
                  <s:element name="Attachment" minOccurs="0"
                    maxOccurs="unbounded">
                    <s:complexType />
                  </s:element>
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

GetAttachmentCollectionResult : Contains the collection of attachments. Each attachment in turn refers to the absolute attachment path on the server as shown in the following element descriptions.

Attachments: The collection of all attachments for the specified list item.

Attachment: The URL for a particular attachment.

3.1.4.15 GetList

The **GetList** operation is used to retrieve properties and fields for a specified list.

```
<wsdl:operation name="GetList">
  <wsdl:input message="GetListSoapIn" />
  <wsdl:output message="GetListSoapOut" />
</wsdl:operation>
```

The protocol client sends a **GetListSoapIn** request message. The protocol server MUST respond with a **GetListSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, the protocol server MUST return a **GetListResponse** element.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site, and if so the protocol server MUST return a **GetListResponse** element.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [38](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.

3.1.4.15.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetListSoapIn	The request to retrieve properties and fields for a specified list.
GetListSoapOut	The response to a request to retrieve properties and fields for a specified list.

3.1.4.15.1.1 GetListSoapIn

This message is the request that contains information required by the **GetList** operation. The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetList
```

The SOAP Body contains a **GetList** element.

3.1.4.15.1.2 GetListSoapOut

This message is the response that contains the results returned by the **GetList** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetList
```

The SOAP Body contains a **GetListResponse** element.

3.1.4.15.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetList	The request to retrieve properties and fields for a specified list.
GetListResponse	Contains the response to a request to retrieve properties and fields for a specified list.

3.1.4.15.2.1 GetList

This element specifies details for the SOAP request that is sent from the protocol client to the protocol server.

```
<s:element name="GetList">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

listName: The GUID or the list title of the list from which to retrieve properties and fields.

3.1.4.15.2.2 GetListResponse

This element contains the SOAP response that is returned from the protocol server to the protocol client.

```
<s:element name="GetListResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="GetListResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="List" type="tns:ListDefinitionSchema" />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```


GetListResult: The container element for the results.

List: Contains information about the properties and schema of the specified list. See section [2.2.4.12](#) for more details.

3.1.4.16 GetListAndView

This operation is used to retrieve properties and fields for a specified list and a view.

```
<wsdl:operation name="GetListAndView">
  <wsdl:input message="GetListAndViewSoapIn" />
  <wsdl:output message="GetListAndViewSoapOut" />
</wsdl:operation>
```

The protocol client sends a **GetListAndViewSoapIn** request message. The server MUST respond with a **GetListAndViewSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, the protocol server MUST return a **GetListAndViewResponse** element.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, the protocol server MUST return a **GetListAndViewReponse** element.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [<39>](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
4. If the specified **viewName** is not specified or is an empty string, the default view properties MUST be returned.
5. If the specified **viewName** is not a valid GUID and is not an empty string, the protocol server MUST return a SOAP fault with error code 0x82000001. This indicates that a required parameter is missing or invalid.
6. If the specified **viewName** does not correspond to an existing **viewName** for the given list and it is not an empty string, the protocol server MUST return a SOAP fault with error code 0x82000005. This indicates that the view does not exist or might have been deleted by another user.

3.1.4.16.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetListAndViewSoapIn	The request to retrieve properties and fields for a specified list and a view.
GetListAndViewSoapOut	The response to a request to retrieve properties and fields for a specified list and a view.

3.1.4.16.1.1 GetListAndViewSoapIn

This message is the request that contains the information required by the **GetListAndView** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListAndView
```

The SOAP Body contains a **GetListAndView** element.

3.1.4.16.1.2 GetListAndViewSoapOut

This message is the response that contains the results returned by the **GetListAndView** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListAndView
```

The SOAP Body contains a **GetListAndViewResponse** element.

3.1.4.16.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetListAndView	The request to retrieve properties and fields for a specified list and a view.
GetListAndViewResponse	Contains the response to a request to retrieve properties and fields for a specified list and a view.

3.1.4.16.2.1 GetListAndView

This element specifies details for the request that is sent from the protocol client to the protocol server.

```
<s:element name="GetListAndView">  
  <s:complexType>  
    <s:sequence>  
      <s:element name="listName" type="s:string" />  
      <s:element name="viewName" type="s:string" minOccurs="0"/>  
    </s:sequence>  
  </s:complexType>  
</s:element>
```

listName : The GUID or the list title of the list from which to retrieve properties and fields.

viewName: The GUID of the view for which to retrieve properties.

3.1.4.16.2.2 GetListAndViewResponse

This element contains the response that is sent from the protocol server to the protocol client.

```
<s:element name="GetListAndViewResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="GetListAndViewResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="ListAndView">
              <s:complexType mixed="true">
                <s:sequence>
                  <s:element name="List" type="tns:ListDefinitionSchema" />
                  <s:element name="View" type="core:ViewDefinition" />
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

GetListAndViewResult: The container element for the results.

ListDefinitionSchema: Specifies the schema for the list.

ViewDefinition: Specifies the schema for the view.

3.1.4.17 GetListCollection

This operation is used to retrieve information about all the lists on the current site.

```
<wsdl:operation name="GetListCollection">
  <wsdl:input message="GetListCollectionSoapIn" />
  <wsdl:output message="GetListCollectionSoapOut" />
</wsdl:operation>
```

The protocol client sends a **GetListCollectionSoapIn** request message and the server responds with a **GetListCollectionSoapOut** response message.

3.1.4.17.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetListCollectionSoapIn	The request to retrieve information about all the lists on the current site.
GetListCollectionSoapOut	The response to the request to retrieve information about all the lists on the current site.

3.1.4.17.1.1 GetListCollectionSoapIn

This message is the request that contains information required by the **GetListCollection** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListCollection
```

The SOAP Body contains a **GetListCollection** element.

3.1.4.17.1.2 GetListCollectionSoapOut

This message is the response that contains the results returned by the **GetListCollection** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListCollection
```

The SOAP Body contains a **GetListCollectionResponse** element.

3.1.4.17.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetListCollection	The request to retrieve information about all the lists on the current site.
GetListCollectionResponse	Contains the response to the request to retrieve information about all the lists on the current site.

3.1.4.17.2.1 GetListCollection

This element specifies details for the request that is sent from the protocol client to the protocol server.

```
<s:element name="GetListCollection">  
  <s:complexType/>  
</s:element>
```

3.1.4.17.2.2 GetListCollectionResponse

This element contains the response that is returned from the protocol server to the protocol client.

```
<s:element name="GetListCollectionResponse">  
  <s:complexType>  
    <s:sequence>  
      <s:element name="GetListCollectionResult" minOccurs="0">  
        <s:complexType mixed="true">  
          <s:sequence>
```

```

    <s:element name="Lists">
      <s:complexType>
        <s:sequence>
          <s:element name="List" type="tns:ListDefinitionCT" minOccurs="0"
            maxOccurs="unbounded"/>
        </s:sequence>
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>
</s:element>

```

GetListCollectionResult: Contains a collection of properties for all the lists within the current site.

Lists: The container element for the individual list results.

List: Contains information about the properties and schema of the specified list. See section [2.2.4.11](#) for more details.

3.1.4.18 GetListContentType

This operation [<40>](#) is used to get content type data for a given content type identifier.

```

<wsdl:operation name="GetListContentType">
  <wsdl:input message="GetListContentTypeSoapIn" />
  <wsdl:output message="GetListContentTypeSoapOut" />
</wsdl:operation>

```

The protocol client sends a **GetListContentTypeSoapIn** request message and the protocol server responds with a **GetListContentTypeSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [<41>](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
4. If **contentTypeId** does not match a content type identifier on the list, the protocol server MUST return a SOAP fault. There is no error code for this fault.
5. If no other errors occurred the protocol server MUST respond with a **GetListContentTypeResponse**.

3.1.4.18.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetListContentTypeSoapIn	The request to get content type data for a given content type identifier.
GetListContentTypeSoapOut	The response to a request to get content type data for a given content type identifier.

3.1.4.18.1.1 GetListContentTypeSoapIn

This message is the request to return content type data. The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListContentType
```

The SOAP Body contains a **GetListContentType** element.

3.1.4.18.1.2 GetListContentTypeSoapOut

This message is the response to the request to return content type data. The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListContentType
```

The SOAP Body contains a **GetListContentTypeResponse** element.

3.1.4.18.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetListContentType	The request to get content type data for a given content type identifier.
GetListContentTypeResponse	Contains the response to a request to get content type data for a given content type identifier.

3.1.4.18.2.1 GetListContentType

This element specifies details for the request to return data for a specific content type.

```
<s:element name="GetListContentType">  
  <s:complexType>  
    <s:sequence>  
      <s:element name="listName" type="s:string" />  
      <s:element name="contentTypeid" type="core:ContentTypeId" />  
    </s:sequence>  
  </s:complexType>  
</s:element>
```

```
</s:complexType>
</s:element>
```

listName : The GUID or the list title of the list.

contentTypeId: The content type identifier that identifies the content type data to return.

3.1.4.18.2.2 GetListContentTypeResponse

This element contains the response to the request to return content type data.

```
<s:element name="GetListContentTypeResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="GetListContentTypeResult" minOccurs="0">
        <s:complexType >
          <s:sequence>
            <s:element name="ContentType" type="core:ContentTypeDefinition"/>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

GetListContentTypeResult: The container for the returned content type data.

ContentType: The node containing data for the content type. See [\[MS-WSSCAML\]](#) section 2.4.

3.1.4.19 GetListContentTypes

This operation [<42>](#) is used to retrieve all content types from a list.

```
<wsdl:operation name="GetListContentTypes">
  <wsdl:input message="GetListContentTypesSoapIn" />
  <wsdl:output message="GetListContentTypesSoapOut" />
</wsdl:operation>
```

The protocol client sends a **GetListContentTypesSoapIn** message and the protocol server responds with a **GetListContentTypesSoapOut** message.

3.1.4.19.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetListContentTypesSoapIn	The request to retrieve all content types from a list.
GetListContentTypesSoapOut	The response to a request to retrieve all content types from a list.

3.1.4.19.1.1 GetListContentTypesSoapIn

This message is the request to return all content types on a list. The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListContentTypes
```

The SOAP Body contains a **GetListContentTypes** element.

3.1.4.19.1.2 GetListContentTypesSoapOut

This message is the response to the request to return all content types on a list. The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListContentTypes
```

The SOAP Body contains a **GetListContentTypesResponse** element.

3.1.4.19.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetListContentTypes	The request to retrieve all content types from a list.
GetListContentTypesResponse	Contains the response to a request to retrieve all content types from a list.

3.1.4.19.2.1 GetListContentTypes

This element specifies the details of the request to return all content types on a list.

```
<s:element name="GetListContentTypes">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
      <s:element name="contentTypeid" type="core:ContentTypeId" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

listName : The GUID or title of the list from which data SHOULD be returned.

contentTypeid : The identifier of a content type for which the client requests that the server find a "best match" among those content types that exist on the list, as follows:

- If **listName** is a valid GUID and corresponds to the identification of a list on the site, the protocol server MUST use that list.

- If **listName** is not a valid GUID or does not correspond to the identification of a list on the site, the protocol server **MUST** check to see if **listName** corresponds to the title of a list on the site and if so, **MUST** use that list.
- If **listName** does not correspond to a list from either of these checks, the protocol server **SHOULD** [43](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user. If the **contentTypeId** parameter is specified, the protocol server **MUST** attempt to find the "best match" to the specified content type from among those on the list.
- If the content type specified by **contentTypeId** exists on the list, the server **MUST** choose that content type as the best match. Otherwise, the algorithm for determining best match is a server implementation choice. If the server succeeds in finding a best match to the specified content type, the server **MUST** set the **BestMatch** attribute of the selected content type to "TRUE" in the **GetListContentTypesResponse** element.
- If the server is unable to find a "best match," the server **MUST NOT** set the **BestMatch** attribute to "TRUE" for any of the content types returned in the **GetListContentTypesResponse** element.

3.1.4.19.2.2 GetListContentTypesResponse

This element is the response that contains all content types on a list.

```

<s:element name="GetListContentTypesResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="GetListContentTypesResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="ContentTypes" >
              <s:complexType>
                <s:sequence>
                  <s:element name="ContentType" maxOccurs="unbounded">
                    <s:complexType>
                      <s:sequence>
                        <s:element name="XmlDocuments"
                          type="core:XmlDocumentDefinitionCollection"
                          minOccurs="0">
                        </s:element>
                      </s:sequence>
                    </s:complexType>
                  </s:element>
                </s:sequence>
                <s:attribute name="Name" type="s:string" use="required" />
                <s:attribute name="ID" type="core:ContentTypeId"
                  use="required" />
                <s:attribute name="Description" type="s:string"
                  use="required" />
                <s:attribute name="Scope" type="s:string" use="required" />
                <s:attribute name="Version" type="s:int" use="required" />
                <s:attribute name="BestMatch" type="tns:TRUEONLY"
                  use="optional" />
              </s:complexType>
            </s:element>
          </s:sequence>
          <s:attribute name="ContentTypeOrder" type="s:string" use="optional" />
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>

```

```

        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>

```

GetListContentTypesResult: The container element for the results.

GetListContentTypesResult.ContentTypes: The container element for the content types.

GetListContentTypesResult.ContentTypes.ContentTypeOrder: A string of content type identifiers delimited by commas. This specifies the order in which the content types are displayed to the user when the user tries to create a new item in this list. This MUST NOT be returned if the list is a user information list. [<44>](#)

GetListContentTypesResult.ContentTypes.ContentType: The container element for a single content type, as specified in [\[MS-WSSCAML\]](#) section 2.4.

GetListContentTypesResult.ContentTypes.ContentType.XmlDocuments: A collection of XML documents associated with this content type, which can include forms and **event receiver** manifests. The **XmlDocumentDefinitionCollection** type is specified in [\[MS-WSSCAML\]](#) section 2.4.12.

GetListContentTypesResult.ContentTypes.ContentType.Name: The name of the content type.

GetListContentTypesResult.ContentTypes.ContentType.ID: The content type identifier for this content type.

GetListContentTypesResult.ContentTypes.ContentType.Description: The description of the content type.

GetListContentTypesResult.ContentTypes.ContentType.Scope: The fully qualified URL to the content type scope.

GetListContentTypesResult.ContentTypes.ContentType.Version: Specifies the current version of the content type. The protocol server MUST increment the value by 1 each time the content type definition is edited. The protocol client MUST ignore this attribute.

GetListContentTypesResult.ContentTypes.ContentType.BestMatch: MUST be specified on at most one **ContentType** element. If specified at all, then MUST be specified on the **ContentType** element that represents the "best match" to the content type identified by the **contentTypeId** element in the **GetListContentTypesSoapIn** message.

3.1.4.20 GetListContentTypesAndProperties

This operation is used to retrieve all content types from a list and specified properties from the list and site **property bags**. All protocol servers SHOULD [<45>](#) support this operation.

```

<wsdl:operation name="GetListContentTypesAndProperties">
  <wsdl:input message="GetListContentTypesAndPropertiesSoapIn" />
  <wsdl:output message="GetListContentTypesAndPropertiesSoapOut" />

```

</wsdl:operation>

The protocol client sends a **GetListContentTypesAndPropertiesSoapIn** message and the protocol server responds with a **GetListContentTypesAndPropertiesSoapOut** message.

3.1.4.20.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetListContentTypesAndPropertiesSoapIn	The request to retrieve all content types from a list and specified properties from the list and site property bags.
GetListContentTypesAndPropertiesSoapOut	The response to a request to retrieve all content types from a list and specified properties from the list and site property bags.

3.1.4.20.1.1 GetListContentTypesAndPropertiesSoapIn

This message is the request to return content types and properties from a list and, optionally, properties from the site. The SOAP Action value of the message is defined as follows:

`http://schemas.microsoft.com/sharepoint/soap/GetListContentTypesAndProperties`

The SOAP Body contains a **GetListContentTypesAndProperties** element.

3.1.4.20.1.2 GetListContentTypesAndPropertiesSoapOut

This message is the response to the **GetListContentTypesAndPropertiesSoapIn** request. The SOAP Action value of the message is defined as follows:

`http://schemas.microsoft.com/sharepoint/soap/GetListContentTypesAndProperties`

The SOAP Body contains a **GetListContentTypesAndPropertiesResponse** element.

3.1.4.20.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetListContentTypesAndProperties	The request to retrieve all content types from a list and specified properties from the list and site property bags.
GetListContentTypesAndPropertiesResponse	Contains the response to a request to retrieve all content types from a list and specified properties from the list and site property bags.

3.1.4.20.2.1 GetListContentTypesAndProperties Element

This element specifies the details of the **GetListContentTypesAndPropertiesSoapIn** request.

```
<s:element name="GetListContentTypesAndProperties">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" maxOccurs="1"/>
      <s:element name="contentTypeId" type="core:ContentTypeId" minOccurs="0"
maxOccurs="1"/>
      <s:element name="propertyPrefix" type="s:string" minOccurs="0" maxOccurs="1"/>
      <s:element name="includeWebProperties" type="s:boolean" minOccurs="0" maxOccurs="1"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

listName : The GUID or title of the list from which data SHOULD be returned.

contentTypeId : The identifier of a content type for which the client requests that the server find a "best match" among those content types that exist on the list.

propertyPrefix : The prefix of the requested property keys.

includeWebProperties : Specifies the protocol client's request for properties and files from the property bag of the site that contains the specified list. If the element is True, the protocol server MUST return properties and files from the site property bag. If the element is False, the protocol server MUST NOT return properties and files from the site property bag.

If **listName** is a valid GUID and corresponds to the identification of a list on the site, the protocol server MUST use that list. If **listName** is not a valid GUID or does not correspond to the identification of a list on the site, the protocol server MUST check to see if **listName** corresponds to the title of a list on the site and if so, MUST use that list. If **listName** does not correspond to a list from either of these checks, the protocol server MUST return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.

If the **contentTypeId** parameter is specified, the protocol server MUST attempt to find the "best match" to the specified content type from among those on the list. If the content type specified by **contentTypeId** exists on the list, the server MUST choose that content type as the best match. Otherwise, the algorithm for determining best match is a server implementation choice. If the server succeeds in finding a best match to the specified content type, the server MUST set the **BestMatch** attribute of the selected content type to "TRUE" in the **GetListContentTypesAndPropertiesResponse** element. If the server is unable to find a "best match," the server MUST NOT set the **BestMatch** attribute to "TRUE" for any of the content types returned in the **GetListContentTypesAndPropertiesResponse** element.

If the **propertyPrefix** parameter is not empty, the protocol server MUST return all of the properties in the list property bag whose property key begins with the **propertyPrefix**. The protocol sever MUST also return the contents of all files referenced by a property in the list property bag whose key begins with **propertyPrefix** and ends with the suffix "_file". If the **propertyPrefix** parameter is empty, the protocol server MUST return all of the properties in the list property bag, regardless of key, and MUST return the contents of all files referenced by properties in the list property bag whose keys end with the "_file" suffix. If the **includeWebProperties** parameter is set to "TRUE" the protocol server MUST return properties and files from the site property bag in addition to properties and files from the list property bag. The selection of properties from the site property bag MUST follow the same rules as used for the list property bag (described in the preceding

paragraph). If **includeWebProperties** is "FALSE", the protocol server MUST NOT return any properties or files from the site property bag.

3.1.4.20.2.2 GetListContentTypesAndPropertiesResponse

This element provides the content types and properties requested by the **GetListContentTypesAndPropertiesSoapIn** message.

```
<s:element name="GetListContentTypesAndPropertiesResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="GetListContentTypesAndPropertiesResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="ContentTypes" >
              <s:complexType>
                <s:sequence>
                  <s:element name="ContentType" maxOccurs="unbounded">
                    <s:complexType>
                      <s:sequence>
                        <s:element name="XmlDocuments"
                          type="core:XmlDocumentDefinitionCollection"
                          minOccurs="0">
                        </s:element>
                      </s:sequence>
                      <s:attribute name="Name" type="s:string" use="required" />
                      <s:attribute name="ID" type="core:ContentTypeId"
                        use="required" />
                      <s:attribute name="Description" type="s:string"
                        use="required" />
                      <s:attribute name="Scope" type="s:string" use="required" />
                      <s:attribute name="Version" type="s:int" use="required" />
                      <s:attribute name="BestMatch" type="tns:TRUEONLY"
                        use="optional" />
                    </s:complexType>
                  </s:element>
                </s:sequence>
                <s:attribute name="ContentTypeOrder" type="s:string" use="optional" />
              </s:complexType>
            </s:element>
            <s:element name="Properties">
              <s:complexType>
                <s:sequence>
                  <s:element name="Property" minOccurs="0" maxOccurs="unbounded">
                    <s:complexType>
                      <s:attribute name="Key" type="s:string"
                        use="required" />
                      <s:attribute name="Value" type="s:string"
                        use="required" />
                    </s:complexType>
                  </s:element>
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

```
</s:sequence>
</s:complexType>
</s:element>
```

GetListContentTypesAndPropertiesResult: The container element for the results.

GetListContentTypesAndPropertiesResult.ContentTypes: The container element for the content types.

GetListContentTypesAndPropertiesResult.ContentTypes.ContentTypeOrder: A string of content type identifiers delimited by commas. This specifies the order in which the content types are displayed to the user when the user tries to create a new item in this list. This MUST NOT be returned if the list is a user information list. [<46>](#)

GetListContentTypesAndPropertiesResult.ContentTypes.ContentType: The container element for a single content type, as specified in [\[MS-WSSCAML\]](#) section 2.4.

GetListContentTypesAndPropertiesResult.ContentTypes.ContentType.XmlDocuments: A collection of XML documents associated with this content type, which can include forms and event receiver manifests. The **XmlDocumentDefinitionCollection** type is specified in [\[MS-WSSCAML\]](#) section 2.4.12

GetListContentTypesAndPropertiesResult.ContentTypes.ContentType.Name: The name of the content type.

GetListContentTypesAndPropertiesResult.ContentTypes.ContentType.ID: The content type identifier for this content type.

GetListContentTypesAndPropertiesResult.ContentTypes.ContentType.Description: The description of the content type.

GetListContentTypesAndPropertiesResult.ContentTypes.ContentType.Scope: The fully qualified URL to the content type scope.

GetListContentTypesAndPropertiesResult.ContentTypes.ContentType.Version: Specifies the current version of the content type. The protocol server MUST increment the value by 1 each time the content type definition is edited. The protocol client MUST ignore this attribute.

GetListContentTypesAndPropertiesResult.ContentTypes.ContentType.BestMatch: MUST be specified on at most one **ContentType** element. If specified at all, then it MUST be specified on the **ContentType** element that represents the "best match" to the content type identified by the **contentTypeid** element in the **GetListContentTypesAndPropertiesSoapIn** message. See section [3.1.4.18.2.1](#) for a description of the "best match" algorithm.

GetListContentTypesAndPropertiesResult.Properties: The container element for the set of returned properties.

GetListContentTypesAndPropertiesResult.Properties.Property: The container element for a single property. A property consists of either a key and value from the list or site property bag or of a file name and file contents from the list or site property bag.

GetListContentTypesAndPropertiesResult.Properties.Property.Key: When the property does not represent a file, the value MUST be the property key. When the property does represent a file, the value MUST be the file name.

GetListContentTypesAndPropertiesResult.Properties.Property.Value: When the property does not represent a file, the value MUST be the property value. When the property does represent a file, the value MUST be the file contents.

3.1.4.21 GetListItemChanges

This operation is used to retrieve the list items that have been inserted or updated after the specified date and time and that match the specified filter criteria. Protocol clients need to use **GetListItemChangesSinceToken** in place of this protocol server because this operation does not return list items that have been deleted on the protocol server.

```
<wsdl:operation name="GetListItemChanges">
  <wsdl:input message="GetListItemChangesSoapIn" />
  <wsdl:output message="GetListItemChangesSoapOut" />
</wsdl:operation>
```

The protocol client sends a **GetListItemChangesSoapIn** request message and the protocol server responds with a **GetListItemChangesSoapOut** response message, as follows:

1. If the specified **since** date time is an empty string or an invalid date-time string the protocol server MUST return a SOAP fault. There is no error code for this fault.
2. If the specified **listName** is an empty string the protocol server MUST return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
3. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
4. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
5. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [<47>](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
6. Otherwise, all list items in the list specified by the **listName** parameter that match the criteria based on the **since** and **contains** parameters MUST be returned, unless the protocol server encounters an unexpected situation and MUST return a SOAP fault.

3.1.4.21.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetListItemChangesSoapIn	The request to retrieve the list items that have been inserted or updated after the specified date and time and that match the specified filter criteria.
GetListItemChangesSoapOut	The response to a GetListItemsSoapIn request.

3.1.4.21.1.1 GetListItemChangesSoapIn

This message is the request of the **GetListItemChanges** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListItemChanges
```

The SOAP Body contains a **GetListItemChanges** element.

3.1.4.21.1.2 GetListItemChangesSoapOut

This message is the response of the **GetListItemChanges** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListItemChanges
```

The SOAP Body contains a **GetListItemChangesResponse** element.

3.1.4.21.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetListItemChanges	The request to retrieve the list items that have been inserted or updated after the specified date and time and that match the specified filter criteria.
GetListItemChangesResponse	Contains the response to a GetListItemsSoapIn request, as specified in section 3.1.4.21.1.1 .

3.1.4.21.2.1 GetListItemChanges

This element specifies details for the request that the protocol client sends to the protocol server to return list items matching specified or pre-defined criteria.

```
<s:element name="GetListItemChanges">  
  <s:complexType>  
    <s:sequence>  
      <s:element name="listName" type="s:string" />  
      <s:element name="viewFields" type="tns:CamlViewFields" minOccurs="0" />  
      <s:element name="since" type="s:string" minOccurs="0"/>  
      <s:element name="contains" type="tns:CamlContains" minOccurs="0" />  
    </s:sequence>  
  </s:complexType>  
</s:element>
```

listName : The GUID or the list title of the list from which to retrieve list items.

viewFields : Indicates which fields of the list item SHOULD be returned. If this is not specified, whether by excluding the element or by including an empty element, all fields MUST be returned from the list item. If the **Properties** attribute of the **ViewFields** structure is set to TRUE and the **MetaInfo** field is referenced in a **FieldRef** element, then metadata properties related to the list item MUST be returned with the prefix "ows_MetaInfo_". For examples of this, see the description of the **viewFields** parameter in section [3.1.4.24](#).

since : The date and time to start retrieving changes in the list. The protocol client can retrieve the value of this parameter from a prior **GetListItemChanges** response. If this value is null, the protocol server MUST return all list items in the list. If the date that is passed in is not in Coordinated Universal Time (UTC) format, it MUST be treated by the protocol server as being in the protocol server's local time zone and converted to UTC time by the protocol server before it searches for list items that have been modified after this time.

contains: Restricts the results returned by giving a specific value to be searched for in the specified list item field. The protocol server MUST return only those list items satisfying this criterion.

3.1.4.21.2.2 GetListItemChangesResponse

This element contains the protocol server response for the **GetListItemChanges** request.

