[MS-LISTSWS]:

Lists Web Service Protocol

Intellectual Property Rights Notice for Open Specifications Documentation

Revision Summary

Date	Revision History	Revision Class	Comments
4/4/2008	0.1	New	Initial Availability
6/27/2008	1.0	Major	Revised and edited the technical content
12/12/2008	1.01	Editorial	Revised and edited the technical content
7/13/2009	1.02	Major	Revised and edited the technical content
8/28/2009	1.03	Editorial	Revised and edited the technical content
11/6/2009	1.04	Editorial	Revised and edited the technical content
2/19/2010	2.0	Major	Updated and revised the technical content
3/31/2010	2.01	Editorial	Revised and edited the technical content
4/30/2010	2.02	Editorial	Revised and edited the technical content
6/7/2010	2.03	Editorial	Revised and edited the technical content
6/29/2010	2.04	Major	Significantly changed the technical content.
7/23/2010	2.04	None	No changes to the meaning, language, or formatting of the technical content.
9/27/2010	2.04	None	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	None	No changes to the meaning, language, or formatting of the technical content.
3/18/2011	2.05	None	No changes to the meaning, language, or formatting of the technical content.
6/10/2011	2.05	None	No changes to the meaning, language, or formatting of the technical content.
1/20/2012	3.0	Major	Significantly changed the technical content.
4/11/2012	3.0	None	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	4.0	Major	Significantly changed the technical content.
9/12/2012	4.0	None	No changes to the meaning, language, or formatting of the technical content.
10/8/2012	4.1	Minor	Clarified the meaning of the technical content.
2/11/2013	5.0	Major	Significantly changed the technical content.
7/30/2013	5.1	Minor	Clarified the meaning of the technical content.
11/18/2013	5.2	Minor	Clarified the meaning of the technical content.
2/10/2014	5.2	None	No changes to the meaning, language, or formatting of the technical content.

Date	Revision History	Revision Class	Comments
4/30/2014	5.3	Minor	Clarified the meaning of the technical content.
7/31/2014	5.4	Minor	Clarified the meaning of the technical content.
10/30/2014	5.5	Minor	Clarified the meaning of the technical content.
3/16/2015	6.0	Major	Significantly changed the technical content.
6/30/2015	7.0	Major	Significantly changed the technical content.
2/26/2016	8.0	Major	Significantly changed the technical content.
4/14/2016	9.0	Major	Significantly changed the technical content.
7/15/2016	9.0	None	No changes to the meaning, language, or formatting of the technical content.
9/14/2016	9.0	None	No changes to the meaning, language, or formatting of the technical content.
12/15/2016	9.1	Minor	Clarified the meaning of the technical content.
6/20/2017	10.0	Major	Significantly changed the technical content.
7/24/2018	11.0	Major	Significantly changed the technical content.
10/1/2018	12.0	Major	Significantly changed the technical content.

Table of Contents

1   Introduction

The List Web Service Protocol is used for the manipulation of list schemas and list data.

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

1.1   Glossary

This document uses the following terms:

access control list (ACL): A list of access control entries (ACEs) that collectively describe the security rules for authorizing access to some resource; for example, an object or set of objects.

ActiveX object: An object that is supplied by a component that supports automation.

anonymous user: A user who presents no credentials when identifying himself or herself. The process for determining an anonymous user can differ based on the authentication protocol, and the documentation for the relevant authentication protocol should be consulted.

assembly name: The name of a collection of one or more files that is versioned and deployed as a unit. See also assembly.

attachment: An external file that is included with an Internet message or associated with an item in a SharePoint list.

base type: An XML-based schema that defines the data and rendering fields that can be used in a **list**. Every list is derived from a specific base type.

check out: The process of retrieving a writable copy of a file or project from a source repository. This locks the file for editing to prevent other users from overwriting or editing it inadvertently.

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

content type group: A named category of **content types** that is used to organize content types of a similar purpose.

content type identifier: A unique identifier that is assigned to a **content type**.

context site: A site that corresponds to the context of the current request.

cookie: A small data file that is stored on a user's computer and carries state information between participating protocol servers and protocol clients.