```
<s:element name="GetListItemChangesResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="GetListItemChangesResult">
        <s:complexType>
          <s:sequence>
            <s:element name="listitems" >
              <s:complexType mixed="true" >
                <s:sequence>
                  <s:any />
                </s:sequence>
                <s:Attribute name="TimeStamp" type="s:string"/>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

GetListItemChangesResult:

This protocol server response included in the **listitems** element is modeled on the Microsoft ADO 2.6 Persistence format [\[MS-PRSTFR\]](#), excluding the **<s:schema>** element. The **listitems** element includes attributes describing the namespaces for the ADO 2.6 Persistence format, which MAY be ignored by the protocol client, as well as the **TimeStamp** attribute. **listitems** contains an inner element named **rs:data**, which is of type **DataDefinition**.

The **TimeStamp** attribute is a string that contains the date in Coordinated Universal Time (UTC) of the request to the protocol server. The protocol client MAY pass this **TimeStamp** back to the protocol server in a subsequent **GetListItemChanges** request as the **since** parameter to retrieve all inserts and updates that have occurred since the request that this **TimeStamp** represents.

There can be a maximum of two **rs:data** elements. The first **rs:data** element contains all the inserted and updated list items that have occurred subsequent to the specified **since** parameter. The second **rs:data** element contains all of the list items currently in the list. The names of the attributes containing the list item data in inner **z:row** elements correspond to the **Name** attribute in the **Field** elements of **GetList** and are prefixed by "ows_". Note that set of fields returned by the method is restricted by the **viewField** parameter.

An example of using this protocol is as follows.

The protocol client makes an initial request to get all rows and excludes the **since** parameter.

```
Content-Type: text/xml; charset=utf-8
SOAPAction: "http://schemas.microsoft.com/sharepoint/soap/GetListItemChanges"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetListItemChanges xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>TestListItem</listName>
    </GetListItemChanges>
  </soap:Body>
</soap:Envelope>
```

The response from the protocol server includes the **TimeStamp** attribute:

```
<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><GetListItemChangesResponse
xmlns="http://schemas.microsoft.com/sharepoint/soap/"><GetListItemChangesResult><listitems
xmlns:s="uuid:BDC6E3F0-6DA3-11d1-A2A3-00AA00C14882" xmlns:dt="uuid:C2F41010-65B3-11d1-A29F-
00AA00C14882" xmlns:rs="urn:schemas-microsoft-com:rowset" xmlns:z="#RowsetSchema"
TimeStamp="2008-02-08T18:18:18Z"><rs:data ItemCount="1"><z:row ows_Attachments="0"
ows_LinkTitle="test" ows_TextCol="test" ows_MetaInfo="1;#" ows_ModerationStatus="0"
ows_Level="1" ows_Title="test" ows_ID="1" ows_owshiddenversion="1"
ows_UniqueId="1;#{82BE0135-DB16-4C40-A4F2-30873DFA49A1}" ows_FSObjType="1;#0"
ows_Created_x0020_Date="1;#2008-02-08T18:12:47Z" ows_Created="2008-02-08T18:12:47Z"
ows_FileLeafRef="1;#1_.000" ows_FileRef="1;#Lists/TestListItem/1_.000" /></rs:data><rs:data
ItemCount="1"><z:row ows_ID="1" ows_MetaInfo="1;#" ows_ModerationStatus="0" ows_Level="1"
ows_owshiddenversion="1" ows_UniqueId="1;#{82BE0135-DB16-4C40-A4F2-30873DFA49A1}"
ows_FSObjType="1;#0" ows_Created="2008-02-08T18:12:47Z"
ows_FileRef="1;#Lists/TestListItem/1_.000"
/></rs:data></listitems></GetListItemChangesResult></GetListItemChangesResponse></soap:Body><
/soap:Envelope>
```

At a later time, the protocol client requests changes that have happened after the previous request by specifying the previously returned **TimeStamp** value as the value of the **since** parameter:

```
Content-Type: text/xml; charset=utf-8
```

```
SOAPAction: "http://schemas.microsoft.com/sharepoint/soap/GetListItemChanges"
```

```
<?xml version="1.0" encoding="utf-8"?>
```

```

<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetListItemChanges xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>TestListItem</listName>
      <since>2008-02-08T18:18:18Z</since>
    </GetListItemChanges>
  </soap:Body>
</soap:Envelope>

```

The protocol client receives the following response with a new **TimeStamp** and the items that have been inserted or updated:

```

<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><GetListItemChangesResponse
xmlns="http://schemas.microsoft.com/sharepoint/soap/"><GetListItemChangesResult><listitems
xmlns:s="uuid:BDC6E3F0-6DA3-11d1-A2A3-00AA00C14882" xmlns:dt="uuid:C2F41010-65B3-11d1-A29F-
00AA00C14882" xmlns:rs="urn:schemas-microsoft-com:rowset" xmlns:z="#RowsetSchema"
TimeStamp="2008-02-08T18:49:39Z"><rs:data ItemCount="2"><z:row ows_Attachments="0"
ows_LinkTitle="test2" ows_TextCol="test2" ows_MetaInfo="2;#" ows_ModerationStatus="0"
ows_Level="1" ows_Title="test2" ows_ID="2" ows_owshiddenversion="1"
ows_UniqueId="2;#{5F33D8A7-54D6-4610-800F-FCFC7A339376}" ows_FSObjType="2;#0"
ows_Created_x0020_Date="2;#2008-02-08T18:20:12Z" ows_Created="2008-02-08T18:20:12Z"
ows_FileLeafRef="2;#2_.000" ows_FileRef="2;#Lists/TestListItem/2_.000" /><z:row
ows_Attachments="0" ows_LinkTitle="test3" ows_TextCol="test3" ows_MetaInfo="3;#"
ows_ModerationStatus="0" ows_Level="1" ows_Title="test3" ows_ID="3"
ows_owshiddenversion="1" ows_UniqueId="3;#{B26DDAB9-EE4A-476B-B054-325E9B3936A8}"
ows_FSObjType="3;#0" ows_Created_x0020_Date="3;#2008-02-08T18:34:42Z" ows_Created="2008-02-
08T18:34:42Z" ows_FileLeafRef="3;#3_.000" ows_FileRef="3;#Lists/TestListItem/3_.000"
/></rs:data></listitems></GetListItemChangesResult></GetListItemChangesResponse></soap:Body><
/soap:Envelope>

```

A different example of using this protocol is as follows.

The protocol client decides to request items that have changed after a particular time where a specific field contains a specified value.

Content-Type: text/xml; charset=utf-8

SOAPAction: "http://schemas.microsoft.com/sharepoint/soap/GetListItemChanges"

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetListItemChanges xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>TestListItem</listName>
      <since>2008-02-08 10:00:00</since>
      <contains><Contains><FieldRef Name="TextColumn" /><Value
Type="Text">SearchingForValue</Value></Contains></contains>

```

```

    </GetListItemChanges>
  </soap:Body>
</soap:Envelope>

```

3.1.4.22 GetListItemChangesSinceToken

This operation [48](#) is used to return changes made to a specified list after the event expressed by the change token, if specified, or to return all the list items in the list.

```

<wsdl:operation name="GetListItemChangesSinceToken">
  <wsdl:input message="GetListItemChangesSinceTokenSoapIn" />
  <wsdl:output message="GetListItemChangesSinceTokenSoapOut" />
</wsdl:operation>

```

The protocol client sends a **GetListItemChangesSinceTokenSoapIn** request message and the protocol server responds with a **GetListItemChangesSinceTokenSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the List title of a list on the site and if so, use that list.
3. If **listName** does not correspond to a list from either of these checks, the protocol server **MUST** return a SOAP fault with error code 0x8200006. This indicates that the list does not exist or might have been deleted by another user.
4. If the **viewFields** parameter is not specified and the **viewName** parameter contains a valid GUID but refers to a view that does not exist, the protocol server **MUST** return a SOAP fault. If the **viewFields** parameter is not specified and the **viewName** parameter contains an invalid GUID, the server **MUST** ignore the **viewName** parameter. Otherwise, if the **viewFields** parameter is specified, the server **MUST** ignore the **viewName** parameter.
5. If the value of the **changeToken** parameter is empty, the protocol server **SHOULD** ignore the **changeToken** parameter [49](#). If the value of the **changeToken** parameter does not match a change token known to the protocol server, the protocol server **MUST** return a SOAP fault. There is no error code for this fault.
6. Otherwise, all list items specified by the **listName** parameter that match the criteria passed in **MUST** be returned, unless the protocol server encounters an unexpected situation and **MUST** return a SOAP fault.

3.1.4.22.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetListItemChangesSinceTokenSoapIn	The request for changes made to a specified list after the event expressed by the change token, if specified, or to return all the list items in the list.

Message	Description
GetListItemChangesSinceTokenSoapOut	The response to a GetListItemChangesSinceTokenSoapIn request.

3.1.4.22.1.1 GetListItemChangesSinceTokenSoapIn

This message is the request of the **GetListItemChangesSinceToken** operation that the protocol client sends to the protocol server.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListItemChangesSinceToken
```

The SOAP Body contains a **GetListItemChangesSinceToken** element.

3.1.4.22.1.2 GetListItemChangesSinceTokenSoapOut

This message is the response of the **GetListItemChangesSinceToken** operation that the protocol client receives from the protocol server.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListItemChangesSinceToken
```

The SOAP Body contains a **GetListItemChangesSinceTokenResponse** element.

3.1.4.22.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetListItemChangesSinceToken	The request for changes made to a specified list after the event expressed by the change token, if specified, or to return all the list items in the list.
GetListItemChangesSinceTokenResponse	Contains the response to a GetListItemChangesSinceTokenSoapIn request, as specified in section 3.1.4.22.1.1 .

3.1.4.22.2.1 GetListItemChangesSinceToken

This element specifies details for the request that the protocol client sends to the protocol server to return list items matching certain criteria.

```
<s:element name="GetListItemChangesSinceToken">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string"/>
      <s:element name="viewName" type="s:string" minOccurs="0" maxOccurs="1"/>
      <s:element name="query" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

```

    <s:complexType>
      <s:sequence>
        <s:element name="Query" type="core:CamlQueryRoot" />
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:element name="viewFields" type="tns:CamlViewFields" minOccurs="0" maxOccurs="1"/>
  <s:element name="rowLimit" type="s:string" minOccurs="0" maxOccurs="1"/>
  <s:element name="queryOptions" type="tns:CamlQueryOptions"
    minOccurs="0" maxOccurs="1" />
  <s:element name="changeToken" type="s:string" minOccurs="0" maxOccurs="1" />
  <s:element name="contains" type="tns:CamlContains" minOccurs="0" maxOccurs="1" />
</s:sequence>
</s:complexType>
</s:element>

```

listName : The GUID or the list title of the list from which to retrieve list items. If the specified list is the user information list, the string is set to "UserInfo" or "User Information list".

viewName : Protocol clients SHOULD NOT include this parameter. If specified, the protocol server MUST ignore this parameter if the **viewFields** parameter is also specified; otherwise, the server MUST only validate that this parameter is both a valid GUID and that the parameter refers to a view of the list.

query : See the description of the **query** element in section [3.1.4.24](#).

viewFields : See the description of the **viewFields** element in section [3.1.4.24](#).

rowLimit : SHOULD be specified to indicate the maximum number of rows of data to return in the response. If specified, the parameter MUST contain the string representation of an unsigned 32-bit integer. Not specifying a value or specifying a value of 0 indicates that all rows SHOULD be retrieved, unless the **changeToken** parameter is specified. (see description of **changeToken** parameter for limitations of the **rowLimit**). The protocol server MUST NOT return more inserted or updated list items than this parameter specifies. Deleted items are not subject to this limitation.

queryOptions : See the description of the **queryOptions** element in section [3.1.4.24](#).

If the **queryOptions** parameter is excluded, the default values that the protocol server MUST use are the following:

- **RecurrenceOrderBy** – True
- **ViewAttributes** – Scope="RecursiveAll"
- **DateInUtc** – True
- **IncludePermissions** – True
- **IncludeAttachmentUrls** – True
- **IncludeAttachmentVersion** – True
- **RecurrencePatternXMLVersion** – v3
- **ExpandUserField** – True
- **MeetingInstanceID** – -1

The default values for **RecurrenceOrderBy** and **RecurrencePatternXMLVersion** MUST NOT affect the response if **RecurrenceData** field is not part of the list item. Additionally, even when **RecurrenceData** is part of the list item, the protocol client MUST specify the following **viewFields** parameter to see the effect of the default values: <ViewFields /> or MUST contain <FieldRef Name="RecurrenceData"/> in **viewFields**.

changeToken : If excluded by the protocol client all items in the list MUST be returned, subject to the default row limit of the view or the overriding **rowLimit** parameter. If specified, it MUST be a **Changes** element with a **LastChangeToken** attribute that is assigned a string comprising a token returned by a previous call to **GetListItemChangesSinceToken**. The response will include changes made to list items after the change token was retrieved, subject to the row limit restrictions just described.

An example of the type of token that MAY be returned by a call to **GetListItemChangesSinceToken** is as follows:

```
<Changes LastChangeToken="1;3;b0c07dce-e4c7-45a9-8b7f-0d5339f69633;633247934080930000;70853"></Changes>
```

When the change token is excluded, protocol clients MAY page data from the protocol server. In this case, protocol clients MUST exclude the **changeToken** parameter for each request until the data is fully fetched by using the **ListItemCollectionPositionNext** paging mechanism (see section [2.2.4.4](#)).

If the **changeToken** parameter is specified and the total number of list items that have been inserted or updated is greater than 100, then only the first 100 items that changed are returned <50>. The **Changes** element in the response MUST include the attribute **MoreChanges** set to "TRUE" to indicate that protocol clients MAY make another call to **GetListItemChangesSinceToken** by using the new change token from the most recent response to retrieve subsequent changes.

contains: Specifies a value to search for. This SHOULD NOT be specified when the **query** parameter is used, but if both are specified, **contains** MUST be ignored by the protocol server.

The following example shows a request that uses the change token extracted from a previous **GetListItemChangesSinceTokenResponse**:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
<GetListItemChangesSinceToken xmlns="http://schemas.microsoft.com/sharepoint/soap/">
<listName>TestListItem</listName>
<query><Query><OrderBy><FieldRef Name="ID"/></OrderBy></Query></query>
<viewFields><ViewFields><FieldRef Name="ID" /><FieldRef
Name="Title" /></ViewFields></viewFields>
<rowLimit>10</rowLimit>
<changeToken> 1;3;e074e60a-0126-4446-a751-a5be7a55ed81;633374187828130000;243</changeToken>
</GetListItemChangesSinceToken>
</soap:Body>
</soap:Envelope>
```

3.1.4.22.2.2 GetListItemChangesSinceTokenResponse

This element contains the protocol server response for the **GetListItemChangesSinceToken** request.

```
<s:element name="GetListItemChangesSinceTokenResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="GetListItemChangesSinceTokenResult">
        <s:complexType>
          <s:sequence>
            <s:element name="listitems" >
              <s:complexType>
                <s:sequence>
                  <s:element name="Changes" >
                    <s:complexType>
                      <s:sequence>
                        <s:element name="List" type="tns:ListDefinitionSchema"
                          minOccurs="0" />
                        <s:element name="Id" type="tns:ListItemChangeDefinition"
                          minOccurs="0"/>
                      </s:sequence>
                    </s:complexType>
                  </s:element>
                  <s:attribute name="LastChangeToken" type="s:string" />
                  <s:attribute name="MoreChanges" type="core:TRUEFALSE" />
                  <s:attribute name="MinTimeBetweenSyncs"
                    type="s:unsignedInt" />
                  <s:attribute name="RecommendedTimeBetweenSyncs"
                    type="s:unsignedInt" />
                  <s:attribute name="MaxBulkDocumentSyncSize"
                    type="s:unsignedInt" />
                  <s:attribute name="AlternateUrls" type="s:string" />
                  <s:attribute name="EffectivePermMask" type="s:string" />
                </s:complexType>
              </s:element>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

GetListItemChangesSinceTokenResult: The top-level element, which contains a **listitems** element. **listitems** includes attributes describing the namespaces for the ADO 2.6 Persistence format, which MAY be ignored by the protocol client, as well as the **Changes** element. The following attributes are specified for **Changes**.

Attribute	Definition
MinTimeBetweenSyncs	Represents the minimum amount of time between user-initiated

Attribute	Definition
	or automatic synchronization calls that is recommended by the server. The value represents a time, in minutes. Protocol clients can respect this value even if the user initiates synchronization manually. That is, if this is set to 5 minutes, protocol clients MAY only send one request per 5 minutes even if the user repeatedly clicks "send/receive".
RecommendedTimeBetweenSynics	The recommended minimum amount of time between synchronizations. The clients can choose to respect this for automatic syncs and override it for user-initiated synchronizations.
MaxBulkDocumentSyncSize	The maximum total size of content that is recommended to be synchronized to the protocol client. The default is 500 megabytes. Setting this to a large value can degrade performance. See [MS-OUTSPS] section 3.2.4.2.6.
AlternateUrls	Alternate URLs are listed in the following zone order, delimited by commas: Intranet,Default,Extranet,Internet,Custom
EffectivePermMask	The permissions on the list. See [MS-WSSFO2] section 2.2.2.14.
LastChangeToken	Protocol clients SHOULD include the value of this attribute in subsequent requests to the protocol server. See notes in the changeToken parameter description for more information about paging of data with change tokens.
MoreChanges	The Changes element in the response MUST include the attribute MoreChanges set to "TRUE" if more changes are known to the protocol server than were returned to the protocol client. When the protocol client finds MoreChanges set to "TRUE", the protocol client can make another call to GetListItemChangesSinceToken by using the new change token value of LastChangeToken to retrieve subsequent changes. MoreChanges MUST be included in the result only if the changeToken parameter was specified by the protocol client in the request.

The inner XML of the **Changes** element MUST contain the full **List** element from **GetList** if no change token was passed in the request or if there have been schema changes made to the list after the change token. [<51>](#) The names of the attributes containing the list item data in inner **z:row** elements correspond to the **Name** attribute in the **Field** elements of **GetList** and are prefixed by "ows_". Note that set of fields returned by the method is restricted by the **viewField** or **viewName** parameter.

The **listitems** element also contains an inner element named **rs:data**, which is of type **DataDefinition**.

The following is an example response from a **GetListItemChangesSinceToken** request:

```
<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><GetListItemChangesSinceTokenResponse
xmlns="http://schemas.microsoft.com/sharepoint/soap/"><GetListItemChangesSinceTokenResult><li
stitems MinTimeBetweenSynics='0' RecommendedTimeBetweenSynics='180'
MaxBulkDocumentSyncSize='500' AlternateUrls='http://siteaddress/'
EffectivePermMask='FullMask' xmlns:s='uuid:BDC6E3F0-6DA3-11d1-A2A3-00AA00C14882'
```

```

xmlns:dt='uuid:C2F41010-65B3-11d1-A29F-00AA00C14882'
xmlns:rs='urn:schemas-microsoft-com:rowset'
xmlns:z='#RowsetSchema'>
<Changes LastChangeToken="1;3;e074e60a-0126-4446-a751-
a5be7a55ed81;633374187828130000;243"><List ... excluded content that matches GetList call ... >
</List></Changes>
<rs:data ItemCount="1">
  <z:row ows_ID='1' ows_Title='Test new entry' ows_MetaInfo='1;#' ows_ModerationStatus='0'
ows_Level='1' ows_owshiddenversion='3' ows_UniqueId='1;#{078CAB5C-5FEA-43EA-B23A-
FB93EF35397D}' ows_FSObjType='1;#0' ows_Created='2008-01-28 08:06:11'
ows_FileRef='1;#Lists/TestListItem/1_.000' />

```

3.1.4.23 GetListItemChangesWithKnowledge

This operation is used to return changes made to a specified list after the event expressed by the **knowledge** parameter, if specified, or to return all the list items in the list. [<52>](#)

```

<wsdl:operation name="GetListItemChangesWithKnowledge">
  <wsdl:input message="GetListItemChangesWithKnowledgeSoapIn" />
  <wsdl:output message="GetListItemChangesWithKnowledgeSoapOut" />
</wsdl:operation>

```

The protocol client sends a **GetListItemChangesWithKnowledgeSoapIn** request message and the protocol server responds with a **GetListItemChangesWithKnowledgeSoapOut** response message, as follows:

- If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
- If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the listName corresponds to the List title of a list on the site and if so, use that list.
- If **listName** does not correspond to a list from either of these checks, the protocol server MUST return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
- If the **viewFields** parameter is not specified and the **viewName** parameter contains a correctly formatted GUID and refers to a view that does not exist, the protocol server MUST return a SOAP fault with error code 0x82000005. If the **viewFields** parameter is not specified and the **viewName** parameter does not contain a correctly formatted GUID, or if the **viewFields** parameter is specified, the server MUST ignore the **viewName** parameter.
- If the value of the knowledge parameter is empty or is not of a valid format, the protocol server MUST return a SOAP fault. There is no error code for this fault.
- Otherwise, all list items specified by the **listName** parameter that match the criteria passed in MUST be returned, unless the protocol server encounters an unexpected situation and MUST return a SOAP fault. There is no specific error code for this fault.

3.1.4.23.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetListItemChangesWithKnowledgeSoapIn	The request to return changes made to a specified list after the event expressed by the knowledge parameter, if specified, or to return all the list items in the list.
GetListItemChangesWithKnowledgeSoapOut	The response to a GetListItemChangesWithKnowledgeSoapIn request.

3.1.4.23.1.1 GetListItemChangeswithKnowledgeSoapIn

This message is the request of the **GetListItemChangesWithKnowledge** operation that the protocol client sends to the protocol server.

The SOAP Action value of the message is defined as:

```
http://schemas.microsoft.com/sharepoint/soap/GetListItemChangesWithKnowledge
```

The SOAP Body contains a **GetListItemChangesWithKnowledge** element.

3.1.4.23.1.2 GetListItemChangeswithKnowledgeSoapOut

This message is the response of the **GetListItemChangesWithKnowledge** operation that the protocol client receives from the protocol server.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetListItemChangesWithKnowledge
```

The SOAP Body contains a **GetListItemChangesWithKnowledgeResponse** element.

3.1.4.23.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetListItemChangesWithKnowledge	The request to return changes made to a specified list after the event expressed by the knowledge parameter, if specified, or to return all the list items in the list.
GetListItemChangesWithKnowledgeResponse	The response to a GetListItemChangesWithKnowledgeSoapIn request, as specified in section 3.1.4.23.1.1 .

3.1.4.23.2.1 GetListItemChangesWithKnowledge

This element specifies details for the request that the protocol client sends to the protocol server to return list items matching certain criteria.

```

<s:element name="GetListItemChangesWithKnowledge">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string"/>
      <s:element name="viewName" type="s:string" minOccurs="0" maxOccurs="1"/>
      <s:element name="query" minOccurs="0">
        <s:complexType>
          <s:sequence>
            <s:element name="Query" type="core:CamlQueryRoot" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="viewFields" type="tns:CamlViewFields" minOccurs="0" maxOccurs="1"/>
      <s:element name="rowLimit" type="s:string" minOccurs="0" maxOccurs="1"/>
      <s:element name="queryOptions" type="tns:CamlQueryOptions"
        minOccurs="0" maxOccurs="1" />
      <s:element name="syncScope" minOccurs="0" maxOccurs="1" type="s:string" />
      <s:element name="knowledge" minOccurs="0" maxOccurs="1" type="sync:syncKnowledge" />
      <s:element name="contains" type="tns:CamlContains" minOccurs="0" maxOccurs="1" />
    </s:sequence>
  </s:complexType>
</s:element>

```

listName: See the description of the **listName** element in **GetListChangesSinceToken** in section [3.1.4.22.2.1](#).

viewName: See the description of the **viewName** element in **GetListChangesSinceToken** in section [3.1.4.22.2.1](#).

query: See the description of the **query** element in **GetListChangesSinceToken** in section [3.1.4.22.2.1](#).

viewFields: See the description of the **viewFields** element in **GetListChangesSinceToken** in section [3.1.4.22.2.1](#).

rowLimit: SHOULD be specified to indicate the maximum number of rows of data to return in the response. If specified, the parameter MUST contain the string representation of an unsigned 32-bit integer. Excluding this parameter, passing an empty element, or specifying a value of 0 indicates that all rows SHOULD be retrieved, unless the **knowledge** parameter is specified. (see description of **knowledge** parameter for limitations of the **rowLimit**). The protocol server MUST NOT return more inserted or updated list items than this parameter specifies. Deleted items are not subject to this limitation.

queryOptions: See the description of the **queryOptions** element in **GetListChangesSinceToken** in section [3.1.4.22.2.1](#).

syncScope: This parameter MUST be null or empty.

knowledge: This parameter specifies the knowledge data structure in XML format as specified in [\[MS-XSSK\]](#) section 2.1.3. If excluded by the protocol client, all items in the list MUST be returned, subject to the default row limit of the view or the overriding **rowLimit** parameter. If specified, the response will include changes made to list items after the knowledge data structure was retrieved, subject to the row limit restrictions just described.

An example of the type of knowledge that MAY be returned by a call to **GetListItemChangesWithKnowledge** is as follows:

```

<sync:syncKnowledge xmlns="http://schemas.microsoft.com/2008/03/sync/"
xmlns:sync="http://schemas.microsoft.com/2008/03/sync/"><idFormatGroup><replicaIdFormat
sync:isVariable="false" sync:maxLength="16" /><itemIdFormat sync:isVariable="false"
sync:maxLength="16" /><changeUnitIdFormat sync:isVariable="false" sync:maxLength="1"
/></idFormatGroup><replicaKeyMap><replicaKeyMapEntry
sync:replicaId="udKODRe/Rtu784eyc03pnQ==" sync:replicaKey="0"
/></replicaKeyMap><clockVector><clockVectorElement sync:replicaKey="0" sync:tickCount="341"
/></clockVector></sync:syncKnowledge>

```

When the **knowledge** parameter is excluded, protocol clients can page data from the protocol server. In this case, protocol clients MUST exclude the **knowledge** parameter for each request until the data is fully fetched by using the **ListItemCollectionPositionNext** paging mechanism (see section [2.2.4.4](#)).

If the **knowledge** parameter is specified and the total number of list items that have been inserted or updated is greater than 100, then only the first 100 items that changed are returned, subject to the default row limit of the view or the overriding **rowLimit** parameter. The **Changes** element in the response MUST include the attribute **MoreChanges** set to "TRUE" to indicate that protocol clients can make another call to **GetListItemChangesWithKnowledge** by using the new knowledge from the most recent response to retrieve subsequent changes.

contains: See the description of the **contains** element in **GetListChangesSinceToken** in section [3.1.4.22.2.1](#).

The following example shows a request that uses the knowledge extracted from a previous **GetListItemChangesWithKnowledgeResponse** response:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
<GetListItemChangesWithKnowledge xmlns="http://schemas.microsoft.com/sharepoint/soap/">
<listName>TestListItem</listName>
<query><Query><OrderBy><FieldRef Name="ID"/></OrderBy></Query></query>
<viewFields><ViewFields><FieldRef Name="ID" /><FieldRef
Name="Title" /></ViewFields></viewFields>
<rowLimit>10</rowLimit>
<knowledge> <sync:syncKnowledge xmlns="http://schemas.microsoft.com/2008/03/sync/"
xmlns:sync="http://schemas.microsoft.com/2008/03/sync/"><idFormatGroup><replicaIdFormat
sync:isVariable="false" sync:maxLength="16" /><itemIdFormat sync:isVariable="false"
sync:maxLength="16" /><changeUnitIdFormat sync:isVariable="false" sync:maxLength="1"
/></idFormatGroup><replicaKeyMap><replicaKeyMapEntry
sync:replicaId="udKODRe/Rtu784eyc03pnQ==" sync:replicaKey="0"
/></replicaKeyMap><clockVector><clockVectorElement sync:replicaKey="0" sync:tickCount="341"
/></clockVector></sync:syncKnowledge></knowledge>
</GetListItemChangesWithKnowledge>
</soap:Body>
</soap:Envelope>

```

3.1.4.23.2.2 GetListItemChangesWithKnowledgeResponse

This element contains the protocol server response for the **GetListItemChangesWithKnowledge** request.

```

<s:element name="GetListItemChangesWithKnowledgeResponse">
<s:complexType>

```

```

<s:sequence>
  <s:element name="GetListItemChangesWithKnowledgeResult">
    <s:complexType>
      <s:sequence>
        <s:element name="listitems" >
          <s:complexType>
            <s:sequence>
              <s:element name="Changes" >
                <s:complexType>
                  <s:sequence>
                    <s:element name="MadeWithKnowledge" minOccurs="0" maxOccurs="1">
                      <s:complexType>
                        <s:sequence>
                          <s:element name="syncKnowledge" type="sync:syncKnowledge" />
                        </s:sequence>
                      </s:complexType>
                    </s:element>
                    <s:element name="Id" type="tns:ListItemChangeDefinition"
minOccurs="0"/>
                    <s:element name="View" type="tns:ViewChangeDefinition"
minOccurs="0"/>
                  </s:sequence>
                  <s:attribute name="SchemaChanged" type="core:TRUEFALSE" />
                  <s:attribute name="ServerTime" type="s:string" />
                  <s:attribute name="MoreChanges" type="core:TRUEFALSE" />
                  <s:attribute name="MinTimeBetweenSyncs"
type="s:unsignedInt" />
                  <s:attribute name="RecommendedTimeBetweenSyncs"
type="s:unsignedInt" />
                  <s:attribute name="MaxBulkDocumentSyncSize"
type="s:unsignedInt" />
                  <s:attribute name="AlternateUrls" type="s:string" />
                  <s:attribute name="EffectivePermMask" type="s:string" />
                </s:complexType>
              </s:element>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>

```

GetListItemChangesWithKnowledgeResult:

The top-level element, which contains a **listitems** element. **listitems** includes attributes describing the namespaces for the ADO 2.6 Persistence format, which can be ignored by the protocol client, as well as the **Changes** element. The following attributes are specified for **Changes**.

Attribute	Definition
MinTimeBetweenSyncs	See the description of the MinTimeBetweenSyncs attribute in GetListItemChangesSinceTokenResponse in section

Attribute	Definition
	3.1.4.22.2.2.
RecommendedTimeBetweenSyncs	See the description of the RecommendedTimeBetweenSyncs attribute in GetListItemChangesSinceTokenResponse in section 3.1.4.22.2.2.
MaxBulkDocumentSyncSize	See the description of the MaxBulkDocumentSyncSize attribute in GetListItemChangesSinceTokenResponse in section 3.1.4.22.2.2.
AlternateUrls	See the description of the AlternateUrls attribute in GetListItemChangesSinceTokenResponse in section 3.1.4.22.2.2.
EffectivePermMask	See the description of the EffectivePermMask attribute in GetListItemChangesSinceTokenResponse in section 3.1.4.22.2.2.
MoreChanges	The Changes element in the response MUST include the attribute MoreChanges set to "TRUE" if more changes are known to the protocol server than were returned to the protocol client. When the protocol client finds MoreChanges set to "TRUE", the protocol client can make another call to GetListItemChangesWithKnowledge by using the new knowledge value of MadeWithKnowledge to retrieve subsequent changes. MoreChanges MUST be included in the result only if the knowledge parameter was specified by the protocol client in the request.
SchemaChanged	The SchemaChanged attribute MUST be set to "TRUE" if there have schema changes made to the list.
ServerTime	The ServerTime attribute specifies the UTC date and time in the Gregorian calendar when the changes were returned by the server in the format "yyyyMMdd hh:mm:ss" where "yyyy" represents the year, "MM" represents the month, "dd" represents the day of the month, "hh" represents the hour, "mm" represents the minute, and "ss" represents the second. This attribute MUST NOT be set if there no changes are returned.

The inner XML of the **MadeWithKnowledge** element in the **Changes** element is the knowledge in XML format, as specified in [\[MS-XSSK\]](#) section 3, that represents the last change in the list that is returned to the client. See notes in the **knowledge** parameter description for more information about paging of data with knowledge.

The names of the attributes containing the list item data in inner **z:row** elements correspond to the **Name** attribute in the **Field** elements of **GetList** and are prefixed by "ows_". Note that set of fields returned by the method is restricted by the **viewField** or **viewName** parameter. If the protocol client requests the **MetaInfo** field for the list items, the protocol server MUST return the **vti_contentchangeunit Metainfo** property if the property was set in the item or document and the list item or document has not been updated since the time the property was set by a client.

The **listitems** element also contains an inner element named **rs:data**, which is of type **DataDefinition**.

The following is an example response from a **GetListItemChangesWithKnowledge** request:

```

<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><GetListItemChangesWithKnowledgeRespo
nse
xmlns="http://schemas.microsoft.com/sharepoint/soap/"><GetListItemChangesWithKnowledgeResult>
<listitems MinTimeBetweenSyncs='0' RecommendedTimeBetweenSyncs='180'
MaxBulkDocumentSyncSize='500' AlternateUrls='http://siteaddress/'
EffectivePermMask='FullMask' xmlns:s='uuid:BDC6E3F0-6DA3-11d1-A2A3-00AA00C14882'
xmlns:dt='uuid:C2F41010-65B3-11d1-A29F-00AA00C14882'
xmlns:rs='urn:schemas-microsoft-com:rowset'
xmlns:z='#RowsetSchema'>
<Changes><MadeWithKnowledge><sync:syncKnowledge
xmlns="http://schemas.microsoft.com/2008/03/sync/"
xmlns:sync="http://schemas.microsoft.com/2008/03/sync/"><idFormatGroup><replicaIdFormat
sync:isVariable="false" sync:maxLength="16" /><itemIdFormat sync:isVariable="false"
sync:maxLength="16" /><changeUnitIdFormat sync:isVariable="false" sync:maxLength="1"
/></idFormatGroup><replicaKeyMap><replicaKeyMapEntry
sync:replicaId="udKODRe/Rtu784eyc03pnQ==" sync:replicaKey="0"
/></replicaKeyMap><clockVector><clockVectorElement sync:replicaKey="0" sync:tickCount="341"
/></clockVector></sync:syncKnowledge></MadeWithKnowledge></Changes>
<rs:data ItemCount="1">
  <z:row ows_ID='1' ows_Title='Test new entry' ows_MetaInfo='1;#' ows_ModerationStatus='0'
ows_Level='1' ows_owshiddenversion='3' ows_UniqueId='1;#{078CAB5C-5FEA-43EA-B23A-
FB93EF35397D}' ows_FSObjType='1;#0' ows_Created='2008-01-28 08:06:11'
ows_FileRef='1;#Lists/TestListItem/1_.000' />

```

3.1.4.24 GetListItems

This operation is used to retrieve details about list items in a list that satisfy specified criteria.

```

<wsdl:operation name="GetListItems">
  <wsdl:input message="GetListItemsSoapIn" />
  <wsdl:output message="GetListItemsSoapOut" />
</wsdl:operation>

```

The protocol client sends a **GetListItemsSoapIn** request message and the protocol server responds with a **GetListItemsSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
3. If **listName** is not empty and does not correspond to a list from either of these checks, the protocol server SHOULD [53](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or MUST have been deleted by another user.
4. If **listName** and **viewName** are both empty, the protocol server MUST return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
5. If the **viewName** parameter is not a GUID, the protocol server MUST return a SOAP fault with error code 0x82000001. This indicates that the **viewName** parameter is missing or invalid. If the **viewName** parameter does not correspond to a view that exists and each of the optional **query**,

viewFields, **queryOptions**, and **rowLimit** parameters are not specified, the protocol server MUST return a SOAP fault. There is no error code for this fault. Otherwise, if each of the **query**, **viewFields**, **queryOptions**, and **rowLimit** parameters are specified, the protocol server MUST ignore the **viewName** parameter.

6. Otherwise, all list items in the list matching the specified criteria MUST be returned, unless the protocol server encounters an unexpected situation and MUST return a SOAP fault. There is no specific error code for this fault.

3.1.4.24.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetListItemsSoapIn	The request to retrieve details about list items in a list that satisfy specified criteria.
GetListItemsSoapOut	The response to a request to retrieve details about list items in a list that satisfy specified criteria.

3.1.4.24.1.1 GetListItemsSoapIn

This message is the request of the **GetListItemsSoapIn** operation that the protocol client sends as a request to this Web protocol server.

The SOAP Action value of the message is defined as:

`http://schemas.microsoft.com/sharepoint/soap/GetListItems`

The SOAP Body contains a **GetListItems** element.

3.1.4.24.1.2 GetListItemsSoapOut

This message is the response of the **GetListItems** operation that the protocol server sends back to a request from the protocol client.

The SOAP Action value of the message is defined as follows:

`http://schemas.microsoft.com/sharepoint/soap/GetListItems`

The SOAP Body contains a **GetListItemsResponse** element.

3.1.4.24.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetListItems	The request to retrieve details about list items in a list that satisfy specified criteria.

Element	Description
GetListItemsResponse	Contains the response to a request to retrieve details about list items in a list that satisfy specified criteria.

3.1.4.24.2.1 GetListItems

This message is the request of the **GetListItems** operation that the protocol client sends as a request to this protocol server.

```
<s:element name="GetListItems">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string"/>
      <s:element name="viewName" type="s:string" minOccurs="0"/>
      <s:element name="query" minOccurs="0">
        <s:complexType>
          <s:sequence>
            <s:element name="Query" type="core:CamlQueryRoot"/>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="viewFields" type="tns:CamlViewFields" minOccurs="0"/>
      <s:element name="rowLimit" type="s:string" minOccurs="0"/>
      <s:element name="queryOptions" type="tns:CamlQueryOptions" minOccurs="0" />
      <s:element name="webID" type="s:string" minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

listName: The GUID or the list title of the list from which to retrieve list items. If the specified list is the user information list, the string is set to "UserInfo" or "User Information List".

viewName: The GUID of the view of the list from which to return list items (see [\[MS-VIEWS\]](#)). If not specified, the default list view is used. **viewFields**, **rowLimit**, and **queryOptions** parameters, if any are specified, **MUST** override corresponding attributes of the view used.

query: The query to determine which records from the list are to be returned and the order in which they will be returned. See section 2.2 in [\[MS-WSSCAML\]](#).

viewFields: Specifies which fields of the list item **SHOULD** be returned. If this is not specified, whether by excluding the element or by including an empty element, the fields defined for the **viewName** parameter view will be used. If the **Properties** attribute of the **ViewFields** structure is set to **TRUE** and the **MetaInfo** field is referenced in a **FieldRef** element, then metadata properties related to the list item **MUST** be returned with the prefix "ows_MetaInfo_".

Example snippet of request for properties:

```
<ViewFields Properties="True"><FieldRef Name="MetaInfo" />...</ViewFields>
```

Example snippet of response:

```
<z:row ows_ID='1' ... ows_MetaInfo_vti_versionhistory='e074e60a01264446a751a5be7a55ed81:2' />
```

The preceding example shows that a **vti_versionhistory** property is set on the list item.

Alternatively, specific properties MAY be requested by using the following format, where the name of the specified property is substituted for "PropertyName":

```
<ViewFields Properties="True"><FieldRef Name="MetaInfo" Property="PropertyName"
/>...</ViewFields>
```

rowLimit: Specifies the maximum number of rows of data to return in the response. If this is not specified, whether by passing an empty element or by excluding the element, the default row limit for the view specified by the **viewName** parameter is used. Otherwise, this MUST be a 32-bit unsigned integer. The protocol server MUST NOT return more list items than this parameter specifies.

queryOptions: Specifies various options for modifying the query.

webID: The GUID of the site that contains the list. If not specified, the default Web site based on the SOAP request is used. The protocol client SHOULD exclude this parameter from the request.

The following is a sample request that retrieves list items from the list "TestItem" by using the default view, ordered by the identification field. In addition to the service fields that are always returned, the example returns the **ID**, **Attachments**, **LinkTitle**, **Num**, **Text**, and **FileDirRef** fields. The example returns a maximum of 10 rows, dates in UTC format, and a flat view of the list (that is not scoped to a specific folder).

```
Content-Type: text/xml; charset=utf-8
SOAPAction: "http://schemas.microsoft.com/sharepoint/soap/GetListItems"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
<GetListItems xmlns="http://schemas.microsoft.com/sharepoint/soap/">
<listName>TestItem</listName>
<viewName></viewName>
<query><Query><OrderBy><FieldRef Name="ID"/></OrderBy></Query></query>
<viewFields><ViewFields><FieldRef Name="ID" /><FieldRef
Name="Attachments"/><FieldRef Name="LinkTitle"/><FieldRef Name="Num"/><FieldRef
Name="Text"/><FieldRef Name="FileDirRef" /></ViewFields></viewFields>
<rowLimit>10</rowLimit>
<queryOptions><QueryOptions><DateInUtc>True</DateInUtc><IncludeMandatoryColumns>False</Includ
eMandatoryColumns><ViewAttributes Scope="RecursiveAll"/></QueryOptions></queryOptions>
</GetListItems>
</soap:Body>
</soap:Envelope>
```

3.1.4.24.2.2 GetListItemsResponse

This element contains the protocol server response for the **GetListItems** request.

```
<s:element name="GetListItemsResponse">
<s:complexType>
<s:sequence>
<s:element name="GetListItemsResult">
<s:complexType>
```

```

    <s:sequence>
      <s:element name="listitems" >
        <s:complexType mixed="true" >
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
</s:sequence>
<s:anyAttribute />
</s:complexType>
</s:element>

```

GetListItemsResult: This protocol server response included in the **listitems** element is modeled on the Microsoft ADO 2.6 Persistence format [\[MS-PRSTFR\]](#), excluding the **<s:schema>** element. The **listitems** element includes attributes describing the namespaces for the ADO 2.6 Persistence format, which MAY be ignored by the protocol client, and contains an inner element named **rs:data**, which is of type **DataDefinition**. Protocol clients SHOULD make a **GetList** call to get the schema of the list. The names of the attributes containing the list item data in this response correspond to the **Name** attribute in the **Field** elements of **GetList** and are prefixed by "ows_". Note that set of fields returned by the method is restricted by the **viewField** or **viewName** parameter.

The following is an example response:

```

<GetListItemsResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
  <GetListItemsResult>
    <listitems xmlns:s='uuid:BDC6E3F0-6DA3-11d1-A2A3-00AA00C14882' xmlns:dt='uuid:C2F41010-65B3-11d1-A29F-00AA00C14882' xmlns:rs='urn:schemas-microsoft-com:rowset' xmlns:z='#RowsetSchema'>
      <rs:data ItemCount="1" ListItemCollectionPositionNext="Paged=TRUE&amp;p_ID=11">
        <z:row ows_Attachments='0' ows_LinkTitle='TestItem' ows_Num='20.00000000000000'
          ows_Text='TestItem' ows_FileDirRef='1;#Lists/TestItem' ows_FSObjType='1;#0'
          ows_MetaInfo='1;#' ows_ModerationStatus='0' ows_Level='1' ows_Title='Test new entry'
          ows_ID='1' ows_owshiddenversion='1' ows_UniqueId='1;#{078CAB5C-5FEA-43EA-B23A-FB93EF35397D}'
          ows_Created_x0020_Date='1;#2008-01-28 08:06:11' ows_Created='2008-01-28 08:06:11'
          ows_FileLeafRef='1;#1_.000' ows_FileRef='1;#Lists/TestListItem/1_.000' />
        </rs:data>
      </listitems>
    </GetListItemsResult>
  </GetListItemsResponse>

```

The **listitems** element contains attributes that define the namespaces. Inside of this element is the **<rs:data>** element, which specifies how many rows of data are being returned, where a row of data corresponds to a list item, and the paging token (if there are more rows in the view than were returned).

3.1.4.25 GetVersionCollection

This operation [<54>](#) is used to return version information for the specified field of a specified list item within a specified list.

```

<wsdl:operation name="GetVersionCollection">

```

```

    <wsdl:input message="GetVersionCollectionSoapIn" />
    <wsdl:output message="GetVersionCollectionSoapOut" />
</wsdl:operation>

```

The protocol client sends a **GetVersionCollectionSoapIn** request message and the protocol server responds with a **GetVersionCollectionSoapOut** response message, as follows:

1. If the **strListItemID** is null or an empty string, the protocol server MUST return a SOAP fault with error code 0x82000001. This indicates that the parameter **strListItemID** is missing or invalid.
2. If the specified **strListID** is a valid GUID and corresponds to the identification of a list on the site, use that list.
3. If the specified **strListID** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **strListID** corresponds to the list title of a list on the site and if so, use that list.
4. If **strListID** does not correspond to a list from either of these checks, the protocol server SHOULD [<55>](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
5. If the **strListItemID** does not correspond to a list item in a specified list on the site, the protocol server MUST return a SOAP fault to indicate that the **strListItemID** parameter is missing or invalid. The error code is 0x82000001 when **strListItemID** is not 0. There is no error code when **strListItemID** is 0.
6. Check to make sure the specified **strFieldName** corresponds to a field in the specified list item of a specified list on the site.
7. If the **strFieldName** does not correspond to a field in a specified list item of a specified list on the site, the protocol server MUST return a SOAP fault. There is no error code for this fault.
8. If the protocol client excludes one or more parameters, or if the **strListID** is null, the protocol server MUST return a SOAP fault. There is no error code for this fault.
9. If the protocol client passes valid input parameters, the protocol server MUST return the version collection.

3.1.4.25.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
GetVersionCollectionSoapIn	The request for version information for the specified field of a specified list item within a specified list.
GetVersionCollectionSoapOut	The response to a request for version information for the specified field of a specified list item within a specified list.

3.1.4.25.1.1 GetVersionCollectionSoapIn

This message is the request of the **GetVersionCollection** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetVersionCollection
```

The SOAP Body contains a **GetVersionCollection** element.

3.1.4.25.1.2 GetVersionCollectionSoapOut

This message is the response of the **GetVersionCollection** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/GetVersionCollection
```

The SOAP Body contains a **GetVersionCollectionResponse** element.

3.1.4.25.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
GetVersionCollection	The request for version information for the specified field of a specified list item within a specified list.
GetVersionCollectionResponse	Contains the response to a request for version information for the specified field of a specified list item within a specified list.

3.1.4.25.2.1 GetVersionCollection

This element specifies details for the request to return version information for the specified field of a list.

```
<s:element name="GetVersionCollection">
  <s:complexType>
    <s:sequence>
      <s:element name="strlistID" type="s:string" />
      <s:element name="strlistItemID" type="s:string" />
      <s:element name="strFieldName" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

strlistID: The GUID or the list title of the list.

strlistItemID: The identification of the list item.

strFieldName : The case-sensitive name of the field.

3.1.4.25.2.2 GetVersionCollectionResponse

This element contains the response to the request to return version information for the specified field of a list.

If the operation succeeds, the collection of versions **MUST** be returned for the specified field; otherwise, the protocol server **MUST** return a SOAP fault.

```
<s:element name="GetVersionCollectionResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="GetVersionCollectionResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="Versions">
              <s:complexType>
                <s:sequence>
                  <s:element name="Version" minOccurs="0" maxOccurs="unbounded">
                    <s:complexType>
                      <s:attribute name="FieldName" type="s:string"/>
                      <s:attribute name="Modified" type="s:string"/>
                    </s:complexType>
                  </s:element>
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

GetVersionCollectionResult: The container element for the version history of the field that is specified by **strFieldName** in the request.

Versions: The collection of versions for the specified list item.

Version: Information about a particular version.

Version.FieldName: A placeholder for one or more properties associated with the list item. The attribute **Name** **MUST** be the name of the field (2) for which the **version** has been queried, the **Value** **MUST** be the value of the field (2) for which the version has been queried. For example, if **Title** is associated with the list item, the **Name** of the attribute **MUST** be "Title" and the **Value** of the attribute **MUST** be the value of the **Title** field for the specified list item.

Version.Modified: The date and time when this field was modified.

3.1.4.26 UndoCheckOut

This operation [<56>](#) is used to undo the **check out** of the specified file in a document library.

```
<wsdl:operation name="UndoCheckOut">
  <wsdl:input message="UndoCheckOutSoapIn" />
  <wsdl:output message="UndoCheckOutSoapOut" />
</wsdl:operation>
```

</wsdl:operation>

The protocol client sends an **UndoCheckOutSoapIn** request message and the protocol server responds with an **UndoCheckOutSoapOut** response message, as follows:

1. If the **pageUrl** is null or an empty string, the protocol server MUST return a SOAP fault with error code 0x82000001. This indicates that the parameter **pageUrl** is missing or invalid.
2. If the **pageUrl** is an invalid URL, the protocol server MUST return a SOAP fault with error code 0x80070002. This indicates that the parameter **pageUrl** is invalid.
3. If the **pageUrl** does not refer to a document, then the protocol server MUST return a SOAP fault. There is no error code for this fault.
4. If there are no errors, the protocol server MUST undo the checkout operation on the specified document.

3.1.4.26.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
UndoCheckOutSoapIn	The request to undo the check out of the specified file in a document library.
UndoCheckOutSoapOut	The response to a request to undo the check out of the specified file in a document library.

3.1.4.26.1.1 UndoCheckOutSoapIn

This message is the request to undo checkout of the specified document based on its location.

The SOAP Action value of the message is defined as follows:

`http://schemas.microsoft.com/sharepoint/soap/UndoCheckOut`

The SOAP Body contains an **UndoCheckOut** element.

3.1.4.26.1.2 UndoCheckOutSoapOut

This message is the response that confirms the operation to undo checkout of the document.

The SOAP Action value of the message is defined as follows:

`http://schemas.microsoft.com/sharepoint/soap/UndoCheckOut`

The SOAP Body contains an **UndoCheckOutResponse** element.

3.1.4.26.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
UndoCheckOut	The request to undo the check out of the specified file in a document library.
UndoCheckOutResponse	Contains the response to a request to undo the check out of the specified file in a document library.

3.1.4.26.2.1 UndoCheckOut

This element specifies details for the request to undo checkout of the document based on the provided **pageUrl**.

```
<s:element name="UndoCheckOut">
  <s:complexType>
    <s:sequence>
      <s:element name="pageUrl" type="s:string"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

pageUrl: The URL of the document for which to undo checkout.

3.1.4.26.2.2 UndoCheckOutResponse

This element contains the response that is sent after a successful **UndoCheckOut** operation.

```
<s:element name="UndoCheckOutResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="UndoCheckOutResult" type="s:boolean"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

UndoCheckOutResult: Specifies whether the call is successful or not. The value is True if the operation is successful; otherwise, False is returned.

3.1.4.27 UpdateContentType

This operation [<57>](#) is used to update a content type on a list.

```
<wsdl:operation name="UpdateContentType">
  <wsdl:input message="UpdateContentTypeSoapIn" />
  <wsdl:output message="UpdateContentTypeSoapOut" />
</wsdl:operation>
```

The protocol client sends an **UpdateContentTypeSoapIn** request message and the protocol server responds with an **UpdateContentTypeSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.

2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and, if so, use that list.
3. If **listName** does not correspond to a list from either of these checks, a SOAP fault SHOULD <58> be returned with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
4. If the **contentTypeId** is not a valid content type identifier or is not the content type identifier of a content type on the list, the protocol server MUST return a SOAP fault. There is no error code for this fault.
5. If the specified content type is found, the protocol server deletes the appropriate fields from the content type. If an error is encountered during this operation, the protocol server MUST return a SOAP fault and all fields that were deleted before the error was encountered MUST stay deleted. There is no specific error code for this fault.
6. The protocol server adds the appropriate fields to the content type. If an error is encountered during this operation, the protocol server MUST return a SOAP fault, all fields deleted from the content type MUST stay deleted, and all fields added to the content type before the error was encountered MUST stay added to the content type. There is no specific error code for this fault.
7. The protocol server updates the appropriate fields on the content type. If an error is encountered during this operation, the protocol server MUST return a SOAP fault, all fields deleted from or added to the content type MUST stay deleted or added, and all fields updated on the content type before the error was encountered MUST stay updated on the content type. There is no specific error code for this fault.
8. The protocol server updates the content type with the given properties. If an error is encountered during this operation, the protocol server MUST return a SOAP fault. There is no specific error code for this fault. The fields that were deleted, added, or updated MUST stay deleted, added, or updated and all properties updated before the error was encountered MUST stay updated.

If changes to the content type are allowed according to the permission settings on the list, the protocol server MUST NOT carry out operations other than deleting, adding, or updating fields.

3.1.4.27.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
UpdateContentTypeSoapIn	The request to update a content type on a list.
UpdateContentTypeSoapOut	The response to a request to update a content type on a list.

3.1.4.27.1.1 UpdateContentTypeSoapIn

This message is the request of the **UpdateContentType** operation.

The SOAP action value of the message is defined as follows:

`http://schemas.microsoft.com/sharepoint/soap/UpdateContentType`

The SOAP body contains an **UpdateContentType** element.

3.1.4.27.1.2 UpdateContentTypeSoapOut

This message is the response of the **UpdateContentType** operation. The SOAP action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/UpdateContentType
```

The SOAP body contains an **UpdateContentTypeResponse** element.

3.1.4.27.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
UpdateContentType	The request to update a content type on a list.
UpdateContentTypeResponse	Contains the response to a request to update a content type on a list.

3.1.4.27.2.1 UpdateContentType

This element specifies details for the request to update a content type on the list.

```
<s:element name="UpdateContentType">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
      <s:element name="contentTypeId" type="core:ContentTypeId" />
      <s:element name="contentTypeProperties" >
        <s:complexType>
          <s:sequence>
            <s:element name="ContentType" type="tns:ContentTypePropertyDefinition"
              minOccurs="0" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="newFields" type="tns:AddOrUpdateFieldsDefinition"
        minOccurs="0"/>
      <s:element name="updateFields" type="tns:AddOrUpdateFieldsDefinition"
        minOccurs="0"/>
      <s:element name="deleteFields" type="tns>DeleteFieldsDefinition"
        minOccurs="0"/>
      <s:element name="addToView" type="core:TRUE_Case_Sensitive_Else_Anything"
        minOccurs="0"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

listName: The GUID or the list title of the list.

contentTypeId: The identification of the content type to be updated.

contentTypeProperties: The container for properties to set on the content type. MUST NOT contain only white space.

ContentType: The container for properties to set on the content type.

newFields: The container for a list of existing fields to be included in the content type.

updateFields: The container for a list of fields to be updated on the content type.

deleteFields: The container for a list of existing fields to be removed from the content type.

addToView: Specifies whether the fields will be added to the default list view, where "TRUE" MUST correspond to true, and all other values correspond to false.

3.1.4.27.2.2 UpdateContentTypeResponse

This element contains the response for a request to update a content type on a list.