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

data type: A property of a field that defines the kind of data that is stored in the field, or defines the kind of data returned by an expression when the expression is evaluated.

data validation: The process of testing the accuracy of data; a set of rules that specify the type and range of data that users can enter.

default list view: The view of a SharePoint list that the owner of the list selected to appear when users browse to the list without specifying a view.

default mobile list view: The view of a SharePoint list that the owner of the list selected to appear when users browse to the list by using a mobile device and without specifying a view.

default view: The layout and organization of a document or list that appears automatically when users open that document or display that list.

discussion board: A list in which users can read, post, and reply to messages from other users who are members of the same discussion board.

discussion item: A remark or response that is posted to an online discussion forum such as a newsgroup, SharePoint list, or electronic bulletin board.

display name: A text string that is used to identify a principal or other object in the user interface. Also referred to as title.

document: An object in a content database such as a file, folder, **list**, or **site (2)**. Each object is identified by a **URI**.

document library: A type of list that is a container for documents and folders.

document template: A file that serves as the basis for new documents.

email address: A string that identifies a user and enables the user to receive Internet messages.

empty GUID: A 128-bit, 16-byte identification number that is represented by all zeros.

event receiver: A structured modular component that enables built-in or user-defined managed code classes to act upon objects, such as list items, **lists**, or content types, when specific triggering actions occur.

external data: Data that is stored in a repository outside a workbook.

feature: A package of SharePoint elements that can be activated or deactivated for a specific feature scope.

feature identifier: A **GUID** that identifies a **feature**.

field: (1) An element or attribute in a data source that can contain data.

(2) A container for metadata within a SharePoint list and associated list items.

folder: A file system construct. File systems organize a volume's data by providing a hierarchy of objects, which are referred to as folders or directories, that contain files and can also contain other folders.

global assembly cache (GAC): A computer-wide code cache that stores .NET Framework assemblies that were installed specifically to be shared by multiple applications on a computer. Applications that are installed in the global assembly cache have a strong name, consisting of an assembly identity, strengthened by a public key and a digital signature.

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

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

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

Information Rights Management (IRM): A technology that provides persistent protection to digital data by using encryption, certificates, and authentication. Authorized recipients or users acquire a license to gain access to the protected files according to the rights or business rules that are set by the content owner.

item: A unit of content that can be indexed and searched by a search application.

language code identifier (LCID): A 32-bit number that identifies the user interface human language dialect or variation that is supported by an application or a client computer.

list: A container within a SharePoint site that stores list items. A list has a customizable schema that is composed of one or more fields.

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

list schema: The Collaborative Application Markup Language (CAML) schema of a list.

list template identifier: A GUID that is used to identify a list template for a SharePoint list.

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

major version: An iteration of a software component, document, or list item that is ready for a larger group to see, or has changed significantly from the previous major version. For an item on a SharePoint site, the **minor version** is always "0" (zero) for a major version.

meeting instance: A collection of data for a meeting that occurs only once or a single occurrence of a meeting that occurs multiple times. The data can be stored in a client application or on a website.

minor version: An iteration of a software component, document, or list item that is in progress or has changed only slightly from the previous version. For an item on a SharePoint site, the minor version number is never "0" (zero) and is incremented for each new version of an item, unless a **major version** is explicitly published. When minor versioning is disabled on a SharePoint site, only major version numbers are incremented, and the minor version is always "0" (zero).

offline: The condition of not being connected to or not being on a network or the Internet. Offline can also refer to a device, such as a printer that is not connected to a computer, and files that are stored on a computer that is not connected to or not on a network or the Internet.

permission: A rule that is associated with an object and that regulates which users can gain access to the object and in what manner. See also rights.

presence: A status indicator on a client device that is transmitted by using the Wide Area Network Device Presence Protocol (WAN DPP).

property bag: A container that stores data but is not defined in the schema for a SharePoint list. Instead of interpreting data in a property bag, the server only passes the data in response to requests. See also metadict.

query: A formalized instruction to a data source to either extract data or perform a specified action. A query can be in the form of a query expression, a method-based query, or a combination of the two. The data source can be in different forms, such as a relational database, **XML document**, or in-memory object. See also search query.

Recycle Bin: The location where deleted files are stored until they are either restored, if they were deleted erroneously, or destroyed permanently.

root folder: The folder at the top of a hierarchy of folders in a list.

security scope: A tree structure of objects in which every object has the same security settings as the root.

server-relative URL: A relative URL that does not specify a scheme or host, and assumes a base URI of the root of the host, as described in [RFC3986].

site: (1) A group of related webpages that is hosted by a server on the World Wide Web or an intranet. Each website has its own entry points, metadata, administration settings, and workflows. Also referred to as web site.