```
<s:element name="UpdateContentTypeResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="UpdateContentTypeResult" minOccurs="0">
        <s:complexType>
          <s:sequence>
            <s:element name="Results">
              <s:complexType>
                <s:sequence>
                  <s:element name="Method" minOccurs="0" maxOccurs="unbounded">
                    <s:complexType>
                      <s:sequence>
                        <s:element name="ErrorCode" type="s:string" />
                        <s:element name="FieldRef"
                          type="tns:FieldReferenceDefinitionCT"
                          minOccurs="0" />
                        <s:element name="Field" type="core:FieldDefinition"
                          minOccurs="0" />
                        <s:element name="ErrorText"
                          type="s:string" minOccurs="0" />
                      </s:sequence>
                    <s:attribute name="ID" type="s:string" use="required"/>
                  </s:complexType>
                </s:element>
              <s:element name="ListProperties">
                <s:complexType>
                  <s:attribute name="Description" type="s:string"
                    use="optional" />
                  <s:attribute name="FeatureId"
                    type="core:UniqueIdentifierWithOrWithoutBraces"
                    use="optional"/>
                  <s:attribute name="Group" type="s:string" use="optional" />
                  <s:attribute name="Hidden"
                    type="core:TRUE_NegOne_Else_Anything"
                    use="optional" />
                  <s:attribute name="ID" type="core:ContentTypeId"
                    use="required" />
                  <s:attribute name="Name" type="s:string" use="required" />
                  <s:attribute name="ReadOnly"
                    type="core:TRUE_NegOne_Else_Anything" />
                </s:complexType>
              </s:element>
            </s:sequence>
          </s:complexType>
        </s:element>
      </s:sequence>
    </s:complexType>
  </s:element>
</s:sequence>
</s:complexType>
</s:element>
```

```

        use="optional" />
        <s:attribute name="Sealed"
            type="core:TRUE_Case_Sensitive_Else_Anything"
            use="optional" />
        <s:attribute name="V2ListTemplateName" type="s:string"
            use="optional"/>
        <s:attribute name="Version" type="s:long" use="optional" />
        <s:anyAttribute namespace="##other" processContents="lax" />
    </s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>

```

UpdateContentTypeResult: If no error conditions, as specified previously, cause the protocol server to return a SOAP fault, an **UpdateContentTypeResult** MUST be returned.

Results: The container for data on the update of a content type.

Method: The container for data on a field add, update, or remove operation.

Method.ID: The **identification** of the **Method** that is specified in the request.

ErrorCode: For an add or delete operation, if the field to be added or deleted is found on the list or context site, the value of **ErrorCode** MUST be "0x00000000". For an update operation, if a reference to the field to be updated is found on the content type, the value MUST be "0x00000000". Otherwise, the value of **ErrorCode** MUST be nonzero.

FieldRef: If the **ErrorCode** is "0x00000000" for an add operation, **FieldRef** MUST contain data for the reference to the field.

Field: A **FieldDefinition**, as specified by [\[MS-WSSFO2\]](#) section 2.2.8.3.3. If the **ErrorCode** is "0x00000000" for an update operation, **Field** MUST contain data for the Field.

ErrorText: If the **ErrorCode** is nonzero, **ErrorText** MUST contain an error string.

ListProperties: Contains updated data for the content type.

ListProperties.Description: The description of the content type.

ListProperties.FeatureId: Specifies which feature this content type is associated with.

ListProperties.Group: The name of the content type group in which this content type is contained.

ListProperties.Hidden: Specifies whether the content type SHOULD appear in user interface.

ListProperties.ID: The content type identifier for this content type.

ListProperties.Name: The name of the content type.

ListProperties.ReadOnly: Specifies whether the content type can be edited.

ListProperties.Sealed: Specifies whether another content type can inherit from the current content type.

ListProperties.V2ListTemplateName: Specifies an additional component that can be used for field names to ensure their uniqueness.

3.1.4.28 UpdateContentTypesXmlDocument

This operation [<59>](#) is used to update XML document properties of the content type collection on a list.

```
<wsdl:operation name="UpdateContentTypesXmlDocument">
  <wsdl:input message="UpdateContentTypesXmlDocumentSoapIn" />
  <wsdl:output message="UpdateContentTypesXmlDocumentSoapOut" />
</wsdl:operation>
```

The protocol client sends an **UpdateContentTypesXmlDocumentSoapIn** request message and the protocol server responds with an **UpdateContentTypesXmlDocumentSoapOut** response message, as follows:

1. If the **newDocument** is valid XML, but does not follow the schema for **newDocument**, the protocol server MUST ignore it, and return an empty **UpdateContentTypesXmlDocumentResult**.
2. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
3. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and, if so, use that list.
4. If the list specified by the **listName** is not found, the protocol server SHOULD [<60>](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
5. If the **newDocument.ContentTypes** element contains more than 1000 child elements, the protocol server MUST return a SOAP fault with error code -2146232832 . This indicates that the request specified too many content types.
6. If the content type specified by the **ContentType.ID** attribute is not found, the protocol server MUST return a SOAP fault. There is no error code for this fault.
7. If the content type specified by the **ContentType.ID** attribute does not contain any child elements, the protocol server MUST return a SOAP fault. There is no error code for this fault.
8. If an XML document with the same namespace URI [\[RFC2396\]](#) already exists, it MUST be replaced by the new XML Document.
9. Otherwise, the new XML Document MUST be added to the XML document collection that the content type specified by the **ContentType.ID** attribute and the protocol server MUST return a success **UpdateContentTypesXmlDocumentResult**.

3.1.4.28.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
UpdateContentTypesXmlDocumentSoapIn	The request to update XML document properties of the content type collection on a list.
UpdateContentTypesXmlDocumentSoapOut	The response to a request to update XML document properties of the content type collection on a list.

3.1.4.28.1.1 UpdateContentTypesXmlDocumentSoapIn

This message is **the request of the UpdateContentTypesXmlDocument** operation. The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/UpdateContentTypesXmlDocument
```

The SOAP Body contains an **UpdateContentTypesXmlDocument** element.

3.1.4.28.1.2 UpdateContentTypesXmlDocumentSoapOut

This message is the response of the **UpdateContentTypesXmlDocument** operation. The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/UpdateContentTypesXmlDocument
```

The SOAP Body contains an **UpdateContentTypesXmlDocumentResponse** element.

3.1.4.28.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
UpdateContentTypesXmlDocument	The request to update XML document properties of the content type collection on a list.
UpdateContentTypesXmlDocumentResponse	Contains the response to a request to update XML document properties of the content type collection on a list.

3.1.4.28.2.1 UpdateContentTypesXmlDocument

The request to update the XML document property of the content type collection on the list.

```
<s:element name="UpdateContentTypesXmlDocument">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
      <s:element name="newDocument">
        <s:complexType>
          <s:sequence>
            <s:element name="ContentTypes">
              <s:complexType>
```

```

<s:sequence>
  <s:element name="ContentType" maxOccurs="unbounded">
    <s:complexType>
      <s:sequence>
        <s:any />
      </s:sequence>
      <s:attribute name="ID" type="s:string" />
    </s:complexType>
  </s:element>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>

```

listName: The GUID or the list title of the list in the context site.

newDocument: The container element for a list of content type and XML document to update. It SHOULD use the following schema to have the protocol server update the XML document. It can be any valid XML, but the protocol server MUST ignore any other XML.

newDocument.ContentTypes: Contains a list of content types with XML document information to update.

newDocument.ContentTypes.ContentType: Contains the XML document information to update for the content type specified by the **ContentType.ID** attribute. The protocol client MUST NOT update the XML document with the following namespace URI [\[RFC2396\]](#):

Namespace URI	Description
http://schemas.microsoft.com/office/2006/metadata/customXsn	[MS-WSSCAML] , customXsn
office.server.policy	[MS-WSSCAML], Policy
microsoft.office.server.policy.changes	[MS-WSSCAML], PolicyDirtyBag
schemas.microsoft.com/sharepoint/v3/contenttype/forms	[MS-WSSCAML], FormTemplateDefinitions
schemas.microsoft.com/sharepoint/events	[MS-WSSCAML], ReceiverDefinitions
schemas.microsoft.com/sharepoint/v3/contenttype/forms/url	[MS-WSSCAML], FormUrlDefinitions

newDocument.ContentTypes.ContentType.ID: The content type identifier of the content type to be updated.

3.1.4.28.2.2 UpdateContentTypesXmlDocumentResponse

This element contains the response for updating the XML document property of the content type collection on the list.


```

<s:element name="UpdateContentTypesXmlDocumentResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="UpdateContentTypesXmlDocumentResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:element name="Success" minOccurs="0">
              <s:complexType />
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>

```

UpdateContentTypesXmlDocumentResult: If no error conditions cause the protocol server to return a SOAP fault, an **UpdateContentTypesXmlDocumentResult** MUST be returned.

Success: An empty element, the presence of which indicates that the request was handled successfully.

3.1.4.29 UpdateContentTypeXmlDocument

This operation [61](#) is used to update the XML document of a list content type.

```

<wsdl:operation name="UpdateContentTypeXmlDocument">
  <wsdl:input message="UpdateContentTypeXmlDocumentSoapIn" />
  <wsdl:output message="UpdateContentTypeXmlDocumentSoapOut" />
</wsdl:operation>

```

The protocol client sends an **UpdateContentTypeXmlDocumentSoapIn** request message and the protocol server responds with an **UpdateContentTypeXmlDocumentSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [62](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
4. If the content type specified by the **contentTypeId** is not found, the protocol server MUST return a SOAP fault. There is no error code for this fault.
5. If an XML document with the same namespace URI [RFC2396](#) already exists, it MUST be replaced by the new XML document. Otherwise, the new XML document MUST be added to the content type XML document.

6. If no SOAP fault is thrown, the protocol server MUST return a success **UpdateContentTypeXmlDocumentResult**.

3.1.4.29.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
UpdateContentTypeXmlDocumentSoapIn	The request to update the XML document of a list content type.
UpdateContentTypeXmlDocumentSoapOut	The response to a request to update the XML document of a list content type.

3.1.4.29.1.1 UpdateContentTypeXmlDocumentSoapIn

This message is the request of the **UpdateContentTypeXmlDocument** operation. The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/UpdateContentTypeXmlDocument
```

The SOAP Body contains an **UpdateContentTypeXmlDocument** element.

3.1.4.29.1.2 UpdateContentTypeXmlDocumentSoapOut

This message is the response of the **UpdateContentTypeXmlDocument** operation. The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/UpdateContentTypeXmlDocument
```

The SOAP Body contains an **UpdateContentTypeXmlDocumentResponse** element.

3.1.4.29.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
UpdateContentTypeXmlDocument	The request to update the XML document of a list content type.
UpdateContentTypeXmlDocumentResponse	Contains the response to a request to update the XML document of a list content type.

3.1.4.29.2.1 UpdateContentTypeXmlDocument

This element specifies details for the request to update the XML document of a list content type.

```
<s:element name="UpdateContentTypeXmlDocument">
  <s:complexType>
```

```

<s:sequence>
  <s:element name="listName" type="s:string" />
  <s:element name="contentTypeId" type="core:ContentTypeId" />
  <s:element name="newDocument" >
    <s:complexType mixed="true">
      <s:sequence>
        <s:any />
      </s:sequence>
    </s:complexType>
  </s:element>
</s:sequence>
</s:complexType>
</s:element>

```

listName: The GUID or the list title of the list in the context site.

contentTypeId: The content type identifier of the content type that is to be updated on the list.

newDocument: The XML document to be added to the content type XML document collection. The **newDocument** MUST include a namespace definition. The protocol client MUST NOT update the XML document with the following namespace URI [\[RFC2396\]](#):

Namespace URI	Description
http://schemas.microsoft.com/office/2006/metadata/customXsn	[MS-WSSCAML] , customXsn
office.server.policy	[MS-WSSCAML], Policy
microsoft.office.server.policy.changes	[MS-WSSCAML], PolicyDirtyBag
schemas.microsoft.com/sharepoint/v3/contenttype/forms	[MS-WSSCAML], FormTemplateDefinitions
schemas.microsoft.com/sharepoint/events	[MS-WSSCAML], ReceiverDefinitions
schemas.microsoft.com/sharepoint/v3/contenttype/forms/url	[MS-WSSCAML], FormUrlDefinitions

3.1.4.29.2.2 UpdateContentTypeXmlDocumentResponse

This element contains the response for updating the XML document of a list content type.

```

<s:element name="UpdateContentTypeXmlDocumentResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="UpdateContentTypeXmlDocumentResult" minOccurs="0">
        <s:complexType mixed="true">
          <s:sequence>
            <s:all/>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>

```

UpdateContentTypeXmlDocumentResult: If the operation succeeds, an **UpdateContentTypeXmlDocumentResult** MUST be returned. If the operation fails, a SOAP fault MUST be returned.

3.1.4.30 UpdateList

The **UpdateList** operation is used to update list properties and add, remove, or update fields.

```
<wsdl:operation name="UpdateList">
  <wsdl:input message="UpdateListSoapIn" />
  <wsdl:output message="UpdateListSoapOut" />
</wsdl:operation>
```

The protocol client sends an **UpdateListSoapIn** request message. The server MUST respond with an **UpdateListSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [return a SOAP fault with error code 0x82000006](#). This indicates that the list does not exist or might have been deleted by another user.
4. If the **listVersion** is null, the list MUST be updated and the protocol server MUST return an **UpdateListReponse** element.
5. If the **listVersion** string cannot be converted to an integer, the protocol server MUST return a SOAP fault. There is no error code for this fault.
6. If the **listVersion** string is numeric but not within the range of an unsigned 32-bit integer, **UpdateList** MUST return a SOAP fault. There is no error code for this fault.
7. If the **listVersion** does not match the version of the list, the protocol server MUST return a SOAP fault with error code 0x81020015. This indicates that the list changes are in conflict with those made by another user.

3.1.4.30.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
UpdateListSoapIn	The request to update list properties and add, remove, or update fields.
UpdateListSoapOut	The response to a request to update list properties and add, remove, or update fields.

3.1.4.30.1.1 UpdateListSoapIn

This message is the request that contains required parameters for the **UpdateList** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/UpdateList
```

The SOAP Body contains an **UpdateList** element.

3.1.4.30.1.2 UpdateListSoapOut

This message is the response that contains the results returned by the **UpdateList** operation.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/UpdateList
```

The SOAP Body contains an **UpdateListResponse** element.

3.1.4.30.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
UpdateList	The request to update list properties and add, remove, or update fields.
UpdateListResponse	Contains the response to a request to update list properties and add, remove, or update fields.

3.1.4.30.2.1 UpdateList

This element specifies the details for updating list properties and adding, removing, or updating fields.

```
<s:element name="UpdateList"> <s:complexType>
  <s:sequence>
    <s:element name="listName" type="s:string" minOccurs="0"/>
    <s:element name="listProperties" minOccurs="0">
      <s:complexType mixed="true">
        <s:sequence>
          <s:element name="List" minOccurs="0">
            <s:complexType>
              <s:sequence>
                <s:element name="Validation"
                  minOccurs="0">
                  <s:complexType>
                    <s:attribute name="Message" type="s:string" use="optional" />
                  </s:complexType>
                </s:element>
              </s:sequence>
            </s:complexType>
          </s:element>
          <s:attribute name="AllowMultiResponses" type="core:TRUEFALSE" />
          <s:attribute name="Description" type="s:string" />
          <s:attribute name="Direction" type="s:string" />
          <s:attribute name="EnableAssignToEmail" type="core:TRUEFALSE" />
          <s:attribute name="EnableAttachments" type="core:TRUEFALSE" />
        </s:sequence>
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>
</s:element>
```

```

    <s:attribute name="EnableModeration" type="core:TRUEFALSE" />
    <s:attribute name="EnableVersioning" type="core:TRUEFALSE" />
    <s:attribute name="EnforceDataValidation" type="s:boolean" use="optional"/>
    <s:attribute name="Hidden" type="core:TRUEFALSE" />
    <s:attribute name="MultipleDataList" type="core:TRUEFALSE" />
    <s:attribute name="Ordered" type="core:TRUEFALSE" />
    <s:attribute name="PreserveEmptyValues" type="s:boolean" />
    <s:attribute name="ShowUser" type="core:TRUEFALSE" />
    <s:attribute name="StrictTypeCoercion" type="s:boolean" />
    <s:attribute name="Title" type="s:string" />
  </s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="newFields" type="tns:UpdateListFieldsRequest"
  minOccurs="0" />
<s:element name="updateFields" type="tns:UpdateListFieldsRequest"
  minOccurs="0" />
<s:element name="deleteFields" type="tns:UpdateListFieldsRequest"
  minOccurs="0" />
<s:element name="listVersion" type="s:string" minOccurs="0"/>
</s:sequence>
</s:complexType>
</s:element>

```

listName: The GUID or the list title of the list from which to update properties and fields.

listProperties: The properties of the specified list.

Validation: An expression, as specified by [\[MS-WSSTS\]](#) section 2.9. Each **data-reference** in the expression MUST correspond to a valid **DisplayName** attribute of a field (2) in the current list (1). If present, it MUST not be longer than 1024 characters.

Validation.Message: A message used to indicate that validation was not satisfied. If present, it MUST not be longer than 1024 characters.

Attribute	Description
AllowMultiReponses	Specifies whether multiple responses are allowed for a given user when the list is a Survey list. The value MUST be ignored when the list is not a Survey list.
Description	A string that contains the description for the list.
Direction	A string that MUST contain one of the following values: LTR if the reading order is left-to-right, RTL if it is right-to-left, or None. If None, the default value is taken from the list template.
EnableAssignedToEmail	TRUE to enable Assigned-to e-mail for a list of type Issues list.
EnableAttachments	TRUE to enable attachments to items in the list. Does not apply to document libraries.
EnableModeration	TRUE to enable Content Approval for the list.
EnableVersioning	TRUE to enable version control for the list.

Attribute	Description
EnforceDataValidation	TRUE if the protocol server SHOULD enforce the field properties that limit the data values that can be stored in that field.
Hidden	TRUE to hide the list so that it does not appear on the Documents and Lists page, Quick Launch bar, Modify Site Content page, or Add Column page as an option for lookup fields.
MultipleDataList	TRUE to specify that the list in a Meeting Workspace site contains data for multiple meeting instances within the site.
Ordered	TRUE to specify that the option to allow users to reorder items in the list is available on the Edit view page for the list.
PreserveEmptyValues	True if Null values are preserved for expression evaluation. False if there are no special requirements on how the protocol server treats Null values. <64>
ShowUser	TRUE to specify whether to show the survey respondent's name in the survey results.
StrictTypeCoercion	True if expressions anywhere in the list that include invalid type casts such as casting a date to a string SHOULD return an error instead of attempting to perform the cast. <65>
Title	A string that contains the title of the list.

newFields: An **UpdateFieldsRequest** object that defines the new fields to be added to the list.

updateFields: An **UpdateFieldsRequest** object that defines the fields to be updated in the list.

deleteFields: An **UpdateFieldsRequest** object that defines the fields to be deleted from the list.

listVersion: A string representation of an integer value that specifies the current version of the list. This is used to determine if the protocol client is working off of the same version of the schema that is currently maintained by the protocol server. If the **listVersion** value sent up in the **UpdateList** request does not match the current version then the protocol server MUST respond with a SOAP fault with error code 0x81020015. This indicates that the changes being requested via **UpdateList** will conflict with whatever changes have already been made to this list that resulted in the protocol server **listVersion** being higher than the one sent by the protocol client.

3.1.4.30.2.2 UpdateListResponse

This element contains the response to the request to update list properties and add, delete, or update fields.

```

<s:element name="UpdateListResponse"> <s:complexType>
  <s:sequence>
    <s:element name="UpdateListResult" minOccurs="0">
      <s:complexType mixed="true">
        <s:sequence>
          <s:element name="Results">
            <s:complexType mixed="true">
              <s:sequence>
                <s:element name="NewFields" type="tns:UpdateListFieldResults" />
                <s:element name="UpdateFields"
                  type="tns:UpdateListFieldResults" />
                <s:element name="DeleteFields"

```

```

        type="tns:UpdateListFieldResults" />
        <s:element name="ListProperties" type="tns:ListDefinitionCT" />
    </s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>

```

UpdateListResult: The results of the **UpdateList** request.

Results: The container element for the result categories.

NewFields: The container element for the results of any add field requests. See section [2.2.4.14](#).

DeleteFields: The container element for the results of any delete field requests. See section [2.2.4.14](#).

UpdateFields: The container element for the results of any update field requests. See section [2.2.4.14](#).

ListProperties: The container element for the results of any Content Type requests.

3.1.4.31 UpdateListItems

The **UpdateListItems** operation is used to make inserts, updates, and deletions to specified list items in a list.

```

<wsdl:operation name="UpdateListItems">
    <wsdl:input message="UpdateListItemsSoapIn" />
    <wsdl:output message="UpdateListItemsSoapOut" />
</wsdl:operation>

```

The protocol client sends an **UpdateListItemsSoapIn** request message and the protocol server responds with an **UpdateListItemsSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and if so, use that list.
3. If **listName** does not correspond to a list from either of these checks, the protocol server SHOULD [66](#) return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
4. If **listName** is an empty string, the server returns the SOAP fault: "Cannot access a closed Stream".

- Otherwise, the protocol server MUST process the batched operations on the list and return success or failure conditions per operation.

3.1.4.31.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
UpdateListItemsSoapIn	The request to make inserts, updates, and deletions to specified list items in a list.
UpdateListItemsSoapOut	The response to a request to make inserts, updates, and deletions to specified list items in a list.

3.1.4.31.1.1 UpdateListItemsSoapIn

This message is the request of the **UpdateListItems** operations that the protocol client sends as a request to this protocol server.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/UpdateListItems
```

The SOAP Body contains an **UpdateListItems** element.

3.1.4.31.1.2 UpdateListItemsSoapOut

This message is the response of the **UpdateListItems** operation that the protocol client receives from this protocol server.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/UpdateListItems
```

The SOAP Body contains an **UpdateListItemsResponse** element.

3.1.4.31.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
UpdateListItems	The request to make inserts, updates, and deletions to specified list items in a list.
UpdateListItemsResponse	Contains the response to a request to make inserts, updates, and deletions to specified list items in a list.

3.1.4.31.2.1 UpdateListItems

This element specifies the request that MUST be sent for the **UpdateListItems** operation.

```
<s:element name="UpdateListItems">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
      <s:element name="updates">
        <s:complexType>
          <s:sequence>
            <s:element name="Batch">
              <s:complexType>
                <s:sequence>
                  <s:element name="Method" maxOccurs="unbounded">
                    <s:complexType mixed="true">
                      <s:sequence>
                        <s:element name="Field" maxOccurs="unbounded">
                          <s:complexType>
                            <s:simpleContent>
                              <s:extension base="s:string">
                                <s:attribute name="Name" type="s:string"/>
                              </s:extension>
                            </s:simpleContent>
                          </s:complexType>
                        </s:element>
                      </s:sequence>
                    </s:complexType>
                  </s:element>
                </s:sequence>
                <s:attribute name="ID" type="s:unsignedInt" use="required" />
                <s:attribute name="Cmd" type="tns:MethodCmdEnum"
                  use="required" />
              </s:complexType>
            </s:element>
          </s:sequence>
          <s:attribute name="ListVersion" type="s:unsignedInt" />
          <s:attribute name="OnError" type="tns:OnErrorEnum" />
          <s:attribute name="Version" type="s:string" />
          <s:attribute name="ViewName" type="s:string" />
          <s:attribute name="PreCalc"
            type="core:TRUE_Case_Sensitive_Else_Anything" />
          <s:attribute name="DateInUtc"
            type="core:TRUE_Case_Sensitive_Else_Anything" />
          <s:attribute name="Properties"
            type="core:TRUE_Case_Sensitive_Else_Anything" />
          <s:attribute name="LockSchema"
            type="core:TRUE_Case_Sensitive_Else_Anything" />
          <s:attribute name="RootFolder" type="s:string" />
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

listName: See the description of the **listName** parameter in section [3.1.4.24.2.1](#).

updates: Specifies the operations to perform on a list item.

There are several attributes supported by the **Batch** element as specified in the following table.

Attribute	Description
ListVersion	Specifies the version number of the list. MUST only change protocol server behavior when used in conjunction with LockSchema="TRUE" . The ListVersion to use is retrieved from a previous GetListItemChangesSinceToken call with no change token parameter included or a GetListItemChangesWithKnowledge call with no knowledge parameter included or where the schema on the protocol server changed or from a call to GetList or GetListAndView .
OnError	Specifies whether encountering an error SHOULD result in the entire operation being aborted, or whether subsequent methods in the batch SHOULD be executed. The default is to continue executing subsequent items.
Version	Specifies the version number of the service that is running on the protocol server. A version number consists of four integers in the format N.N.N.NNNN, which represent the major, minor, phase, and incremental versions of the product. This MUST be ignored by the protocol server.
ViewName	Specifies the GUID for the view to update.
PreCalc	Can be sent by the protocol client. MUST be ignored by the protocol server.
DateInUtc	Specifies whether dates passed up and returned are in UTC format. This defaults to FALSE.
Properties	Specifies whether the properties in the list item property collection are to be returned as separate fields.
LockSchema	If TRUE, and ListVersion is specified with a number, the protocol server MUST return a SOAP fault with error code 0x80070666 if the schema version passed in by the protocol client does not match the list schema version on the protocol server. This defaults to FALSE.
RootFolder	Specifies the URL of the folder where list items or list folders SHOULD be inserted, updated, or deleted. This can be a fully qualified URL or a site-relative URL. In the following examples, the list is called TestListItem which contains a folder TestNewFolder. Fully qualified URL example: <code>http://siteservice/Lists/TestListItem/TestNewFolder</code> Site-relative URL: <code>/Lists/TestListItem/TestNewFolder</code>

There MUST be one **Method** element per operation on a list item. The **Method.ID** attribute SHOULD be set to a value that is unique among the other **Method** elements used (for example, `<Method ID="1">` and `<Method ID="2">`). It is important to note that this identification is not related to the list item identifier that is specified in the **Field** element. If the **Method ID** attribute is unique, the protocol server MUST use the method identification to match up the request made to the protocol server with the protocol server response, so the protocol client can determine whether a particular operation failed or succeeded.

The **Method** operation supports several values for the **Cmd** attribute as specified in the following table.