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

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

site content type: A named and uniquely identifiable collection of settings and fields that store metadata for lists within individual sites.

site template: An XML-based definition of site settings, including formatting, lists, views, and elements such as text, graphics, page layout, and styles. Site templates are stored in .stp files in the content database.

SOAP: A lightweight protocol for exchanging structured information in a decentralized, distributed environment. **SOAP** uses XML technologies to define an extensible messaging framework, which provides a message construct that can be exchanged over a variety of underlying protocols. The framework has been designed to be independent of any particular programming model and other implementation-specific semantics. SOAP 1.2 supersedes SOAP 1.1. See [SOAP1.2-1/2003].

SOAP action: The HTTP request header field used to indicate the intent of the **SOAP** request, using a **URI** value. See [SOAP1.1] section 6.1.1 for more information.

SOAP body: A container for the payload data being delivered by a SOAP message to its recipient. See [SOAP1.2-1/2007] section 5.3 for more information.

SOAP fault: A container for error and status information within a SOAP message. See [SOAP1.2-1/2007] section 5.4 for more information.

subsite: A complete website that is stored in a named subdirectory of another website. The parent website can be the top-level site of a site collection or another subsite. Also referred to as subweb.

thumbnail: A miniature version of an image that is typically used to browse multiple images quickly.

time zone: A geographical area that observes the same local time. The local time has a positive, zero, or negative offset from **Coordinated Universal Time (UTC)**. The offset can be different during standard time and daylight saving time.

top-level site: The first site in a site collection. All other sites within a site collection are child sites of the top-level site. The URL of the top-level site is also the URL of the site collection.

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

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

user identifier: An integer that uniquely identifies a security principal as distinct from all other security principals and site groups within the same site collection.

user information list: A list that contains items, each of which represents a security principal in a site collection. Each site collection has only one such list and it resides in the top-level site of the site collection.

version: See displayed version, historical version, **major version**, and **minor version**.

view: See form view (Microsoft InfoPath), list view (SharePoint Products and Technologies), or **View** (Microsoft Business Connectivity Services).

View: A set of Fields of an Entity, or the schema of an Entity. An Entity can have several Views, each consisting of a different set of Fields, but all containing Identifier Fields. Views are associated with MethodInstances and define the shape of a MethodInstance's input or output.

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

workflow: An automation of business processes that passes business documents and tasks automatically from one user to another for action, according to a defined sequence.

XML document: A document object that is well formed, as described in [XML10/5], and might be valid. An XML document has a logical structure that is composed of declarations, elements, comments, character references, and processing instructions. It also has a physical structure that is composed of entities, starting with the root, or document, entity.

XML namespace: A collection of names that is used to identify elements, types, and attributes in XML documents identified in a URI reference [RFC3986]. A combination of XML namespace and local name allows XML documents to use elements, types, and attributes that have the same names but come from different sources. For more information, see [XMLNS-2ED].

XML namespace prefix: An abbreviated form of an **XML namespace**, as described in [XML].

XML Path Language (XPath): A language used to create expressions that can address parts of an XML document, manipulate strings, numbers, and Booleans, and can match a set of nodes in the document, as specified in [XPATH]. XPath models an XML document as a tree of nodes of different types, including element, attribute, and text. XPath expressions can identify the nodes in an XML document based on their type, name, and values, as well as the relationship of a node to other nodes in the document.

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

1.2 References

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

1.2.1 Normative References

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

[ISO-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=1 40&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".

[MS-PRSTFR] Microsoft Corporation, "ADO XML Persistence Format".

[MS-VIEWSS] Microsoft Corporation, "Views Web Service Protocol".

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

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

[MS-WSSFO3] Microsoft Corporation, "Windows SharePoint Services (WSS): File Operations Database Communications Version 3 Protocol".

[MS-WSSTS] Microsoft Corporation, "Windows SharePoint Services".

[MS-XSSK] Microsoft Corporation, "XML Serialization of Synchronization Knowledge".

[RFC1123] Braden, R., "Requirements for Internet Hosts - Application and Support", 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://www.rfc-editor.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.rfc-editor.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.rfc-editor.org/rfc/rfc2616.txt

[RFC2822] Resnick, P., Ed., "Internet Message Format", 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.rfc-editor.org/rfc/rfc4648.txt

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", W3C Note, 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., Beech, D., Maloney, M., and Mendelsohn, N., Eds., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/

1.2.2 Informative References

None.

1.3 Overview

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.
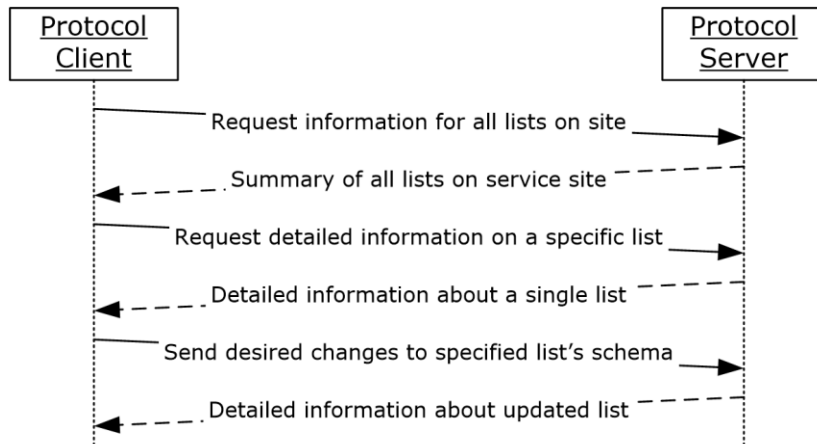
Figure 1: Path of information about list schemas

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.
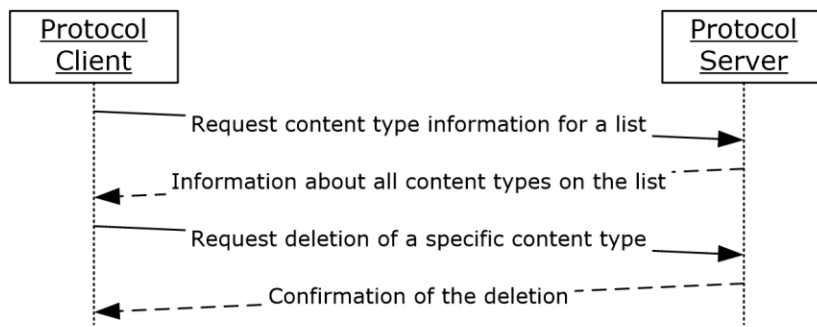
Figure 2: Path of information about content type metadata

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.
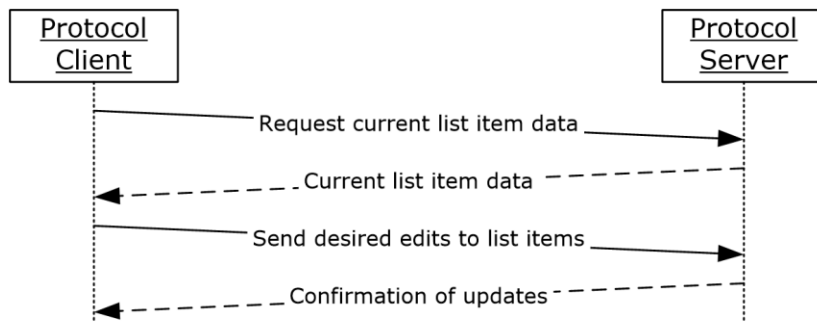
Figure 3: Path of information about list items

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 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.
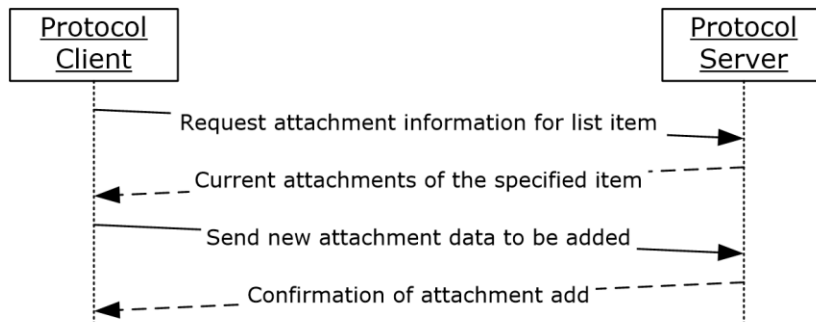
Figure 4: Path of information about list item attachments

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 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 checks in the file.