Value	Description
New	Adds a new list item to the specified list. <Field Name="ID"> MUST be ignored, if specified, because the identification is a unique number generated and set only by the protocol server. Additional field references MUST set those columns to the value specified or an error will be returned.
Update	Updates fields for a specific list item. <Field Name="ID"> MUST be specified with the value equal to the value of the identification column for the list item that SHOULD be updated by the protocol server. Additional field references MUST set those columns to the value specified or an error will be returned.
Delete	Deletes a specific list item. <Field Name="ID"> MUST be specified with the value being the value of the identification column for the list item that SHOULD be deleted by the protocol server.
Moderate	Allows the protocol client to change the moderation status field for a specific list item. <Field Name="ID"> MUST be included to specify which item. The moderation status field MUST only be updated when the list has EnableModeration set to True. The protocol client MAY update the moderation status field to any 32-bit integer value, but SHOULD interpret the following integers in a specific way: <ul style="list-style-type: none"> ▪ 0 – The list item contains data that has been approved through the content approval process. ▪ 1 – The list item contains data that has been rejected through the content approval process. ▪ 2 – The list item contains data that is pending content approval. ▪ 3 – The list item contains data that is in an incomplete or draft state. ▪ 4 – The list item is currently scheduled in the content approval process.
Move <67>	Allows the protocol client to move a document in a document library to a new destination inside the same document library. <Field Name="ID"> MUST be specified with the value equal to the value of the identification column for the list item that SHOULD be moved by the protocol server. <Field Name="MoveNewUrl"> MUST be specified to be the new destination of the document inside the same document library.

The protocol client MUST set the **Name** attribute of the **Field** element equal to the **Name** attribute of the **Field** returned by **GetList** to specify which field to modify (or add). The protocol server MUST set the field's value equal to the value of the **Field** element in the protocol client response or return an error indicating why the update failed, unless the field is a special field. The identification field is a special field that indicates to the protocol server which list item SHOULD be updated. The **owshiddenversion** field is a special field that is used for conflict detection. The protocol server MUST NOT update these special fields with the values that the protocol client includes in the request. To operate on a **document** in a document library, the protocol client MUST set the **FileRef** field to be the URL of the document.

Additionally, the property collection for a list item is added or updated on the protocol server by including one of the following field references:

Value	Description
<Field Name="MetaInfo">Value</Field>	Adds or updates the entire property collection with specified name/value pairs.

Value	Description
<Field Name="MetaInfo" Property="Name">Value</Field>	Adds or updates a specific name/value pair in the property collection. Name is replaced by the property name to be set and Value is replaced by the value to store.

The **owshiddenversion** field is for conflict detection. The protocol client and protocol server use this field to detect and manage conflicts as follows:

- If the protocol client does not specify the **owshiddenversion** field reference in the **UpdateListItems** request, then the protocol server MUST overwrite any changes in the list item or return an error.
- If the protocol client specifies the **owshiddenversion** field reference in the **UpdateListItems** request to ensure conflict detection, the protocol client SHOULD send the value of the **owshiddenversion** field that it has kept from its most recent **GetListItems**, **GetListItemChanges**, or **GetListItemChangesSinceToken**, or **GetListItemChangesWithKnowledge** request.
- If the **owshiddenversion** value specified by the protocol client is equal to the **owshiddenversion** field's value for the list item on the protocol server, the protocol server MUST update the list item or return an error indicating why the update failed.
- If the **owshiddenversion** specified by the protocol client is different than the current value of the **owshiddenversion** field's value for the list item on the protocol server, the protocol server MUST return error code 0x00000000 and the protocol client MUST decide how to deal with the conflict:
- The protocol client MAY discard the changes and revert the list item values that it has stored to the values on the protocol server.
 - The protocol client MAY choose to incorporate changes on the protocol server that are returned by the **UpdateListItems** response and send back an **UpdateListItems** request containing the updates to be made to the list item with the **owshiddenversion** value set to the value returned by the failed **UpdateListItems** request.
 - The protocol client MAY choose to overwrite the list item values on the protocol server by passing back the same **UpdateListItems** request to the protocol server, except with the **owshiddenversion** value set to the value returned by the failed **UpdateListItems** request.

The following is an example of handling conflicts.

The first request is made to update Item 1 with a new title as follows:

```
Content-Type: text/xml; charset=utf-8
SOAPAction: "http://schemas.microsoft.com/sharepoint/soap/UpdateListItems"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <UpdateListItems xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>TestListItem</listName>
      <updates>
        <Batch OnError="Return">
          <Method ID="1" Cmd="Update">
```

```

        <Field Name="ID">1</Field>
        <Field Name="Title">Test new title</Field>
        <Field Name="owshiddenversion">1</Field>
    </Method>
</Batch>
</updates>
</UpdateListItems>
</soap:Body>
</soap:Envelope>

```

The protocol server gives a response that indicates a conflict:

```

<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><UpdateListItemsResponse
xmlns="http://schemas.microsoft.com/sharepoint/soap/"><UpdateListItemsResult><Results><Result
ID="1,Update" List="{E074E60A-0126-4446-A751-A5BE7A55ED81}"
Version="2"><ErrorCode>0x81020015</ErrorCode><ErrorText>Save Conflict\n\nYour changes
conflict with those made concurrently by another user. To apply changes, click Back in your
Web browser, refresh the page, and resubmit your changes.</ErrorText><z:row
ows_ContentTypeId="0x010047334455D6BA5346AC55E03BB37AD578" ows_Title="Test new entry"
ows_Num="20.0000000000000" ows_Text="TestTest" ows_DateTime="2008-03-03 15:20:00"
ows_lookup="1;#Admin" ows_ID="1" ows_ContentType="Item" ows_Modified="2008-01-31 15:26:19"
ows_Created="2008-01-28 08:06:11" ows_Author="1;#Admin" ows_Editor="1;#Admin"
ows_owshiddenversion="3" ows_WorkflowVersion="1" ows_UIVersion="512"
ows_UIVersionString="1.0" ows_Attachments="0" ows_ModerationStatus="0"
ows_LinkTitleNoMenu="Test new entry" ows_LinkTitle="Test new entry" ows_SelectTitle="1"
ows_Order="100.0000000000000" ows_GUID="{2398096C-624B-4FA1-997F-36F6AC4776A2}"
ows_FileRef="1;#Lists/TestListItem/1.000" ows_FileDirRef="1;#Lists/TestListItem"
ows_Last_x0020_Modified="1;#2008-01-28 08:06:11" ows_Created_x0020_Date="1;#2008-01-28
08:06:11" ows_FSObjType="1;#0" ows_PermMask="0x7fffffff" ows_FileLeafRef="1;#1.000"
ows_UniqueId="1;#{078CAB5C-5FEA-43EA-B23A-FB93EF35397D}" ows_ProgId="1;#"
ows_ScopeId="1;#{E1CCC98A-9A01-4193-8AE5-6E6765AC3054}" ows_EditMenuTableStart="1.000"
ows_EditMenuTableEnd="1" ows_LinkFilenameNoMenu="1.000" ows_LinkFilename="1.000"
ows_ServiceUrl="/Lists/TestListItem/1.000"
ows_EncodedAbsUrl="http://siteaddress/Lists/TestListItem/1.000" ows_BaseName="1_"
ows_MetaInfo="1;#" ows_Level="1" ows_IsCurrentVersion="1" xmlns:z="#RowsetSchema"
/></Result></Results></UpdateListItemsResult></UpdateListItemsResponse></soap:Body></soap:Env
elope>

```

The protocol client makes a second request to the protocol server including the updated **owshiddenversion** value from the protocol server's response:

```

Content-Type: text/xml; charset=utf-8
SOAPAction: "http://schemas.microsoft.com/sharepoint/soap/UpdateListItems"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <UpdateListItems xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>TestListItem</listName>
      <updates>
        <Batch OnError="Return">
          <Method ID="1" Cmd="Update">
            <Field Name="ID">1</Field>

```

```

        <Field Name="Title">Test new title</Field>
        <Field Name="owshiddenversion">3</Field>
    </Method>
</Batch>
</updates>

```

The protocol server then responds indicating a successful update:

```

<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"><soap:Body><UpdateListItemsResponse
xmlns="http://schemas.microsoft.com/sharepoint/soap/"><UpdateListItemsResult><Results><Result
ID="1,Update" List="{E074E60A-0126-4446-A751-A5BE7A55ED81}"
Version="4"><ErrorCode>0x00000000</ErrorCode><z:row
ows_ContentTypeId="0x010047334455D6BA5346AC55E03BB37AD578" ows_Title="Test new title"
ows_Num="20.0000000000000" ows_Text="TestTest" ows_DateTime="2008-03-03 15:20:00"
ows_lookup="1;#Admin" ows_ID="1" ows_ContentType="Item" ows_Modified="2008-02-05 10:49:48"
ows_Created="2008-01-28 08:06:11" ows_Author="1;#Admin" ows_Editor="1;#Admin"
ows_owshiddenversion="4" ows_WorkflowVersion="1" ows_UIVersion="512"
ows_UIVersionString="1.0" ows_Attachments="0" ows_ModerationStatus="0"
ows_LinkTitleNoMenu="Test new title" ows_LinkTitle="Test new title" ows_SelectTitle="1"
ows_Order="100.0000000000000" ows_GUID="{2398096C-624B-4FA1-997F-36F6AC4776A2}"
ows_FileRef="1;#Lists/TestListItem/1_000" ows_FileDirRef="1;#Lists/TestListItem"
ows_Last_x0020_Modified="1;#2008-01-28 08:06:11" ows_Created_x0020_Date="1;#2008-01-28
08:06:11" ows_FSObjType="1;#0" ows_PermMask="0x7fffffff" ows_FileLeafRef="1;#1_000"
ows_UniqueId="1;#{078CAB5C-5FEA-43EA-B23A-FB93EF35397D}" ows_ProgId="1;#"
ows_ScopeId="1;#{E1CCC98A-9A01-4193-8AE5-6E6765AC3054}" ows_EditMenuTableStart="1_000"
ows_EditMenuTableEnd="1" ows_LinkFilenameNoMenu="1_000" ows_LinkFilename="1_000"
ows_ServerUrl="/Lists/TestListItem/1_000"
ows_EncodedAbsUrl="http://siteaddress/Lists/TestListItem/1_000" ows_BaseName="1_"
ows_MetaInfo="1;#" ows_Level="1" ows_IsCurrentVersion="1" xmlns:z="#RowsetSchema"
/></Result></Results></UpdateListItemsResult></UpdateListItemsResponse></soap:Body></soap:Env
elope>

```

The following is an example of adding a new folder. Folders are a specific type of list item. To signify that the new item SHOULD be a folder, set **FSObjType** to 1 and set the **BaseName** to the name of the folder.

```

Content-Type: text/xml; charset=utf-8
SOAPAction: "http://schemas.microsoft.com/sharepoint/soap/UpdateListItems"

```

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <UpdateListItems xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>TestListItem</listName>
      <updates>
        <Batch OnError="Continue">
          <Method ID="1" Cmd="New">
            <Field Name="FSObjType">1</Field>
            <Field Name="BaseName">TestCreateNewFolder</Field>
          </Method>
        </Batch>
      </updates>
    </UpdateListItems>
  </soap:Body>
</soap:Envelope>

```

```

    </UpdateListItems>
  </soap:Body>
</soap:Envelope>

```

The following is an example of updating an existing item. The example updates the **Title** and **Number** field for the list item whose identification is 1, as well as adds a new list item:

```

Content-Type: text/xml; charset=utf-8
SOAPAction: "http://schemas.microsoft.com/sharepoint/soap/UpdateListItems"

```

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <UpdateListItems xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>TestListItem</listName>
      <updates>
        <Batch OnError="Continue">
          <Method ID="1" Cmd="Update">
            <Field Name="ID">1</Field>
            <Field Name="Title">ChangeThisItem</Field>
            <Field Name="Number">3</Field>
          </Method>
          <Method ID="2" Cmd="New">
            <Field Name="Title">MakeANewItem</Field>
            <Field Name="Number">12</Field>
          </Method>
        </Batch>
      </updates>
    </UpdateListItems>
  </soap:Body>
</soap:Envelope>

```

3.1.4.31.2.2 UpdateListItemsResponse

This element contains the protocol server response to the **UpdateListItems** request.

```

<s:element name="UpdateListItemsResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="UpdateListItemsResult">
        <s:complexType>
          <s:sequence>
            <s:element name="Results" >
              <s:complexType>
                <s:sequence>
                  <s:element name="Result" maxOccurs="unbounded">
                    <s:complexType>
                      <s:sequence>
                        <s:element name="ErrorCode" type="s:string" />
                        <s:element name="ErrorText" type="s:string" />
                        <s:any />
                      </s:sequence>
                    </s:complexType>
                  </s:element>
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>

```



```

        <s:attribute name="ID" type="s:string" />
        <s:attribute name="List" type="s:string" minOccurs="0" maxOccurs="1" />
        <s:attribute name="Version" type="s:string" minOccurs="0"
maxOccurs="1"/>
        </s:complexType>
    </s:element>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
<s:anyAttribute />
</s:complexType></s:element>

```

- **UpdateListItemsResult:** If the **OnError** attribute of the **Batch** element is set to **Return**, then there **MUST** be a **Result** element for each **Method** operation. If a **Method** operation fails, then all subsequent **Method** operations in the **Batch** **MUST** fail with an error of 0x800704c7. If the **OnError** attribute of the **Batch** element is set to **Continue**, then each **Method** corresponds to a **Result**. The outcome of a Method operation **MUST** not affect the other Method operations. The **ID** attribute of the **Method** parameters **MUST** correspond to the **ID** attribute of the **Result** element and the **Result ID** is the **Method ID**, followed by a comma, followed by the **Method** operation. The **List** and **Version** attributes **MUST** be returned if a Method operation fails with an error of 0x81020015. The **List** attribute **MUST** be the GUID associated with the list on which the operation is finished. The **Version** attribute **MUST** be the schema version of the list and **MUST** correspond to the **ListVersion** that can be returned from a **GetList** call.

The **Result** element **MUST** contain an **ErrorCode** element. The **ErrorCode** element is a string representation of a hexadecimal number indicating whether the operation succeeded or failed. If an operation completes successfully, the value of the **ErrorCode** **MUST** be set to 0x00000000. Otherwise, the **ErrorCode** **MUST** be set to a hexadecimal representation of the error encountered and the **ErrorText** element **MUST** have a description of the error.

If the operation succeeded, the **z:row** element **MUST** contain the updated data for the list item that is operated on. On failure, the **z:row** element **MUST** contain the list item data that exists on the server if the list item specified in the request exists.

Common errors that **MAY** be encountered in an **UpdateListItemsResponse** are specified in the following table.

ErrorCode	Error description
0x81020015	This error code MUST be used to specify a data conflict. The changes requested conflict with those made previously by another protocol client.
0x81020014	This specifies a generic error has been encountered, such as an invalid value being specified for a Field.
0x81020016	This specifies that the list item referred to in the request does not exist.

3.1.4.32 UpdateListItemsWithKnowledge

The **UpdateListItemsWithKnowledge** operation is used to make inserts, updates, and deletions to specified list items in a list. [<68>](#)

```

<wsdl:operation name="UpdateListItemsWithKnowledge">
  <wsdl:input message="UpdateListItemsWithKnowledgeSoapIn" />
  <wsdl:output message="UpdateListItemsWithKnowledgeSoapOut" />
</wsdl:operation>

```

The protocol client sends an **UpdateListItemsWithKnowledgeSoapIn** request message and the protocol server responds with an **UpdateListItemsWithKnowledgeSoapOut** response message, as follows:

1. If the specified **listName** is a valid GUID and corresponds to the identification of a list on the site, use that list.
2. If the specified **listName** is not a valid GUID or does not correspond to the identification of a list on the site, check if the **listName** corresponds to the list title of a list on the site and, if so, use that list.
3. If the specified **listName** is empty, the protocol server MUST return a SOAP fault with no error code.
4. If **listName** does not correspond to a list from either of these checks, the protocol server MUST return a SOAP fault with error code 0x82000006. This indicates that the list does not exist or might have been deleted by another user.
5. Otherwise, the protocol server MUST process the batched operations on the list and return success or failure conditions per operation.

The protocol client MUST use the **UpdateListItemsWithKnowledge** operation to set the **vti_clientid** and **vti_contentchangeunit** Metainfo properties on a list item.

3.1.4.32.1 Messages

The following table summarizes the set of WSDL message definitions that are specific to this operation.

Message	Description
UpdateListItemsWithKnowledgeSoapIn	The request to make inserts, updates, and deletions to specified list items in a list.
UpdateListItemsWithKnowledgeSoapOut	The response to a request to make inserts, updates, and deletions to specified list items in a list.

3.1.4.32.1.1 UpdateListItemsWithKnowledgeSoapIn

This message is the request of the **UpdateListItemsWithKnowledge** operation that the protocol client sends as a request to this protocol server.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/UpdateListItemsWithKnowledge
```

The SOAP Body contains an **UpdateListItemsWithKnowledge** element.

3.1.4.32.1.2 UpdateListItemsWithKnowledgeSoapOut

This message is the response of the **UpdateListItemsWithKnowledge** operation that the protocol client receives from this protocol server.

The SOAP Action value of the message is defined as follows:

```
http://schemas.microsoft.com/sharepoint/soap/UpdateListItemsWithKnowledge
```

The SOAP Body contains an **UpdateListItemsWithKnowledgeResponse** element.

3.1.4.32.2 Elements

The following table summarizes the XML schema element definitions that are specific to this operation.

Element	Description
UpdateListItemsWithKnowledge	The request to make inserts, updates, and deletions to specified list items in a list.
UpdateListItemsWithKnowledgeResponse	Contains the response to a request to make inserts, updates, and deletions to specified list items in a list.

3.1.4.32.2.1 UpdateListItemsWithKnowledge

This element specifies the request that **MUST** be sent for the **UpdateListItemsWithKnowledge** operation.

```
<s:element name="UpdateListItemsWithKnowledge">
  <s:complexType>
    <s:sequence>
      <s:element name="listName" type="s:string" />
      <s:element name="updates">
        <s:complexType>
          <s:sequence>
            <s:element name="Batch">
              <s:complexType>
                <s:sequence>
                  <s:element name="Method" maxOccurs="unbounded">
                    <s:complexType mixed="true">
                      <s:sequence>
                        <s:element name="Field" maxOccurs="unbounded">
                          <s:complexType>
                            <s:simpleContent>
                              <s:extension base="s:string">
                                <s:attribute name="Name" type="s:string"/>
                              </s:extension>
                            </s:simpleContent>
                          </s:complexType>
                        </s:element>
                      </s:sequence>
                    </s:complexType>
                  </s:element>
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
          <s:attribute name="ID" type="s:unsignedInt" use="required" />
          <s:attribute name="Cmd" type="tns:MethodCmdEnum"
            use="required" />
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

```

        </s:element>
      </s:sequence>
      <s:attribute name="ListVersion" type="s:unsignedInt" />
      <s:attribute name="OnError" type="tns:OnErrorEnum" />
      <s:attribute name="Version" type="s:string" />
      <s:attribute name="ViewName" type="s:string" />
      <s:attribute name="PreCalc"
        type="core:TRUE_Case_Sensitive_Else_Anything" />
      <s:attribute name="DateInUtc"
        type="core:TRUE_Case_Sensitive_Else_Anything" />
      <s:attribute name="Properties"
        type="core:TRUE_Case_Sensitive_Else_Anything" />
      <s:attribute name="LockSchema"
        type="core:TRUE_Case_Sensitive_Else_Anything" />
      <s:attribute name="RootFolder" type="s:string" />
    </s:complexType>
  </s:element>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="knowledge" minOccurs="0" maxOccurs="1" nullable="true">
  <s:complexType mixed="true">
    <s:sequence>
      <s:any />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="syncScope" minOccurs="0" maxOccurs="1" type="s:string"
  nullable="true"/>
</s:sequence>
</s:complexType>
</s:element>

```

listName: See the description of the **listName** element in section [3.1.4.31.2.1](#).

updates: See the description of the **updates** element in section [3.1.4.31.2.1](#).

Batch: See the description of the **Batch** element in section [3.1.4.31.2.1](#).

Method: See the description of the **Method** element in section [3.1.4.31.2.1](#).

Field: See the description of the **Field** element in section [3.1.4.31.2.1](#).

knowledge: MUST be null.

syncScope: This parameter is reserved and MUST be ignored.

3.1.4.32.2.2 UpdateListItemsWithKnowledgeResponse

This element contains the protocol server response to the **UpdateListItemsWithKnowledge** request.

```

<s:element name="UpdateListItemsWithKnowledgeResponse">
  <s:complexType>
    <s:sequence>
      <s:element name="UpdateListItemsWithKnowledgeResult">
        <s:complexType>

```

```

<s:sequence>
  <s:element name="Results" >
    <s:complexType>
      <s:sequence>
        <s:element name="Result" maxOccurs="unbounded">
          <s:complexType>
            <s:sequence>
              <s:any />
            </s:sequence>
            <s:attribute name="ID" type="s:string" />
            <s:attribute name="List" type="s:string" minOccurs="0" maxOccurs="1" />
            <s:attribute name="Version" type="s:string" minOccurs="0"
maxOccurs="1"/>
          </s:complexType>
        </s:element>
      </s:sequence>
    </s:complexType>
  </s:element>
</s:sequence>
</s:element>

```

UpdateListItemsWithKnowledgeResult : See description of **UpdateListItemsResult** element in section [3.1.4.31.2.2](#)

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

4.1 New Content Type

Overall scenario: A protocol client wants to discover which content types exist on a protocol server and choose one from which to retrieve all data.

A protocol client that does not know of any content type identifiers on a protocol server uses the **GetListContentTypes** operation to determine which content types exist on the list. The list needs to be known before this protocol can be used, such as by using **GetListCollection**.

A sample request is similar to the following:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetListContentTypes xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>{C7A6517B-BC8E-4937-B405-FC2EB2F88736}</listName>
      <contentTypeId></contentTypeId>
    </GetListContentTypes>
  </soap:Body>
</soap:Envelope>
```

A sample response is similar to the following (white space added for readability):

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetListContentTypesResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <GetListContentTypesResult>
        <ContentTypes>
          <ContentType Name="Document" ID="0x010100953225EF09D1E743BA2FD70C51A612DE"
Description="Create a new document." Scope="http://server-test5/CT Doc Lib" Version="0">
            <XmlDocuments>
              <XmlDocument
NamespaceURI="http://schemas.microsoft.com/sharepoint/v3/contenttype/forms">
                <FormTemplates xmlns="http://schemas.microsoft.com/sharepoint/v3/contenttype/forms">
                  <Display>DocumentLibraryForm</Display>
                  <Edit>DocumentLibraryForm</Edit>
                  <New>DocumentLibraryForm</New>
                </FormTemplates>
              </XmlDocument>
            </XmlDocuments>
          </ContentType>
          <ContentType Name="Folder" ID="0x012000AA08B7BC1F57FD4D80C737A837E5069F"
Description="Create a new folder." Scope="http://server-test5/CT Doc Lib" Version="0">
            <XmlDocuments>
              <XmlDocument
NamespaceURI="http://schemas.microsoft.com/sharepoint/v3/contenttype/forms">
                <FormTemplates xmlns="http://schemas.microsoft.com/sharepoint/v3/contenttype/forms">
                  <Display>ListForm</Display>
                  <Edit>ListForm</Edit>
                  <New>ListForm</New>
                </FormTemplates>
              </XmlDocument>
            </XmlDocuments>
          </ContentType>
        </ContentTypes>
      </GetListContentTypesResult>
    </GetListContentTypesResponse>
  </soap:Body>
</soap:Envelope>
```

```

    </XmlDocuments>
  </ContentType>
  <ContentType Name="Dublin Core Columns" ID="0x01010B00818A37E4A8A29B4E80B97DDDB6F45D8E"
  Description="The Dublin Core metadata element set." Scope="http://server-test5/CT Doc Lib"
  Version="1">
    <XmlDocuments>
      <XmlDocument
        NamespaceURI="http://schemas.microsoft.com/sharepoint/v3/contenttype/forms">
        <FormTemplates xmlns="http://schemas.microsoft.com/sharepoint/v3/contenttype/forms">
          <Display>DocumentLibraryForm</Display>
          <Edit>DocumentLibraryForm</Edit>
          <New>DocumentLibraryForm</New>
        </FormTemplates>
      </XmlDocument>
    </XmlDocuments>
  </ContentType>
</ContentTypes>
</GetListContentTypesResult>
</GetListContentTypesResponse>
</soap:Body>
</soap:Envelope>

```

The response of the server includes three content types: (0x010100953225EF09D1E743BA2FD70C51A612DE, 0x012000AA08B7BC1F57FD4D80C737A837E5069F, and 0x01010B00818A37E4A8A29B4E80B97DDDB6F45D8E). The response can be used to receive more information about the "Dublin Core Columns" by using the **GetListContentType** method.

A sample request for this content type is similar to the following:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetListContentType xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>{C7A6517B-BC8E-4937-B405-FC2EB2F88736}</listName>
      <contentTypeID>0x01010B00818A37E4A8A29B4E80B97DDDB6F45D8E</contentTypeID>
    </GetListContentType>
  </soap:Body>
</soap:Envelope>

```

This request uses the content type identifier that was received from the **GetListContentTypes** method to retrieve information about the "Dublin Core Columns" content type.

A sample response from the server is similar to the following (white spaced added for readability):

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetListContentTypeResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <GetListContentTypeResult>
        <ContentType ID="0x01010B00818A37E4A8A29B4E80B97DDDB6F45D8E" Name="Dublin Core Columns"
        Group="Document Content Types" Description="The Dublin Core metadata element set."
        Version="1" FeatureId="{695b6570-a48b-4a8e-8ea5-26ea7fc1d162}">

```

```

<Folder TargetName="Forms/Dublin Core Columns"/>
<Fields>
  <Field RowOrdinal="0" Type="Choice" Format="Dropdown" FillInChoice="FALSE"
Sealed="FALSE" Name="ContentType" ColName="tp_ContentType"
SourceID="http://schemas.microsoft.com/sharepoint/v3 ID="{c042a256-787d-4a6f-8a8a-
cf6ab767f12d}" DisplayName="Content Type" StaticName="ContentType" Group="_Hidden"
PITarget="MicrosoftWindowsSharePointServices" PIAttribute="ContentTypeID">
  <Default>Document</Default>
  <CHOICES>
    <CHOICE>Document</CHOICE>
    <CHOICE>Folder</CHOICE>
    <CHOICE>Dublin Core Columns</CHOICE>
  </CHOICES>
</Field>
  <Field ID="{5f47e085-2150-41dc-b661-442f3027f552}" ReadOnly="TRUE" Type="Computed"
Name="SelectFilename" DisplayName="Select" Hidden="TRUE" CanToggleHidden="TRUE"
Sortable="FALSE" Filterable="FALSE" AuthoringInfo="(web part connection)"
HeaderImage="blank.gif" SourceID="http://schemas.microsoft.com/sharepoint/v3"
StaticName="SelectFilename" FromBaseType="TRUE">
  <FieldRefs>
    <FieldRef Name="ID"/>
  </FieldRefs>
  <DisplayPattern>
    <IfEqual>
      <Expr1>
        <GetVar Name="SelectedID"/>
      </Expr1>
      <Expr2>
        <Column Name="ID"/>
      </Expr2>
      <Then>
        <HTML><![CDATA[
</HTML>
        <HTML>Selected</HTML>
        <HTML><![CDATA[">]]></HTML>
      </Then>
      <Else>
        <HTML><![CDATA[<a href="javascript:SelectField(')]></HTML>
        <GetVar Name="View"/>
        <HTML><![CDATA[',']]></HTML>
        <ScriptQuote NotAddingQuote="TRUE">
          <Column Name="ID"/>
        </ScriptQuote>
        <HTML><![CDATA[');return false;" onclick="javascript:SelectField(')]></HTML>
        <GetVar Name="View"/>
        <HTML><![CDATA[',']]></HTML>
        <ScriptQuote NotAddingQuote="TRUE">
          <Column Name="ID"/>
        </ScriptQuote>
        <HTML><![CDATA[');return false;" target="_self"]>]]></HTML>
        <HTML><![CDATA[</HTML>
        <HTML>Normal</HTML>
        <HTML><![CDATA[">]]></HTML>
        <HTML><![CDATA[</a>]]></HTML>
      </Else>
    </IfEqual>
  </DisplayPattern>
</Field>

```



```

    <Field ID="{8553196d-ec8d-4564-9861-3dbe931050c8}" ShowInFileDlg="FALSE"
    ShowInVersionHistory="FALSE" Type="File" Name="FileLeafRef" DisplayName="Name"
    AuthoringInfo="(for use in forms)" List="Docs" FieldRef="ID" ShowField="LeafName"
    JoinColName="DoclibRowId" JoinRowOrdinal="0" JoinType="INNER" Required="TRUE"
    SourceID="http://schemas.microsoft.com/sharepoint/v3" StaticName="FileLeafRef"
    FromBaseType="TRUE"/>
    <Field ID="{8c06beca-0777-48f7-91c7-6da68bc07b69}" ColName="tp_Created" RowOrdinal="0"
    ReadOnly="TRUE" Type="DateTime" Name="Created" DisplayName="Created" StorageTZ="TRUE"
    SourceID="http://schemas.microsoft.com/sharepoint/v3" StaticName="Created"
    FromBaseType="TRUE" Hidden="TRUE"/>
    <Field ID="{28cf69c5-fa48-462a-b5cd-27b6f9d2bd5f}" ColName="tp_Modified" RowOrdinal="0"
    ReadOnly="TRUE" Type="DateTime" Name="Modified" DisplayName="Modified" StorageTZ="TRUE"
    SourceID="http://schemas.microsoft.com/sharepoint/v3" StaticName="Modified"
    FromBaseType="TRUE" Hidden="TRUE"/>
    <Field ID="{822c78e3-1ea9-4943-b449-57863ad33ca9}" ReadOnly="TRUE" Hidden="FALSE"
    Type="Text" Name="Modified_x0020_By" DisplayName="Document Modified By"
    SourceID="http://schemas.microsoft.com/sharepoint/v3" StaticName="Modified_x0020_By"
    FromBaseType="TRUE" ColName="nvarchar1"/>
    <Field ID="{4dd7e525-8d6b-4cb4-9d3e-44ee25f973eb}" ReadOnly="TRUE" Hidden="FALSE"
    Type="Text" Name="Created_x0020_By" DisplayName="Document Created By"
    SourceID="http://schemas.microsoft.com/sharepoint/v3" StaticName="Created_x0020_By"
    FromBaseType="TRUE" ColName="nvarchar2"/>
    <Field ID="{370B7779-0344-4b9f-8F2D-DC1C62EAE801}" Type="Note" NumLines="2" Group="Core
    Document Columns" Name="_Contributor" DisplayName="Contributor"
    SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="_Contributor"
    Description="One or more people or organizations that contributed to this resource"
    Customization="" ColName="ntext2" RowOrdinal="0"/>
    <Field ID="{3B1D59C0-26B1-4de6-ABBD-3EDB4E2C6ECA}" Type="Text" Group="Core Document
    Columns" Name="_Coverage" DisplayName="Coverage"
    SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="_Coverage"
    Description="The extent or scope" Customization="" ColName="nvarchar10" RowOrdinal="0"/>
    <Field ID="{246D0907-637C-46b7-9AA0-0BB914DAA832}" Name="_Author" Group="Core Document
    Columns" Type="Text" DisplayName="Creator"
    SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="_Author"
    Description="The primary author" Sealed="TRUE" AllowDeletion="TRUE" ShowInFileDlg="FALSE"
    Customization="" ColName="nvarchar11" RowOrdinal="0"/>
    <Field ID="{9F8B4EEO-84B7-42c6-A094-5CBDE2115EB9}" Type="DateTime" Format="DateTime"
    Group="Core Document Columns" Name="_DCDateCreated" DisplayName="Date Created"
    SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="_DCDateCreated"
    Description="The date on which this resource was created" Customization=""
    ColName="datetime1" RowOrdinal="0"/>
    <Field ID="{810DBD02-BBF5-4c67-B1CE-5AD7C5A512B2}" Type="DateTime" Format="DateTime"
    Group="Core Document Columns" Name="_DCDateModified" DisplayName="Date Modified"
    SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="_DCDateModified"
    Description="The date on which this resource was last modified" Customization=""
    ColName="datetime2" RowOrdinal="0"/>
    <Field ID="{52578FC3-1F01-4f4d-B016-94CCBCF428CF}" Name="_Comments"
    SourceID="http://schemas.microsoft.com/sharepoint/v3" StaticName="_Comments" Group="Core
    Document Columns" Type="Note" DisplayName="Description" Sortable="FALSE" Description="A
    summary of this resource" Sealed="TRUE" UnlimitedLengthInDocumentLibrary="TRUE"
    AllowDeletion="TRUE" ShowInFileDlg="FALSE" Customization="" ColName="ntext3" RowOrdinal="0"/>
    <Field ID="{36111FDD-2C65-41ac-B7EF-48B9B8DA4526}" Type="Text" Group="Core Document
    Columns" Name="_Format" DisplayName="Format"
    SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="_Format"
    Description="Media type, file format or dimensions" Customization="" ColName="nvarchar12"
    RowOrdinal="0"/>
    <Field ID="{3C76805F-AD45-483a-9C85-7AC24506CE1A}" Type="Text" Group="Core Document
    Columns" Name="_Identifier" DisplayName="Resource Identifier"
    SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="_Identifier"
    Description="An identifying string or number, usually conforming to a formal identification
    system" Customization="" ColName="nvarchar13" RowOrdinal="0"/>
    <Field ID="{D81529E8-384C-4ca6-9C43-C86A256E6A44}" Name="Language"
    SourceID="http://schemas.microsoft.com/sharepoint/v3" StaticName="Language" Group="Base

```

Columns" Type="Choice" DisplayName="Language" FillInChoice="TRUE" Customization=""
ColName="nvarchar14" RowOrdinal="0">

<CHOICES>

<CHOICE>Arabic (Saudi Arabia)</CHOICE>
<CHOICE>Bulgarian (Bulgaria)</CHOICE>
<CHOICE>Chinese (Hong Kong S.A.R.)</CHOICE>
<CHOICE>Chinese (PRC)</CHOICE>
<CHOICE>Chinese (Taiwan)</CHOICE>
<CHOICE>Croatian (Croatia)</CHOICE>
<CHOICE>Czech (Czech Republic)</CHOICE>
<CHOICE>Danish (Denmark)</CHOICE>
<CHOICE>Dutch (Netherlands)</CHOICE>
<CHOICE>English</CHOICE>
<CHOICE>Estonian (Estonia)</CHOICE>
<CHOICE>Finnish (Finland)</CHOICE>
<CHOICE>French (France)</CHOICE>
<CHOICE>German (Germany)</CHOICE>
<CHOICE>Greek (Greece)</CHOICE>
<CHOICE>Hebrew (Israel)</CHOICE>
<CHOICE>Hindi (India)</CHOICE>
<CHOICE>Hungarian (Hungary)</CHOICE>
<CHOICE>Indonesian (Indonesia)</CHOICE>
<CHOICE>Italian (Italy)</CHOICE>
<CHOICE>Japanese (Japan)</CHOICE>
<CHOICE>Korean (Korea)</CHOICE>
<CHOICE>Latvian (Latvia)</CHOICE>
<CHOICE>Lithuanian (Lithuania)</CHOICE>
<CHOICE>Malay (Malaysia)</CHOICE>
<CHOICE>Norwegian (Bokmal) (Norway)</CHOICE>
<CHOICE>Polish (Poland)</CHOICE>
<CHOICE>Portuguese (Brazil)</CHOICE>
<CHOICE>Portuguese (Portugal)</CHOICE>
<CHOICE>Romanian (Romania)</CHOICE>
<CHOICE>Russian (Russia)</CHOICE>
<CHOICE>Serbian (Latin) (Serbia)</CHOICE>
<CHOICE>Slovak (Slovakia)</CHOICE>
<CHOICE>Slovenian (Slovenia)</CHOICE>
<CHOICE>Spanish (Spain)</CHOICE>
<CHOICE>Swedish (Sweden)</CHOICE>
<CHOICE>Thai (Thailand)</CHOICE>
<CHOICE>Turkish (Turkey)</CHOICE>
<CHOICE>Ukrainian (Ukraine)</CHOICE>
<CHOICE>Urdu (Islamic Republic of Pakistan)</CHOICE>
<CHOICE>Vietnamese (Vietnam)</CHOICE>

</CHOICES>

<Default>English</Default>

</Field>

<Field ID="{2EEDD0AE-4281-4b77-99BE-68F8B3AD8A7A}" Type="Text" Group="Core Document
Columns" Name="_Publisher" DisplayName="Publisher"
SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="_Publisher"
Description="The person, organization or service that published this resource"
Customization="" ColName="nvarchar15" RowOrdinal="0"/>

<Field ID="{5E75C854-6E9D-405d-B6C1-F8725BAE5822}" Type="Note" NumLines="2" Group="Core
Document Columns" Name="_Relation" DisplayName="Relation"
SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="_Relation"
Description="References to related resources" Customization="" ColName="ntext4"
RowOrdinal="0"/>

<Field ID="{ADA3F0CB-6F95-4588-BB08-D97CC0623522}" Type="Note" NumLines="2" Group="Core
Document Columns" Name="_RightsManagement" DisplayName="Rights Management"
SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="_RightsManagement"

```

Description="Information about rights held in or over this resource" Customization=""
ColName="ntext5" RowOrdinal="0"/>
  <Field ID="{B0A3C1DB-FAF1-48f0-9BE1-47D2FC8CB5D6}" Type="Note" NumLines="2" Group="Core
Document Columns" Name="_Source" DisplayName="Source"
SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="_Source"
Description="References to resources from which this resource was derived" Customization=""
ColName="ntext6" RowOrdinal="0"/>
  <Field ID="{B66E9B50-A28E-469b-B1A0-AF0E45486874}" Type="Note" Group="Core Document
Columns" Name="Keywords" DisplayName="Keywords"
SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="Keywords"
Sealed="TRUE" NumLines="1" UnlimitedLengthInDocumentLibrary="TRUE" AllowDeletion="TRUE"
ShowInFileDlg="FALSE" Customization="" ColName="ntext7" RowOrdinal="0"/>
  <Field ID="{76A81629-44D4-4ce1-8D4D-6D7EBCD885FC}" Name="Subject" Group="Core Document
Columns" Type="Text" DisplayName="Subject"
SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="Subject"
Sealed="TRUE" AllowDeletion="TRUE" ShowInFileDlg="FALSE" Customization=""
ColName="nvarchar16" RowOrdinal="0"/>
  <Field ID="{fa564e0f-0c70-4ab9-b863-0177e6ddd247}" Type="Text" Name="Title"
ShowInNewForm="FALSE" ShowInFileDlg="FALSE" DisplayName="Title" Sealed="TRUE"
SourceID="http://schemas.microsoft.com/sharepoint/v3" StaticName="Title"
ColName="nvarchar7"/>
  <Field ID="{EDEC70-F6E2-4c3c-A4C7-F61A515DFAA9}" Type="Text" Group="Core Document
Columns" Name="ResourceType" DisplayName="Resource Type"
SourceID="http://schemas.microsoft.com/sharepoint/v3/fields" StaticName="ResourceType"
Description="A set of categories, functions, genres or aggregation levels" Customization=""
ColName="nvarchar17" RowOrdinal="0"/>
</Fields>
<DocumentTemplate TargetName="Forms/Dublin Core Columns/template.doc"/>
<XmlDocuments>
  <XmlDocument
NamespaceURI="http://schemas.microsoft.com/sharepoint/v3/contenttype/forms">
  <FormTemplates xmlns="http://schemas.microsoft.com/sharepoint/v3/contenttype/forms">
    <Display>DocumentLibraryForm</Display>
    <Edit>DocumentLibraryForm</Edit>
    <New>DocumentLibraryForm</New>
  </FormTemplates>
  </XmlDocument>
</XmlDocuments>
</ContentType>
</GetListContentTypeResult>
</GetListContentTypeResponse>
</soap:Body>
</soap:Envelope>

```

This response contains all the data about the content type requested.

Assume that the user wants to create a new content type on the list "Tokyo Project" for documents relating to travel expenses for the Tokyo project. The user wants this content type to be called "Tokyo Travel Expenses" with the description "Document related to travel expenses for the Tokyo Project" and wants the content type to inherit from the existing site content type "Travel Document", which has content type identifier 0x010100D0C022956380194BB5407321762A3E3F. The protocol client makes the following request:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <CreateContentType xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>Tokyo Project</listName>

```

```

    <displayName>Tokyo Travel Expenses</displayName>
    <parentType>0x010100D0C022956380194BB5407321762A3E3F</parentType>
    <fields></fields>
    <contentTypeProperties>
      <ContentType Title="Tokyo Travel Expenses" Description="Document related to travel
expenses for the Tokyo Project" />
    </contentTypeProperties>
  </CreateContentType>
</soap:Body>
</soap:Envelope>

```

The protocol client receives the following response from the protocol server:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <CreateContentTypeResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <CreateContentTypeResult>0x010100D0C022956380194BB5407321762A3E3F007604A508847BC04E8FB0A9FE36
BCA8CA</CreateContentTypeResult></CreateContentTypeResponse>
    </soap:Body>
  </soap:Envelope>

```

Assume that the user has already created the column "Cost in Yen" (which has internal name `Cost_x0020_in_x0020_Yen`) on the "Tokyo Project" list, and that the user wants this field to be included in the content type. The user decides that the more general column "Cost" (which has internal name `Cost0`) inherited from the parent content type SHOULD be removed. By using the content type identifier from the previous response, the protocol client makes the following request:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <UpdateContentType xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>Tokyo Project</listName>
      <contentTypeId>0x010100D0C022956380194BB5407321762A3E3F007604A508847BC04E8FB0A9FE36BCA8CA</co
ntentTypeId>
      <contentTypeProperties>
        <ContentType Title="Tokyo Travel Expenses"/>
      </contentTypeProperties>
      <newFields>
        <Fields>
          <Method ID="Insert">
            <Field Name="Cost_x0020_in_x0020_Yen">
              </Field>
            </Method>
          </Fields>
        </newFields>
      <deleteFields>
        <Fields>
          <Method ID="Delete">

```

```

        <Field Name="Cost0">
        </Field>
    </Method>
</Fields>
</deleteFields>
</UpdateContentType>
</soap:Body>
</soap:Envelope>

```

The protocol client receives the following response from the protocol server:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <UpdateContentTypeResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <UpdateContentTypeResult>
        <Results>
          <Method ID="Delete">
            <ErrorCode>0x00000000</ErrorCode>
          </Method>
          <Method ID="Insert">
            <ErrorCode>0x00000000</ErrorCode>
            <FieldRef ID="{fea344e3-3c7c-44a3-abd0-36dd45a1684a}"
Name="Cost_x0020_in_x0020_Yen" Required="FALSE" Hidden="FALSE" Customization=""
ReadOnly="FALSE" PITarget="" PrimaryPITarget="" PIAttribute="" PrimaryPIAttribute=""
Aggregation="" Node="" />
          </Method>
          <ListProperties
ID="0x010100D0C022956380194BB5407321762A3E3F007604A508847BC04E8FB0A9FE36BCA8CA" Name="Tokyo
Travel Expenses" Group="Custom Content Types" Description="Document related to travel
expenses for the Tokyo Project" Version="6" />
          </Results>
        </UpdateContentTypeResult>
      </UpdateContentTypeResponse>
    </soap:Body>
  </soap:Envelope>

```

Assume that a document library named "Test Document Library" contains a content type with identification 0x0101005B1A76E874B25B49A1D942FE13273862.

A protocol client wants to add some custom information in the XML Document to the list content type by using the **UpdateContentTypesXmlDocument** operation.

A sample request for **UpdateContentTypesXmlDocument** looks similar to the following:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <UpdateContentTypesXmlDocument xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>Test Document Library</listName>
      <newDocument>

```

```

    <ContentTypes xmlns="">
      <ContentType ID="0x0101005B1A76E874B25B49A1D942FE13273862">
        <customInfo xmlns="http://www.contoso.com/customInfo">Here is some custom
information</customInfo>
      </ContentType>
    </ContentTypes>
  </newDocument>
</UpdateContentTypesXmlDocument>
</soap:Body>
</soap:Envelope>

```

A sample response for **UpdateContentTypesXmlDocument** looks similar to the following:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <UpdateContentTypesXmlDocumentResponse
xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <UpdateContentTypesXmlDocumentResult>
        <Success/>
      </UpdateContentTypesXmlDocumentResult>
    </UpdateContentTypesXmlDocumentResponse>
  </soap:Body>
</soap:Envelope>

```

A protocol client wants to add some custom information in the XML document to the list content type. This can be done by calling the **UpdateContentTypeXmlDocument** operation.

A sample request for **UpdateContentTypeXmlDocument** looks similar to the following:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <UpdateContentTypeXmlDocument xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>Test Document Library</listName>
      <contentTypeId>
0x0101005B1A76E874B25B49A1D942FE13273862</contentTypeId>
      <newDocument>
        <customInfo xmlns="http://www.contoso.com/customInfo">Here is some custom
information</customInfo>
      </newDocument>
    </UpdateContentTypeXmlDocument>
  </soap:Body>
</soap:Envelope>

```

A sample response for **UpdateContentTypeXmlDocument** looks similar to the following:

```

<?xml version="1.0" encoding="utf-8"?>

```

```

<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <UpdateContentTypeXmlDocumentResponse
xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <UpdateContentTypeXmlDocumentResult>
        <Success/>
      </UpdateContentTypeXmlDocumentResult>
    </UpdateContentTypeXmlDocumentResponse>
  </soap:Body>
</soap:Envelope>

```

After the custom information is added, the XML document of the list looks similar to the following:

```

<XmlDocument NamespaceURI="http://www.contoso.com/customInfo">
  <customInfo xmlns="http://www.contoso.com/customInfo">Here is some custom
information</customInfo>
</XmlDocument>

```

The protocol client can delete the custom information by calling **DeleteContentTypeXmlDocument** operation.

A sample request for **DeleteContentTypeXmlDocument** looks similar to the following:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <DeleteContentTypeXmlDocument xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>Test Document Library</listName>
      <contentTypeId>
0x0101005B1A76E874B25B49A1D942FE13273862</contentTypeId>
      <documentUri>http://www.contoso.com/customInfo</documentUri>
    </DeleteContentTypeXmlDocument>
  </soap:Body>
</soap:Envelope>

```

A sample response for **DeleteContentTypeXmlDocument** looks similar to the following:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <DeleteContentTypeXmlDocumentResponse
xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <DeleteContentTypeXmlDocumentResult>
        <Success/>
      </DeleteContentTypeXmlDocumentResult>
    </DeleteContentTypeXmlDocumentResponse>
  </soap:Body>

```

```
</soap:Envelope>
```

4.2 Create and Update a List

In this scenario, the user wants to create and update a new list named "ListOfColors" and use the **CustomList** template to create this list. The user looks up the information and finds out that the list template identifier for the **CustomList** template (for more information, see [\[MS-WSSFO2\]](#) section 2.2.3.2) is 100. The user further wants to get a change token before adding any rows so the user can retrieve all data updates that have happened to "ListOfColors" at a later point. The user then wants to add three new rows: one row that has the title field set to "red", a second row that has the title field set to "blue", and a third row that has the title field set to "green". Finally, the user wants to retrieve all data updates to the list that have happened after the list was created.

The steps for this task are as follows (note that any ... in the following examples indicate that part of the response was left out to improve readability):

The protocol client issues an **AddList** command as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <AddList xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>ListOfColors</listName>
      <templateID>100</templateID>
    </AddList>
  </soap:Body>
</soap:Envelope>
```

The server responds with an **AddList** response as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <AddListResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <AddListResult>
        <List ...> . . .
      </List>
    </AddListResult>
  </AddListResponse>
</soap:Body>
</soap:Envelope>
```

The protocol client issues a **GetListItemChangesSinceToken**, without a token, to get the latest change token for use later, as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
```



```

<GetListItemChangesSinceToken xmlns="http://schemas.microsoft.com/sharepoint/soap/">
  <listName>ListOfColors</listName>
  <viewName></viewName>
  <query>
    <Query>
      <OrderBy>
        <FieldRef Name="ID"/>
      </OrderBy>
    </Query>
  </query>
  <viewFields>
    <ViewFields>
      <FieldRef Name="ID" />
      <FieldRef Name="Title" />
    </ViewFields>
  </viewFields>
  <rowLimit>10</rowLimit>
  <queryOptions>
    <Query/>
  </queryOptions>
</GetListItemChangesSinceToken>
</soap:Body>
</soap:Envelope>

```

The server responds with a **GetListItemChangesSinceTokenResponse** as follows:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-
  instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetListItemChangesSinceTokenResponse xmlns="http://schemas.microsoft.com/sharepoint/
  soap/">
      <GetListItemChangesSinceTokenResult>
        <listitems MinTimeBetweenSyncs='0' RecommendedTimeBetweenSyncs='180'
  MaxBulkDocumentSyncSize='500' AlternateUrls='http://www.tailspintoys.com/'
  EffectivePermMask='FullMask' xmlns:s='uuid:BDC6E3F0-6DA3-11d1-A2A3-00AA00C14882'
  xmlns:dt='uuid:C2F41010-65B3-11d1-A29F-00AA00C14882'
  xmlns:rs='urn:schemas-microsoft-com:rowset'
  xmlns:z='#RowsetSchema'>
          <Changes LastChangeToken="1;3;7ffc2f13-9163-4e9f-bf4e-
  a3f34d882e8f;633386310765830000;799">
            <List ...> // deleted because of size
            ...
          </List>
        </Changes>
        <rs:data ItemCount="0">
          </rs:data>
        </listitems>
      </GetListItemChangesSinceTokenResult>
    </GetListItemChangesSinceTokenResponse>
  </soap:Body>
</soap:Envelope>

```

The protocol client issues an **UpdateListItems** request to add new rows as follows:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <UpdateListItems xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>ListOfColors</listName>
      <updates>
        <Batch OnError="Continue">
          <Method ID="1" Cmd="New">
            <Field Name="ID">New</Field>
            <Field Name="Title">Red</Field>
          </Method>
          <Method ID="2" Cmd="New">
            <Field Name="ID">New</Field>
            <Field Name="Title">Blue</Field>
          </Method>
          <Method ID="3" Cmd="New">
            <Field Name="ID">New</Field>
            <Field Name="Title">Green</Field>
          </Method>
        </Batch>
      </updates>
    </UpdateListItems>
  </soap:Body>
</soap:Envelope>

```

The server responds with an **UpdateListItemsResponse** as follows:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <UpdateListItemsResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <UpdateListItemsResult>
        <Results>
          <Result ID="1,New">
            <ErrorCode>0x00000000</ErrorCode>
            <ID />
            <z:row ows_ID="1" ows_ContentTypeId="0x0100C6549B8AFD10DD409B9ED11D
66B944E3" ows_ContentType="Item" ows_Title="Red" ows_Modified="2008-02-14 16:13:02"
ows_Created="2008-02-14 16:13:02" ows_Author="1;#Lori Kane" ows_Editor="1;#Lori Kane"
ows_owshiddenversion="1" ows_WorkflowVersion="1" ows_UIVersion="512"
ows_UIVersionString="1.0" ows_Attachments="0" ows_ModerationStatus="0"
ows_LinkTitleNoMenu="Red" ows_LinkTitle="Red" ows_SelectTitle="1"
ows_Order="100.000000000000" ows_GUID="{9E943591-254C-4B61-AE10-3504
454D96A2}" ows_FileRef="1;#Lists/ListOfColors/1_.000" ows_FileDirRef="1;#Lists/ListOfColors"
ows_Last_x0020_Modified="1;#2008-02-14 16:13:02" ows_Created_x0020_Date="1;#2008-02-14
16:13:02"
ows_FSObjType="1;#0" ows_PermMask="0x7fffffff"
ows_FileLeafRef="1;#1_.000" ows_UniqueId="1;#{DC043241-B825-4B59-8A77-ED4C3F71B186}"
ows_ProgId="1;#"
ows_ScopeId="1;#{E274659B-1C41-4FE7-910D-551CC39D583B}" ows_EditMenuTableStart="1_.000"
ows_EditMenuTableEnd="1" ows_LinkFilenameNoMenu="1_.000" ows_LinkFilename="1_.000"
ows_ServerUrl="/Lists/ListOfColors/1_.000"
ows_EncodedAbsUrl="http://www.tailspintoy.com/Lists/ListOfColors/1_.000" ows_BaseName="1_"
ows_

```

```

MetaInfo="1;#" ows_Level="1" ows_IsCurrentVersion="1" xmlns:z="#RowsetSchema" />
</Result>
<Result ID="2,New">
  <ErrorCode>0x00000000</ErrorCode>
  <ID />
  <z:row ows_ID="2" ows_ContentTypeId="0x0100C6549B8AFD10DD409B9ED11D66B944E3"
ows_ContentType="Item" ows_Title="Blue" ows_Modified="2008-02-14 16:13:02" o
ws_Created="2008-02-14 16:13:02" ows_Author="1;#Lori Kane" ows_Editor="1;#Lori Kane"
ows_owshiddenversion="1" ows_WorkflowVersion="1" ows_UIVers
ion="512" ows_UIVersionString="1.0" ows_Attachments="0" ows_ModerationStatus="0"
ows_LinkTitleNoMenu="Blue" ows_LinkTitle="Blue" ows_SelectTitle="2"
ows_Order="200.000000000000" ows_GUID="{7B3A8798-55BF-442D-9211-B1BF358FFFD9}"
ows_FileRef="2;#Lists/ListOfColors/2_.000" ows_FileDirRef="2;#Lists/List
OfColors" ows_Last_x0020_Modified="2;#2008-02-14 16:13:02" ows_Created_x0020_Date="2;#2008-
02-14 16:13:02" ows_FSObjType="2;#0" ows_PermMask="0x7ffffff
ffffffff" ows_FileLeafRef="2;#2_.000" ows_UniqueId="2;#{6445B756-366D-4527-AF2B-
31199DAA4D36}" ows_ProgId="2;#" ows_ScopeId="2;#{E274659B-1C41-4FE7-91
0D-551CC39D583B}" ows_EditMenuTableStart="2_.000" ows_EditMenuTableEnd="2"
ows_LinkFilenameNoMenu="2_.000" ows_LinkFilename="2_.000" ows_ServerUrl="/
Lists/ListOfColors/2_.000"
ows_EncodedAbsUrl="http://www.tailspintoys.com/Lists/ListOfColors/2_.000" ows_BaseName="2_"
ows_MetaInfo="2;#" ows_Level="1" ows_IsCu
rrentVersion="1" xmlns:z="#RowsetSchema" /></Result><Result
ID="3,New"><ErrorCode>0x00000000</ErrorCode><ID /><z:row ows_ID="3" ows_ContentTypeId="0x01
00C6549B8AFD10DD409B9ED11D66B944E3" ows_ContentType="Item" ows_Title="Green"
ows_Modified="2008-02-14 16:13:02" ows_Created="2008-02-14 16:13:02" ows_A
uthor="1;#Lori Kane" ows_Editor="1;#Lori Kane" ows_owshiddenversion="1"
ows_WorkflowVersion="1" ows_UIVersion="512" ows_UIVersionString="1.0" o
ws_Attachments="0" ows_ModerationStatus="0" ows_LinkTitleNoMenu="Green"
ows_LinkTitle="Green" ows_SelectTitle="3" ows_Order="300.000000000000" ows_GUI
D="{8A156CFF-0CBA-41A4-BAA1-C6696847F8BD}" ows_FileRef="3;#Lists/ListOfColors/3_.000"
ows_FileDirRef="3;#Lists/ListOfColors" ows_Last_x0020_Modified="3
;#2008-02-14 16:13:02" ows_Created_x0020_Date="3;#2008-02-14 16:13:02" ows_FSObjType="3;#0"
ows_PermMask="0x7ffffff
0" ows_UniqueId="3;#{1ADFAB90-5569-4CEC-BB73-39D43720AC51}" ows_ProgId="3;#"
ows_ScopeId="3;#{E274659B-1C41-4FE7-910D-551CC39D583B}" ows_EditMenuTable
Start="3_.000" ows_EditMenuTableEnd="3" ows_LinkFilenameNoMenu="3_.000"
ows_LinkFilename="3_.000" ows_ServerUrl="/Lists/ListOfColors/3_.000" ows_Enco
dedAbsUrl="http://www.tailspintoys.com/Lists/ListOfColors/3_.000" ows_BaseName="3_"
ows_MetaInfo="3;#" ows_Level="1" ows_IsCurrentVersion="1" xmlns:z="#RowsetSch
ema" />
</Result>
</Results>
</UpdateListItemsResult>
</UpdateListItemsResponse>
</soap:Body>
</soap:Envelope>

```

The protocol client issues another **GetListItemChangesSinceToken** request, this time with a **changeToken** parameter, to get all changes after that **changeToken** was issued, as follows:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:soap="http://schemas.xmlsoap.or
g/soap/envelope/">
  <soap:Body>
    <GetListItemChangesSinceToken xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <listName>ListOfColors</listName>

```

```

<changeToken>
  1;3;0746ca42-e21e-4d24-a713-f5de94f8189f;633386250004270000;779
</changeToken>
<viewName></viewName>
<query>
  <Query>
    <OrderBy>
      <FieldRef Name="ID"/>
    </OrderBy>
  </Query>
</query>
<viewFields>
  <ViewFields>
    <FieldRef Name="ID" />
    <FieldRef Name="Title" />
  </ViewFields>
</viewFields>
<rowLimit>10</rowLimit>
<queryOptions>
  <Query/>
</queryOptions>
</GetListItemChangesSinceToken>
</soap:Body>
</soap:Envelope>

```

The server responds with a **GetListItemChangesSinceTokenResponse** containing the three rows just added, as follows:

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetListItemChangesSinceTokenResponse
      xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <GetListItemChangesSinceTokenResult>
        <listitems MinTimeBetweenSyncs='0' RecommendedTimeBetweenSyncs='180'
          MaxBulkDocumentSyncSize='500' AlternateUrls='http://www.tailspintoys.com/'
          EffectivePermMask='FullMask' xmlns:s='uuid:BDC6E3F0-6DA3-11d1-A2A3-00AA00C14882'
          xmlns:dt='uuid:C2F41010-65B3-11d1-A29F-00AA00C14882'
          xmlns:rs='urn:schemas-microsoft-com:rowset'
          xmlns:z='#RowsetSchema'>
          <Changes LastChangeToken="1;3;7ffc2f13-9163-4e9f-bf4e-
            a3f34d882e8f;633386995628530000;803">
            </Changes>
            <rs:data ItemCount="3">
              <z:row_ows_ID='1' ows_Title='Red' ows_MetaInfo='1;#' ows_ModerationStatus='0'
                ows_Level='1' ows_owshiddenversion='1' ows_UniqueId='1;#{DC043241-B825-4B59-8A77-
                ED4C3F71B186}' ows_FSObjType='1;#0' ows_Created='2008-02-14 16:13:02'
                ows_FileRef='1;#Lists/ListOfColors/1_.000' />
              <z:row_ows_ID='2' ows_Title='Blue' ows_MetaInfo='2;#' ows_ModerationStatus='0'
                ows_Level='1' ows_owshiddenversion='1' ows_UniqueId='2;#{6445B756-366D-4527-AF2B-
                31199DAA4D36}' ows_FSObjType='2;#0' ows_Created='2008-02-14 16:13:02'
                ows_FileRef='2;#Lists/ListOfColors/2_.000' />
              <z:row_ows_ID='3' ows_Title='Green' ows_MetaInfo='3;#' ows_ModerationStatus='0'
                ows_Level='1' ows_owshiddenversion='1' ows_UniqueId='3;#{1ADFAB90-5569-4CEC-BB73-
                39D43720AC51}' ows_FSObjType='3;#0' ows_Created='2008-02-14 16:13:02'
                ows_FileRef='3;#Lists/ListOfColors/3_.000' />
            </rs:data>
          </listitems>
        </GetListItemChangesSinceTokenResult>
      </GetListItemChangesSinceTokenResponse>
    </soap:Body>
  </soap:Envelope>

```

```
</listitems>  
</GetListItemChangesSinceTokenResult>  
</GetListItemChangesSinceTokenResponse>  
</soap:Body>  
</soap:Envelope>
```

Preliminary

5 Security

5.1 Security Considerations for Implementers

This protocol introduces no additional security considerations beyond those applicable to its underlying protocols.

5.2 Index of Security Parameters

None.

Preliminary

6 Appendix A: Full WSDL

For ease of implementation, the full WSDL and schema are provided in this appendix.

A link to wsscaml.xsd denotes a reference to schemas described in [\[MS-WSSCAML\]](#).

```
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tm="http://microsoft.com/wsdl/mime/textMatching/"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:mime="http://schemas.xmlsoap.org/wsdl/mime/"
xmlns:tns="http://schemas.microsoft.com/sharepoint/soap/"
xmlns:s1="http://microsoft.com/wsdl/types/" xmlns:s="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
targetNamespace="http://schemas.microsoft.com/sharepoint/soap/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <s:schema elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/sharepoint/soap/">
      <s:import namespace="http://www.w3.org/2001/XMLSchema" />
      <s:import namespace="http://microsoft.com/wsdl/types/" />
      <s:element name="GetList">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetListResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetListResult">
              <s:complexType mixed="true">
                <s:sequence>
                  <s:any />
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetListAndView">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="viewName" type="s:string" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetListAndViewResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetListAndViewResult">
              <s:complexType mixed="true">
                <s:sequence>
                  <s:any />
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:schema>
  </wsdl:types>

```

```

        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="DeleteList">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="listName" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="DeleteListResponse">
    <s:complexType />
</s:element>
<s:element name="AddList">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="listName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="description" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="templateID" type="s:int" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="AddListResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="AddListResult">
                <s:complexType mixed="true">
                    <s:sequence>
                        <s:any />
                    </s:sequence>
                </s:complexType>
            </s:element>
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="AddListFromFeature">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="listName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="description" type="s:string" />
            <s:element minOccurs="1" maxOccurs="1" name="featureID" type="s1:guid" />
            <s:element minOccurs="1" maxOccurs="1" name="templateID" type="s:int" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="AddListFromFeatureResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="AddListFromFeatureResult">
                <s:complexType mixed="true">
                    <s:sequence>
                        <s:any />
                    </s:sequence>
                </s:complexType>
            </s:element>
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="UpdateList">

```



```

<s:complexType>
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
    <s:element minOccurs="0" maxOccurs="1" name="listProperties">
      <s:complexType mixed="true">
        <s:sequence>
          <s:any />
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element minOccurs="0" maxOccurs="1" name="newFields">
      <s:complexType mixed="true">
        <s:sequence>
          <s:any />
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element minOccurs="0" maxOccurs="1" name="updateFields">
      <s:complexType mixed="true">
        <s:sequence>
          <s:any />
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element minOccurs="0" maxOccurs="1" name="deleteFields">
      <s:complexType mixed="true">
        <s:sequence>
          <s:any />
        </s:sequence>
      </s:complexType>
    </s:element>
    <s:element minOccurs="0" maxOccurs="1" name="listVersion" type="s:string" />
  </s:sequence>
</s:complexType>
</s:element>
<s:element name="UpdateListResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="UpdateListResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetListCollection">
  <s:complexType />
</s:element>
<s:element name="GetListCollectionResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetListCollectionResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>

```

```

        </s:complexType>
    </s:element>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetListItems">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="viewName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="query">
                <s:complexType mixed="true">
                    <s:sequence>
                        <s:any />
                    </s:sequence>
                </s:complexType>
            </s:element>
            <s:element minOccurs="0" maxOccurs="1" name="viewFields">
                <s:complexType mixed="true">
                    <s:sequence>
                        <s:any />
                    </s:sequence>
                </s:complexType>
            </s:element>
            <s:element minOccurs="0" maxOccurs="1" name="rowLimit" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="queryOptions">
                <s:complexType mixed="true">
                    <s:sequence>
                        <s:any />
                    </s:sequence>
                </s:complexType>
            </s:element>
            <s:element minOccurs="0" maxOccurs="1" name="webID" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetListItemsResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetListItemsResult">
                <s:complexType mixed="true">
                    <s:sequence>
                        <s:any />
                    </s:sequence>
                </s:complexType>
            </s:element>
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="GetListItemChanges">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="viewFields">
                <s:complexType mixed="true">
                    <s:sequence>
                        <s:any />
                    </s:sequence>
                </s:complexType>
            </s:element>
        </s:sequence>
    </s:complexType>

```

```

</s:element>
<s:element minOccurs="0" maxOccurs="1" name="since" type="s:string" />
<s:element minOccurs="0" maxOccurs="1" name="contains">
  <s:complexType mixed="true">
    <s:sequence>
      <s:any />
    </s:sequence>
  </s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetListItemChangesResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetListItemChangesResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetListItemChangesWithKnowledge">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="viewName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="query">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element minOccurs="0" maxOccurs="1" name="viewFields">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element minOccurs="0" maxOccurs="1" name="rowLimit" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="queryOptions">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element minOccurs="0" maxOccurs="1" name="syncScope" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="knowledge">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>

```

```

    </s:element>
    <s:element minOccurs="0" maxOccurs="1" name="contains">
      <s:complexType mixed="true">
        <s:sequence>
          <s:any />
        </s:sequence>
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>
</s:element>
<s:element name="GetListItemChangesWithKnowledgeResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1"
name="GetListItemChangesWithKnowledgeResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetListItemChangesSinceToken">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="viewName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="query">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element minOccurs="0" maxOccurs="1" name="viewFields">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element minOccurs="0" maxOccurs="1" name="rowLimit" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="queryOptions">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element minOccurs="0" maxOccurs="1" name="changeToken" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="contains">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>

```

```

        </s:element>
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:element name="GetListItemChangesSinceTokenResponse">
    <s:complexType>
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1" name="GetListItemChangesSinceTokenResult">
          <s:complexType mixed="true">
            <s:sequence>
              <s:any />
            </s:sequence>
          </s:complexType>
        </s:element>
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:element name="UpdateListItems">
    <s:complexType>
      <s:sequence>
        <s:element name="listName" type="s:string" />
        <s:element name="updates">
          <s:complexType mixed="true">
            <s:sequence>
              <s:any />
            </s:sequence>
          </s:complexType>
        </s:element>
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:element name="UpdateListItemsResponse">
    <s:complexType>
      <s:sequence>
        <s:element minOccurs="0" maxOccurs="1" name="UpdateListItemsResult">
          <s:complexType mixed="true">
            <s:sequence>
              <s:any />
            </s:sequence>
          </s:complexType>
        </s:element>
      </s:sequence>
    </s:complexType>
  </s:element>
  <s:element name="UpdateListItemsWithKnowledge">
    <s:complexType>
      <s:sequence>
        <s:element name="listName" type="s:string" />
        <s:element name="updates">
          <s:complexType mixed="true">
            <s:sequence>
              <s:any />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:element minOccurs="0" maxOccurs="1" name="syncScope" type="s:string" />
        <s:element minOccurs="0" maxOccurs="1" name="knowledge">
          <s:complexType mixed="true">
            <s:sequence>

```

```

        <s:any />
      </s:sequence>
    </s:complexType>
  </s:element>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="UpdateListItemsWithKnowledgeResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="UpdateListItemsWithKnowledgeResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="AddDiscussionBoardItem">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="message" type="s:base64Binary" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="AddDiscussionBoardItemResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="AddDiscussionBoardItemResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="AddWikiPage">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="strListName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="listRelPageUrl" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="wikiContent" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="AddWikiPageResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="AddWikiPageResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>

```



```

        </s:sequence>
        </s:complexType>
        </s:element>
    </s:sequence>
</s:complexType>
</s:element>
<s:element name="DeleteAttachment">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="listItemID" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="url" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="DeleteAttachmentResponse">
    <s:complexType />
</s:element>
<s:element name="CheckOutFile">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="pageUrl" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="checkoutToLocal" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="lastmodified" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="CheckOutFileResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="CheckOutFileResult" type="s:boolean"
/>
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="UndoCheckOut">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="pageUrl" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="UndoCheckOutResponse">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="UndoCheckOutResult" type="s:boolean"
/>
        </s:sequence>
    </s:complexType>
</s:element>
<s:element name="CheckInFile">
    <s:complexType>
        <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="pageUrl" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="comment" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="CheckinType" type="s:string" />
        </s:sequence>
    </s:complexType>
</s:element>

```



```

<s:element name="CheckInFileResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="CheckInFileResult" type="s:boolean"
/>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetListContentTypes">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="contentTypeId" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetListContentTypesResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetListContentTypesResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetListContentTypesAndProperties">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="contentTypeId" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="propertyPrefix" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="includeWebProperties"
type="s:boolean" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetListContentTypesAndPropertiesResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1"
name="GetListContentTypesAndPropertiesResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetListContentType">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="contentTypeId" type="s:string" />

```

```

    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="GetListContentTypeResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetListContentTypeResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="CreateContentType">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="displayName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="parentType" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="fields">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element minOccurs="0" maxOccurs="1" name="contentTypeProperties">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element minOccurs="0" maxOccurs="1" name="addToView" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="CreateContentTypeResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="CreateContentTypeResult"
type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="UpdateContentType">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="contentTypeid" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="contentTypeProperties">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>

```

```

</s:element>
<s:element minOccurs="0" maxOccurs="1" name="newFields">
  <s:complexType mixed="true">
    <s:sequence>
      <s:any />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element minOccurs="0" maxOccurs="1" name="updateFields">
  <s:complexType mixed="true">
    <s:sequence>
      <s:any />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element minOccurs="0" maxOccurs="1" name="deleteFields">
  <s:complexType mixed="true">
    <s:sequence>
      <s:any />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element minOccurs="0" maxOccurs="1" name="addToView" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
<s:element name="UpdateContentTypeResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="UpdateContentTypeResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="DeleteContentType">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="contentTypeid" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="DeleteContentTypeResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="DeleteContentTypeResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>

```

```

</s:element>
<s:element name="UpdateContentTypeXmlDocument">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="contentTypeid" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="newDocument">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="UpdateContentTypeXmlDocumentResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="UpdateContentTypeXmlDocumentResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:all />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="UpdateContentTypesXmlDocument">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="newDocument">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="UpdateContentTypesXmlDocumentResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1"
name="UpdateContentTypesXmlDocumentResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="DeleteContentTypeXmlDocument">
  <s:complexType>

```

```

    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="contentTypeId" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="documentUri" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="DeleteContentTypeXmlDocumentResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="DeleteContentTypeXmlDocumentResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="ApplyContentTypeToList">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="webUrl" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="contentTypeId" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
<s:element name="ApplyContentTypeToListResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="ApplyContentTypeToListResult">
        <s:complexType mixed="true">
          <s:sequence>
            <s:any />
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
</s:schema>
<s:schema elementFormDefault="qualified"
targetNamespace="http://microsoft.com/wsdl/types/">
  <s:simpleType name="guid">
    <s:restriction base="s:string">
      <s:pattern value="[0-9a-fA-F]{8}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-9a-fA-F]{4}-[0-
9a-fA-F]{12}" />
    </s:restriction>
  </s:simpleType>
</s:schema>
</wsdl:types>
<wsdl:message name="GetListSoapIn">
  <wsdl:part name="parameters" element="tns:GetList" />
</wsdl:message>
<wsdl:message name="GetListSoapOut">
  <wsdl:part name="parameters" element="tns:GetListResponse" />
</wsdl:message>

```

```

<wsdl:message name="GetListAndViewSoapIn">
  <wsdl:part name="parameters" element="tns:GetListAndView" />
</wsdl:message>
<wsdl:message name="GetListAndViewSoapOut">
  <wsdl:part name="parameters" element="tns:GetListAndViewResponse" />
</wsdl:message>
<wsdl:message name="DeleteListSoapIn">
  <wsdl:part name="parameters" element="tns>DeleteList" />
</wsdl:message>
<wsdl:message name="DeleteListSoapOut">
  <wsdl:part name="parameters" element="tns>DeleteListResponse" />
</wsdl:message>
<wsdl:message name="AddListSoapIn">
  <wsdl:part name="parameters" element="tns:AddList" />
</wsdl:message>
<wsdl:message name="AddListSoapOut">
  <wsdl:part name="parameters" element="tns:AddListResponse" />
</wsdl:message>
<wsdl:message name="AddListFromFeatureSoapIn">
  <wsdl:part name="parameters" element="tns:AddListFromFeature" />
</wsdl:message>
<wsdl:message name="AddListFromFeatureSoapOut">
  <wsdl:part name="parameters" element="tns:AddListFromFeatureResponse" />
</wsdl:message>
<wsdl:message name="UpdateListSoapIn">
  <wsdl:part name="parameters" element="tns:UpdateList" />
</wsdl:message>
<wsdl:message name="UpdateListSoapOut">
  <wsdl:part name="parameters" element="tns:UpdateListResponse" />
</wsdl:message>
<wsdl:message name="GetListCollectionSoapIn">
  <wsdl:part name="parameters" element="tns:GetListCollection" />
</wsdl:message>
<wsdl:message name="GetListCollectionSoapOut">
  <wsdl:part name="parameters" element="tns:GetListCollectionResponse" />
</wsdl:message>
<wsdl:message name="GetListItemsSoapIn">
  <wsdl:part name="parameters" element="tns:GetListItems" />
</wsdl:message>
<wsdl:message name="GetListItemsSoapOut">
  <wsdl:part name="parameters" element="tns:GetListItemsResponse" />
</wsdl:message>
<wsdl:message name="GetListItemChangesSoapIn">
  <wsdl:part name="parameters" element="tns:GetListItemChanges" />
</wsdl:message>
<wsdl:message name="GetListItemChangesSoapOut">
  <wsdl:part name="parameters" element="tns:GetListItemChangesResponse" />
</wsdl:message>
<wsdl:message name="GetListItemChangesWithKnowledgeSoapIn">
  <wsdl:part name="parameters" element="tns:GetListItemChangesWithKnowledge" />
</wsdl:message>
<wsdl:message name="GetListItemChangesWithKnowledgeSoapOut">
  <wsdl:part name="parameters" element="tns:GetListItemChangesWithKnowledgeResponse" />
</wsdl:message>
<wsdl:message name="GetListItemChangesSinceTokenSoapIn">
  <wsdl:part name="parameters" element="tns:GetListItemChangesSinceToken" />
</wsdl:message>
<wsdl:message name="GetListItemChangesSinceTokenSoapOut">
  <wsdl:part name="parameters" element="tns:GetListItemChangesSinceTokenResponse" />

```

```

</wsdl:message>
<wsdl:message name="UpdateListItemsSoapIn">
  <wsdl:part name="parameters" element="tns:UpdateListItems" />
</wsdl:message>
<wsdl:message name="UpdateListItemsSoapOut">
  <wsdl:part name="parameters" element="tns:UpdateListItemsResponse" />
</wsdl:message>
<wsdl:message name="UpdateListItemsWithKnowledgeSoapIn">
  <wsdl:part name="parameters" element="tns:UpdateListItemsWithKnowledge" />
</wsdl:message>
<wsdl:message name="UpdateListItemsWithKnowledgeSoapOut">
  <wsdl:part name="parameters" element="tns:UpdateListItemsWithKnowledgeResponse" />
</wsdl:message>
<wsdl:message name="AddDiscussionBoardItemSoapIn">
  <wsdl:part name="parameters" element="tns:AddDiscussionBoardItem" />
</wsdl:message>
<wsdl:message name="AddDiscussionBoardItemSoapOut">
  <wsdl:part name="parameters" element="tns:AddDiscussionBoardItemResponse" />
</wsdl:message>
<wsdl:message name="AddWikiPageSoapIn">
  <wsdl:part name="parameters" element="tns:AddWikiPage" />
</wsdl:message>
<wsdl:message name="AddWikiPageSoapOut">
  <wsdl:part name="parameters" element="tns:AddWikiPageResponse" />
</wsdl:message>
<wsdl:message name="GetVersionCollectionSoapIn">
  <wsdl:part name="parameters" element="tns:GetVersionCollection" />
</wsdl:message>
<wsdl:message name="GetVersionCollectionSoapOut">
  <wsdl:part name="parameters" element="tns:GetVersionCollectionResponse" />
</wsdl:message>
<wsdl:message name="AddAttachmentSoapIn">
  <wsdl:part name="parameters" element="tns:AddAttachment" />
</wsdl:message>
<wsdl:message name="AddAttachmentSoapOut">
  <wsdl:part name="parameters" element="tns:AddAttachmentResponse" />
</wsdl:message>
<wsdl:message name="GetAttachmentCollectionSoapIn">
  <wsdl:part name="parameters" element="tns:GetAttachmentCollection" />
</wsdl:message>
<wsdl:message name="GetAttachmentCollectionSoapOut">
  <wsdl:part name="parameters" element="tns:GetAttachmentCollectionResponse" />
</wsdl:message>
<wsdl:message name="DeleteAttachmentSoapIn">
  <wsdl:part name="parameters" element="tns>DeleteAttachment" />
</wsdl:message>
<wsdl:message name="DeleteAttachmentSoapOut">
  <wsdl:part name="parameters" element="tns>DeleteAttachmentResponse" />
</wsdl:message>
<wsdl:message name="CheckOutFileSoapIn">
  <wsdl:part name="parameters" element="tns:CheckOutFile" />
</wsdl:message>
<wsdl:message name="CheckOutFileSoapOut">
  <wsdl:part name="parameters" element="tns:CheckOutFileResponse" />
</wsdl:message>
<wsdl:message name="UndoCheckOutSoapIn">
  <wsdl:part name="parameters" element="tns:UndoCheckOut" />
</wsdl:message>
<wsdl:message name="UndoCheckOutSoapOut">

```

```

    <wsdl:part name="parameters" element="tns:UndoCheckOutResponse" />
</wsdl:message>
<wsdl:message name="CheckInFileSoapIn">
  <wsdl:part name="parameters" element="tns:CheckInFile" />
</wsdl:message>
<wsdl:message name="CheckInFileSoapOut">
  <wsdl:part name="parameters" element="tns:CheckInFileResponse" />
</wsdl:message>
<wsdl:message name="GetListContentTypesSoapIn">
  <wsdl:part name="parameters" element="tns:GetListContentTypes" />
</wsdl:message>
<wsdl:message name="GetListContentTypesSoapOut">
  <wsdl:part name="parameters" element="tns:GetListContentTypesResponse" />
</wsdl:message>
<wsdl:message name="GetListContentTypesAndPropertiesSoapIn">
  <wsdl:part name="parameters" element="tns:GetListContentTypesAndProperties" />
</wsdl:message>
<wsdl:message name="GetListContentTypesAndPropertiesSoapOut">
  <wsdl:part name="parameters" element="tns:GetListContentTypesAndPropertiesResponse" />
</wsdl:message>
<wsdl:message name="GetListContentTypeSoapIn">
  <wsdl:part name="parameters" element="tns:GetListContentType" />
</wsdl:message>
<wsdl:message name="GetListContentTypeSoapOut">
  <wsdl:part name="parameters" element="tns:GetListContentTypeResponse" />
</wsdl:message>
<wsdl:message name="CreateContentTypeSoapIn">
  <wsdl:part name="parameters" element="tns:CreateContentType" />
</wsdl:message>
<wsdl:message name="CreateContentTypeSoapOut">
  <wsdl:part name="parameters" element="tns:CreateContentTypeResponse" />
</wsdl:message>
<wsdl:message name="UpdateContentTypeSoapIn">
  <wsdl:part name="parameters" element="tns:UpdateContentType" />
</wsdl:message>
<wsdl:message name="UpdateContentTypeSoapOut">
  <wsdl:part name="parameters" element="tns:UpdateContentTypeResponse" />
</wsdl:message>
<wsdl:message name="DeleteContentTypeSoapIn">
  <wsdl:part name="parameters" element="tns>DeleteContentType" />
</wsdl:message>
<wsdl:message name="DeleteContentTypeSoapOut">
  <wsdl:part name="parameters" element="tns>DeleteContentTypeResponse" />
</wsdl:message>
<wsdl:message name="UpdateContentTypeXmlDocumentSoapIn">
  <wsdl:part name="parameters" element="tns:UpdateContentTypeXmlDocument" />
</wsdl:message>
<wsdl:message name="UpdateContentTypeXmlDocumentSoapOut">
  <wsdl:part name="parameters" element="tns:UpdateContentTypeXmlDocumentResponse" />
</wsdl:message>
<wsdl:message name="UpdateContentTypesXmlDocumentSoapIn">
  <wsdl:part name="parameters" element="tns:UpdateContentTypesXmlDocument" />
</wsdl:message>
<wsdl:message name="UpdateContentTypesXmlDocumentSoapOut">
  <wsdl:part name="parameters" element="tns:UpdateContentTypesXmlDocumentResponse" />
</wsdl:message>
<wsdl:message name="DeleteContentTypeXmlDocumentSoapIn">
  <wsdl:part name="parameters" element="tns>DeleteContentTypeXmlDocument" />
</wsdl:message>

```



```

<wsdl:message name="DeleteContentTypeXmlDocumentSoapOut">
  <wsdl:part name="parameters" element="tns:DeleteContentTypeXmlDocumentResponse" />
</wsdl:message>
<wsdl:message name="ApplyContentTypeToListSoapIn">
  <wsdl:part name="parameters" element="tns:ApplyContentTypeToList" />
</wsdl:message>
<wsdl:message name="ApplyContentTypeToListSoapOut">
  <wsdl:part name="parameters" element="tns:ApplyContentTypeToListResponse" />
</wsdl:message>
<wsdl:portType name="ListsSoap">
  <wsdl:operation name="GetList">
    <wsdl:input message="tns:GetListSoapIn" />
    <wsdl:output message="tns:GetListSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="GetListAndView">
    <wsdl:input message="tns:GetListAndViewSoapIn" />
    <wsdl:output message="tns:GetListAndViewSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="DeleteList">
    <wsdl:input message="tns>DeleteListSoapIn" />
    <wsdl:output message="tns>DeleteListSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="AddList">
    <wsdl:input message="tns:AddListSoapIn" />
    <wsdl:output message="tns:AddListSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="AddListFromFeature">
    <wsdl:input message="tns:AddListFromFeatureSoapIn" />
    <wsdl:output message="tns:AddListFromFeatureSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="UpdateList">
    <wsdl:input message="tns:UpdateListSoapIn" />
    <wsdl:output message="tns:UpdateListSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="GetListCollection">
    <wsdl:input message="tns:GetListCollectionSoapIn" />
    <wsdl:output message="tns:GetListCollectionSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="GetListItems">
    <wsdl:input message="tns:GetListItemsSoapIn" />
    <wsdl:output message="tns:GetListItemsSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="GetListItemChanges">
    <wsdl:input message="tns:GetListItemChangesSoapIn" />
    <wsdl:output message="tns:GetListItemChangesSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="GetListItemChangesWithKnowledge">
    <wsdl:input message="tns:GetListItemChangesWithKnowledgeSoapIn" />
    <wsdl:output message="tns:GetListItemChangesWithKnowledgeSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="GetListItemChangesSinceToken">
    <wsdl:input message="tns:GetListItemChangesSinceTokenSoapIn" />
    <wsdl:output message="tns:GetListItemChangesSinceTokenSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="UpdateListItems">
    <wsdl:input message="tns:UpdateListItemsSoapIn" />
    <wsdl:output message="tns:UpdateListItemsSoapOut" />
  </wsdl:operation>
  <wsdl:operation name="UpdateListItemsWithKnowledge">

```

```

    <wsdl:input message="tns:UpdateListItemsWithKnowledgeSoapIn" />
    <wsdl:output message="tns:UpdateListItemsWithKnowledgeSoapOut" />
</wsdl:operation>
<wsdl:operation name="AddDiscussionBoardItem">
    <wsdl:input message="tns:AddDiscussionBoardItemSoapIn" />
    <wsdl:output message="tns:AddDiscussionBoardItemSoapOut" />
</wsdl:operation>
<wsdl:operation name="AddWikiPage">
    <wsdl:input message="tns:AddWikiPageSoapIn" />
    <wsdl:output message="tns:AddWikiPageSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetVersionCollection">
    <wsdl:input message="tns:GetVersionCollectionSoapIn" />
    <wsdl:output message="tns:GetVersionCollectionSoapOut" />
</wsdl:operation>
<wsdl:operation name="AddAttachment">
    <wsdl:input message="tns:AddAttachmentSoapIn" />
    <wsdl:output message="tns:AddAttachmentSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetAttachmentCollection">
    <wsdl:input message="tns:GetAttachmentCollectionSoapIn" />
    <wsdl:output message="tns:GetAttachmentCollectionSoapOut" />
</wsdl:operation>
<wsdl:operation name="DeleteAttachment">
    <wsdl:input message="tns>DeleteAttachmentSoapIn" />
    <wsdl:output message="tns>DeleteAttachmentSoapOut" />
</wsdl:operation>
<wsdl:operation name="CheckOutFile">
    <wsdl:input message="tns:CheckOutFileSoapIn" />
    <wsdl:output message="tns:CheckOutFileSoapOut" />
</wsdl:operation>
<wsdl:operation name="UndoCheckOut">
    <wsdl:input message="tns:UndoCheckOutSoapIn" />
    <wsdl:output message="tns:UndoCheckOutSoapOut" />
</wsdl:operation>
<wsdl:operation name="CheckInFile">
    <wsdl:input message="tns:CheckInFileSoapIn" />
    <wsdl:output message="tns:CheckInFileSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetListContentTypes">
    <wsdl:input message="tns:GetListContentTypesSoapIn" />
    <wsdl:output message="tns:GetListContentTypesSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetListContentTypesAndProperties">
    <wsdl:input message="tns:GetListContentTypesAndPropertiesSoapIn" />
    <wsdl:output message="tns:GetListContentTypesAndPropertiesSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetListContentType">
    <wsdl:input message="tns:GetListContentTypeSoapIn" />
    <wsdl:output message="tns:GetListContentTypeSoapOut" />
</wsdl:operation>
<wsdl:operation name="CreateContentType">
    <wsdl:input message="tns:CreateContentTypeSoapIn" />
    <wsdl:output message="tns:CreateContentTypeSoapOut" />
</wsdl:operation>
<wsdl:operation name="UpdateContentType">
    <wsdl:input message="tns:UpdateContentTypeSoapIn" />
    <wsdl:output message="tns:UpdateContentTypeSoapOut" />
</wsdl:operation>

```

```

<wsdl:operation name="DeleteContentType">
  <wsdl:input message="tns:DeleteContentTypeSoapIn" />
  <wsdl:output message="tns:DeleteContentTypeSoapOut" />
</wsdl:operation>
<wsdl:operation name="UpdateContentTypeXmlDocument">
  <wsdl:input message="tns:UpdateContentTypeXmlDocumentSoapIn" />
  <wsdl:output message="tns:UpdateContentTypeXmlDocumentSoapOut" />
</wsdl:operation>
<wsdl:operation name="UpdateContentTypesXmlDocument">
  <wsdl:input message="tns:UpdateContentTypesXmlDocumentSoapIn" />
  <wsdl:output message="tns:UpdateContentTypesXmlDocumentSoapOut" />
</wsdl:operation>
<wsdl:operation name="DeleteContentTypeXmlDocument">
  <wsdl:input message="tns:DeleteContentTypeXmlDocumentSoapIn" />
  <wsdl:output message="tns:DeleteContentTypeXmlDocumentSoapOut" />
</wsdl:operation>
<wsdl:operation name="ApplyContentTypeToList">
  <wsdl:input message="tns:ApplyContentTypeToListSoapIn" />
  <wsdl:output message="tns:ApplyContentTypeToListSoapOut" />
</wsdl:operation>
</wsdl:portType>
<wsdl:binding name="ListsSoap" type="tns:ListsSoap">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="GetList">
    <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/GetList"
style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetListAndView">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListAndView" style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="DeleteList">
    <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteList"
style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="AddList">
    <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/AddList"
style="document" />
    <wsdl:input>
      <soap:body use="literal" />
    </wsdl:input>
  </wsdl:operation>

```

```

        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="AddListFromFeature">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/AddListFromFeature" style="document"
/>
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UpdateList">
        <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateList"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListCollection">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListCollection" style="document"
/>
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListItems">
        <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListItems"
style="document" />
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListItemChanges">
        <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListItemChanges" style="document"
/>
        <wsdl:input>
            <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListItemChangesWithKnowledge">

```

```

    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListItemChangesWithKnowledge"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetListItemChangesSinceToken">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListItemChangesSinceToken"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UpdateListItems">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateListItems" style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UpdateListItemsWithKnowledge">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateListItemsWithKnowledge"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="AddDiscussionBoardItem">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/AddDiscussionBoardItem"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="AddWikiPage">
    <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/AddWikiPage"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>

```

```

        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetVersionCollection">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetVersionCollection"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="AddAttachment">
    <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/AddAttachment"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetAttachmentCollection">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetAttachmentCollection"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DeleteAttachment">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteAttachment" style="document"
/>
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CheckOutFile">
    <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/CheckOutFile"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UndoCheckOut">
    <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/UndoCheckOut"
style="document" />
    <wsdl:input>

```

```

        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CheckInFile">
    <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/CheckInFile"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetListContentTypes">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListContentTypes"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetListContentTypesAndProperties">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListContentTypesAndProperties"
style="document" />
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetListContentType">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListContentType" style="document"
/>
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CreateContentType">
    <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/CreateContentType" style="document"
/>
    <wsdl:input>
        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>

```

```

    <wsdl:operation name="UpdateContentType">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateContentType" style="document"
/>
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="DeleteContentType">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteContentType" style="document"
/>
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UpdateContentTypeXmlDocument">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateContentTypeXmlDocument"
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UpdateContentTypesXmlDocument">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateContentTypesXmlDocument"
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="DeleteContentTypeXmlDocument">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteContentTypeXmlDocument"
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="ApplyContentTypeToList">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/ApplyContentTypeToList"
style="document" />
      <wsdl:input>

```



```

        <soap:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal" />
    </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:binding name="ListsSoap12" type="tns:ListsSoap">
    <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="GetList">
        <soap12:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/GetList"
style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListAndView">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListAndView" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="DeleteList">
        <soap12:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteList"
style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="AddList">
        <soap12:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/AddList"
style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="AddListFromFeature">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/AddListFromFeature" style="document"
/>
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>

```

```

    </wsdl:operation>
    <wsdl:operation name="UpdateList">
      <soap12:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateList"
style="document" />
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListCollection">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListCollection" style="document"
/>
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListItems">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListItems" style="document" />
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListItemChanges">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListItemChanges" style="document"
/>
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListItemChangesWithKnowledge">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListItemChangesWithKnowledge"
style="document" />
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListItemChangesSinceToken">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListItemChangesSinceToken"
style="document" />
      <wsdl:input>
        <soap12:body use="literal" />

```

```

        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UpdateListItems">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateListItems" style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UpdateListItemsWithKnowledge">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateListItemsWithKnowledge"
style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="AddDiscussionBoardItem">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/AddDiscussionBoardItem"
style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="AddWikiPage">
        <soap12:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/AddWikiPage"
style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetVersionCollection">
        <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetVersionCollection"
style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="AddAttachment">

```

```

    <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/AddAttachment" style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetAttachmentCollection">
    <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetAttachmentCollection"
style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="DeleteAttachment">
    <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteAttachment" style="document"
/>
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CheckOutFile">
    <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/CheckOutFile" style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="UndoCheckOut">
    <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/UndoCheckOut" style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CheckInFile">
    <soap12:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/CheckInFile"
style="document" />
    <wsdl:input>
        <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal" />
    </wsdl:output>

```

```

    </wsdl:operation>
    <wsdl:operation name="GetListContentTypes">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListContentTypes"
style="document" />
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListContentTypesAndProperties">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListContentTypesAndProperties"
style="document" />
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetListContentType">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetListContentType" style="document"
/>
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="CreateContentType">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/CreateContentType" style="document"
/>
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UpdateContentType">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateContentType" style="document"
/>
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="DeleteContentType">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteContentType" style="document"
/>

```

```

        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UpdateContentTypeXmlDocument">
        <soap12:operation
            soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateContentTypeXmlDocument"
            style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UpdateContentTypesXmlDocument">
        <soap12:operation
            soapAction="http://schemas.microsoft.com/sharepoint/soap/UpdateContentTypesXmlDocument"
            style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="DeleteContentTypeXmlDocument">
        <soap12:operation
            soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteContentTypeXmlDocument"
            style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="ApplyContentTypeToList">
        <soap12:operation
            soapAction="http://schemas.microsoft.com/sharepoint/soap/ApplyContentTypeToList"
            style="document" />
        <wsdl:input>
            <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal" />
        </wsdl:output>
    </wsdl:operation>
</wsdl:binding>
<wsdl:service name="Lists">
    <wsdl:port name="ListsSoap" binding="tns:ListsSoap">
        <soap:address location="http://my/sites/aditima/_vti_bin/lists.asmx" />
    </wsdl:port>
    <wsdl:port name="ListsSoap12" binding="tns:ListsSoap12">
        <soap12:address location="http://my/sites/aditima/_vti_bin/lists.asmx" />
    </wsdl:port>
</wsdl:service>

```

</wsdl:definitions>

Preliminary

7 Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft® Office 2003
- The 2007 Microsoft® Office system
- Microsoft® Office 2010 suites
- Microsoft® Office 15 Technical Preview
- Windows® SharePoint® Services 2.0
- Windows® SharePoint® Services 3.0
- Microsoft® SharePoint® Foundation 2010
- Microsoft® SharePoint® Foundation 15 Technical Preview

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product 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 2.2.4.11: **EventSinkAssembly**, **EventSinkClass**, and **EventSinkData**](#) attributes of the **List** element were deprecated in Windows SharePoint Services 3.0 protocol servers.

[<2> Section 2.2.4.11: **EventSinkAssembly**, **EventSinkClass**, and **EventSinkData**](#) attributes of the **List** element were deprecated in Windows SharePoint Services 3.0 protocol servers.

[<3> Section 2.2.4.11: **EventSinkAssembly**, **EventSinkClass**, and **EventSinkData**](#) attributes of the **List** element were deprecated in Windows SharePoint Services 3.0 protocol servers.

[<4> Section 2.2.4.11:](#) This attribute is not returned by Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 servers.

[<5> Section 2.2.4.11:](#) This attribute is not returned by Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 servers.

[<6> Section 2.2.4.11:](#) This attribute is not returned by Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 servers.

[<7> Section 2.2.4.11:](#) This attribute is not returned by Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 servers.

[<8> Section 2.2.4.11:](#) This attribute is not returned by Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 servers.

[<9> Section 2.2.4.11:](#) This attribute is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<10> [Section 2.2.4.11](#): This attribute is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<11> [Section 2.2.4.11](#): This attribute is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<12> [Section 2.2.4.11](#): This attribute is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<13> [Section 2.2.4.11](#): This attribute is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<14> [Section 2.2.4.11](#): This attribute is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<15> [Section 2.2.4.12](#): This attribute is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<16> [Section 2.2.4.12](#): This attribute is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<17> [Section 2.2.4.12](#): This attribute is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<18> [Section 2.2.4.13](#): This attribute is not returned in Windows SharePoint Services 3.0.

<19> [Section 2.2.4.13](#): This attribute is not returned in Windows SharePoint Services 3.0.

<20> [Section 2.2.4.13](#): This attribute is not returned in Windows SharePoint Services 3.0.

<21> [Section 3.1.4.1](#): Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<22> [Section 3.1.4.2](#): Windows SharePoint Services 2.0 does not support this operation.

<23> [Section 3.1.4.2](#): Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<24> [Section 3.1.4.4](#): Windows SharePoint Services 2.0 does not support this operation.

<25> [Section 3.1.4.6](#): Windows SharePoint Services 2.0 does not support this operation.

<26> [Section 3.1.4.6](#): Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<27> [Section 3.1.4.7](#): Windows SharePoint Services 2.0 does not support this operation.

<28> [Section 3.1.4.8](#): Windows SharePoint Services 2.0 does not support this operation.

<29> [Section 3.1.4.9](#): Windows SharePoint Services 2.0 does not support this operation.

<30> [Section 3.1.4.9](#): Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<31> [Section 3.1.4.10:](#) Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx)".

<32> [Section 3.1.4.11:](#) Windows SharePoint Services 2.0 does not support this operation.

<33> [Section 3.1.4.11:](#) Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx)".

<34> [Section 3.1.4.12:](#) Windows SharePoint Services 2.0 does not support this operation.

<35> [Section 3.1.4.12:](#) Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx)".

<36> [Section 3.1.4.13:](#) Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx)".

<37> [Section 3.1.4.14:](#) Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx)".

<38> [Section 3.1.4.15:](#) Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx)".

<39> [Section 3.1.4.16:](#) Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx)".

<40> [Section 3.1.4.18:](#) Windows SharePoint Services 2.0 does not support this operation.

<41> [Section 3.1.4.18:](#) Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx)".

<42> [Section 3.1.4.19:](#) Windows SharePoint Services 2.0 does not support this operation.

<43> [Section 3.1.4.19.2.1:](#) Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx)".

<44> [Section 3.1.4.19.2.2:](#) The attribute is not returned in Windows SharePoint Services 3.0.

<45> [Section 3.1.4.20:](#) Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 do not support the **GetListContentTypesAndProperties** operation.

<46> [Section 3.1.4.20.2.2:](#) The attribute is not returned in Windows SharePoint Services 3.0.

<47> [Section 3.1.4.21:](#) Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxx)".

<48> [Section 3.1.4.22:](#) Windows SharePoint Services 2.0 does not support this operation.

<49> [Section 3.1.4.22](#): In Windows SharePoint Services 3.0, if the value of the **changeToken** parameter is empty, the protocol server will return a SOAP fault. There is no error code for this fault.

<50> [Section 3.1.4.22.2.1](#): In SharePoint Foundation 2010, if the **changeToken** parameter is specified and the total number of list items that have been inserted or updated is greater than 40, then only the first 40 items that changed are returned.

<51> [Section 3.1.4.22.2.2](#): The **FileFragment** element is not returned in Windows SharePoint Services 3.0.

<52> [Section 3.1.4.23](#): Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 do not support this operation.

<53> [Section 3.1.4.24](#): Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<54> [Section 3.1.4.25](#): Windows SharePoint Services 2.0 does not support this operation.

<55> [Section 3.1.4.25](#): Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<56> [Section 3.1.4.26](#): Windows SharePoint Services 2.0 does not support this operation.

<57> [Section 3.1.4.27](#): Windows SharePoint Services 2.0 does not support this operation.

<58> [Section 3.1.4.27](#): Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<59> [Section 3.1.4.28](#): Windows SharePoint Services 2.0 does not support this operation.

<60> [Section 3.1.4.28](#): Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<61> [Section 3.1.4.29](#): Windows SharePoint Services 2.0 does not support this operation.

<62> [Section 3.1.4.29](#): Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<63> [Section 3.1.4.30](#): Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<64> [Section 3.1.4.30.2.1](#): This attribute is not returned by the server in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<65> [Section 3.1.4.30.2.1](#): This attribute is not returned by the server in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<66> [Section 3.1.4.31](#): Windows SharePoint Services 3.0 and Windows SharePoint Services 2.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

[<67> Section 3.1.4.31.2.1:](#) Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 do not support this behavior.

[<68> Section 3.1.4.32:](#) Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 do not support this method.

Preliminary

8 Change Tracking

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

Preliminary

9 Index

A

Abstract data model
 [server](#) 44
[AddOrUpdateFieldDefinition complex type](#) 19
[AddOrUpdateFieldsDefinition complex type](#) 20
[Applicability](#) 16
[Attribute groups](#) 43
[Attributes](#) 42

C

[CamlContains complex type](#) 21
[CamlQueryOptions complex type](#) 22
[CamlViewFields complex type](#) 26
[Capability negotiation](#) 17
[Change tracking](#) 205
[ChangeTypeEnum simple type](#) 39
Client
 [overview](#) 44
Common data structures ([section 2.2.9](#) 43, [section 2.2.9](#) 43)
[Complex types](#) 19
 [AddOrUpdateFieldDefinition](#) 19
 [AddOrUpdateFieldsDefinition](#) 20
 [CamlContains](#) 21
 [CamlQueryOptions](#) 22
 [CamlViewFields](#) 26
 [ContentTypePropertyDefinition](#) 26
 [DataDefinition](#) 27
 [DeleteFieldDefinition](#) 27
 [DeleteFieldsDefinition](#) 27
 [FieldReferenceDefinitionCT](#) 28
 [ListDefinitionCT](#) 29
 [ListDefinitionSchema](#) 34
 [ListItemChangeDefinition](#) 36
 [UpdateListFieldResults](#) 37
 [UpdateListFieldsRequest](#) 37
 [ViewChangeDefinition](#) 38
[Content type management](#) 13
[ContentTypePropertyDefinition complex type](#) 26
[Create and update a list example](#) 160

D

Data model - abstract
 [server](#) 44
[DataDefinition complex type](#) 27
[DeleteFieldDefinition complex type](#) 27
[DeleteFieldsDefinition complex type](#) 27

E

[EnumOptimizeFor simple type](#) 40
[EnumRecurrencePatternXMLVersion simple type](#) 40
[EnumViewAttributes simple type](#) 41
Events
 [local - server](#) 149
 [timer - server](#) 149

Examples

[create and update a list example](#) 160
[new content type](#) 150

F

[FieldReferenceDefinitionCT complex type](#) 28
[Fields - vendor-extensible](#) 17
[Full WSDL](#) 167

G

[Glossary](#) 9
[Groups](#) 42

I

[Implementer - security considerations](#) 166
[Index of security parameters](#) 166
[Informative references](#) 12
Initialization
 [server](#) 44
[Introduction](#) 9

L

[List data management](#) 14
[List schema management](#) 13
[ListDefinitionCT complex type](#) 29
[ListDefinitionSchema complex type](#) 34
[ListItemChangeDefinition complex type](#) 36
Local events
 [server](#) 149

M

Message processing
 server ([section 3.1.4](#) 44, [section 3.1.4](#) 44)
Messages
 [AddOrUpdateFieldDefinition complex type](#) 19
 [AddOrUpdateFieldsDefinition complex type](#) 20
 [attribute groups](#) 43
 [attributes](#) 42
 [CamlContains complex type](#) 21
 [CamlQueryOptions complex type](#) 22
 [CamlViewFields complex type](#) 26
 [ChangeTypeEnum simple type](#) 39
 common data structures ([section 2.2.9](#) 43, [section 2.2.9](#) 43)
 [complex types](#) 19
 [ContentTypePropertyDefinition complex type](#) 26
 [DataDefinition complex type](#) 27
 [DeleteFieldDefinition complex type](#) 27
 [DeleteFieldsDefinition complex type](#) 27
 [elements](#) 19
 [enumerated](#) 19
 [EnumOptimizeFor simple type](#) 40
 [EnumRecurrencePatternXMLVersion simple type](#) 40

- [EnumViewAttributes simple type](#) 41
- [FieldReferenceDefinitionCT complex type](#) 28
- [groups](#) 42
- [ListDefinitionCT complex type](#) 29
- [ListDefinitionSchema complex type](#) 34
- [ListItemChangeDefinition complex type](#) 36
- [MethodCmdEnum simple type](#) 41
- [namespaces](#) 18
- [OnErrorEnum simple type](#) 42
- [simple types](#) 38
- [syntax](#) 18
- [transport](#) 18
- [TRUEONLY simple type](#) 42
- [UpdateListFieldResults complex type](#) 37
- [UpdateListFieldsRequest complex type](#) 37
- [ViewChangeDefinition complex type](#) 38
- [MethodCmdEnum simple type](#) 41

N

- [Namespaces](#) 18
- [New content type example](#) 150
- [Normative references](#) 11

O

- [OnErrorEnum simple type](#) 42
- Operations
 - [AddAttachment](#) 46
 - [AddDiscussionBoardItem](#) 48
 - [AddList](#) 51
 - [AddListFromFeature](#) 53
 - [AddWikiPage](#) 56
 - [ApplyContentTypeToList](#) 58
 - [CheckInFile](#) 60
 - [CheckOutFile](#) 62
 - [CreateContentType](#) 65
 - [DeleteAttachment](#) 67
 - [DeleteContentType](#) 70
 - [DeleteContentTypeXmlDocument](#) 72
 - [DeleteList](#) 74
 - [GetAttachmentCollection](#) 76
 - [GetList](#) 79
 - [GetListAndView](#) 81
 - [GetListCollection](#) 83
 - [GetListContentType](#) 85
 - [GetListContentTypes](#) 87
 - [GetListContentTypesAndProperties](#) 90
 - [GetListItemChanges](#) 95
 - [GetListItemChangesSinceToken](#) 100
 - [GetListItemChangesWithKnowledge](#) 106
 - [GetListItems](#) 112
 - [GetVersionCollection](#) 116
 - [UndoCheckOut](#) 119
 - [UpdateContentType](#) 121
 - [UpdateContentTypesXmlDocument](#) 126
 - [UpdateContentTypeXmlDocument](#) 129
 - [UpdateList](#) 132
 - [UpdateListItems](#) 136
 - [UpdateListItemsWithKnowledge](#) 145
- [Overview \(synopsis\)](#) 12

P

- [Parameters - security index](#) 166
- [Preconditions](#) 16
- [Prerequisites](#) 16
- [Product behavior](#) 200
- [Protocol clients](#) 13
- [Protocol server](#) 12

R

- [References](#) 11
 - [informative](#) 12
 - [normative](#) 11
- [Relationship to other protocols](#) 16
- [Roles](#) 12

S

- Scenarios
 - [content type management](#) 13
 - [list data management](#) 14
 - [list schema management](#) 13
 - [overview](#) 13
- Security
 - [implementer considerations](#) 166
 - [parameter index](#) 166
- Sequencing rules
 - server ([section 3.1.4](#) 44, [section 3.1.4](#) 44)
- Server
 - [abstract data model](#) 44
 - [AddAttachment operation](#) 46
 - [AddDiscussionBoardItem operation](#) 48
 - [AddList operation](#) 51
 - [AddListFromFeature operation](#) 53
 - [AddWikiPage operation](#) 56
 - [ApplyContentTypeToList operation](#) 58
 - [CheckInFile operation](#) 60
 - [CheckOutFile operation](#) 62
 - [CreateContentType operation](#) 65
 - [DeleteAttachment operation](#) 67
 - [DeleteContentType operation](#) 70
 - [DeleteContentTypeXmlDocument operation](#) 72
 - [DeleteList operation](#) 74
 - [GetAttachmentCollection operation](#) 76
 - [GetList operation](#) 79
 - [GetListAndView operation](#) 81
 - [GetListCollection operation](#) 83
 - [GetListContentType operation](#) 85
 - [GetListContentTypes operation](#) 87
 - [GetListContentTypesAndProperties operation](#) 90
 - [GetListItemChanges operation](#) 95
 - [GetListItemChangesSinceToken operation](#) 100
 - [GetListItemChangesWithKnowledge operation](#) 106
 - [GetListItems operation](#) 112
 - [GetVersionCollection operation](#) 116
 - [initialization](#) 44
 - [local events](#) 149
 - message processing ([section 3.1.4](#) 44, [section 3.1.4](#) 44)

[overview](#) 44
sequencing rules ([section 3.1.4](#) 44, [section 3.1.4](#) 44)
[timer events](#) 149
[timers](#) 44
[UndoCheckOut operation](#) 119
[UpdateContentType operation](#) 121
[UpdateContentTypesXmlDocument operation](#) 126
[UpdateContentTypeXmlDocument operation](#) 129
[UpdateList operation](#) 132
[UpdateListItems operation](#) 136
[UpdateListItemsWithKnowledge operation](#) 145
Simple types 38
[ChangeTypeEnum](#) 39
[EnumOptimizeFor](#) 40
[EnumRecurrencePatternXMLVersion](#) 40
[EnumViewAttributes](#) 41
[MethodCmdEnum](#) 41
[OnErrorEnum](#) 42
[TRUEONLY](#) 42
Standards assignments 17
Syntax
[messages - overview](#) 18

T

Timer events
[server](#) 149
Timers
[server](#) 44
[Tracking changes](#) 205
[Transport](#) 18
[TRUEONLY simple type](#) 42
Types
[complex](#) 19
[simple](#) 38

U

[UpdateListFieldResults complex type](#) 37
[UpdateListFieldsRequest complex type](#) 37

V

[Vendor-extensible fields](#) 17
[Versioning](#) 17
[ViewChangeDefinition complex type](#) 38

W

[WSDL](#) 167