4. The protocol server responds with confirmation that the file was checked in successfully.

The following diagram shows this exchange.
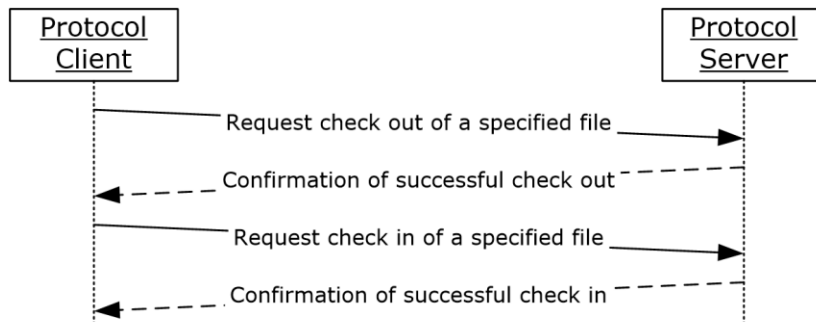
Figure 5: Path of information about documents

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:
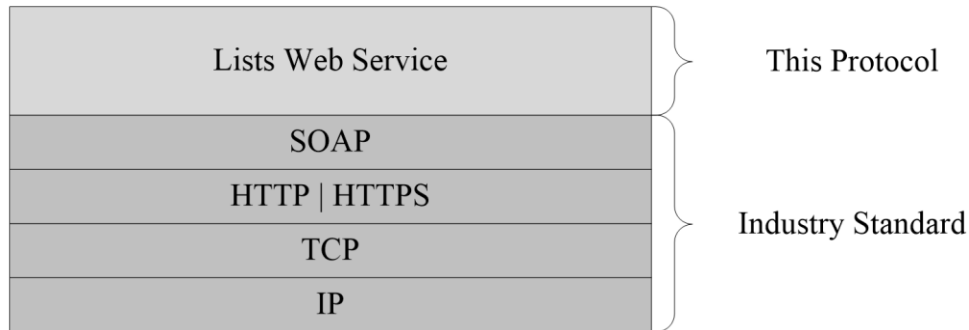
Figure 6: This protocol in relation to other protocols

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 [XMLSCHEMA1/2] and [XMLSCHEMA2/2], 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 the **GetListItems** (section 3.1.4.24)**, GetListItemChangesSinceToken** (section 3.1.4.22), and **GetListItemChangesWithKnowledge** (section 3.1.4.23) operations.
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

The **AddOrUpdateFieldDefinition** complex type 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 **XML Path Language (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

The **AddOrUpdateFieldsDefinition** complex type 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

The **CamlContains** complex type 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 (section 3.1.4.15).

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

The **CamlQueryOptions** complex type specifies parameters to alter the data and the format of the data returned from the following operations: **GetListItems** (section 3.1.4.24)**, GetListItemChangesSinceToken** (section 3.1.4.22) and **GetListItemChangesWithKnowledge** (section 3.1.4.23).

```
<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="IncludeAttachmentVersion"
                    type="core:TRUEFALSE" minOccurs="0" />
          <s:element name="RemoveInvalidXmlCharacters"
                    type="core:TRUEFALSE" minOccurs="0" />
          <s:element name="OptimizeFor" type="tns:EnumOptimizeFor" minOccurs="0" />
          <s:element name="ExtraIds" type="s:string" minOccurs="0" />
          <s:element name="OptimizeLookups" 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	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**. An example of setting this option is as follows: `<DatesInUtc>True</DatesInUtc>` 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.
Folder	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. An example of setting this option is as follows: `<Folder>Shared Documents/foldername</Folder>` This example retrieves only **list items** from the *foldername* folder.
Paging	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. An example snippet from the response to a **GetListItems** call is as follows. `<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&p_FSObjType=1&p_ID=11">…</GetListItemsResponse>`  The **ListItemCollectionPositionNext** value is extracted from this response and passed back to the protocol server to request the next page by including the query options with the following Paging element specified:  `<Paging ListItemCollectionPositionNext="Paged=TRUE&p_FSO`

Element name	Description
	`bjType=1&p_ID=11" />`
IncludeMandatoryColumns	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**. It is important to note that there MAY be additional fields that are returned by the protocol server independent of this option.
MeetingInstanceID	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(or Zero) 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. **AllButSeries** Indicates all meeting occurrences that have been activated in the workspace, excluding the meeting entry representing the meeting series. **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. **Series** Indicates the meeting entry corresponding to the meeting series, provided a series is linked to the workspace. **UnSpecified** Indicates the meeting instance is unspecified. The first upcoming meeting occurrence is used in this case.
ViewAttributes	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.
RecurrencePatternXMLVersion	Specifies the version supported for recurrence patterns. The default is an empty string or NULL.
RecurrenceOrderBy	If set to **True**, this element specifies that for each recurring series, the master item is returned first, and 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**.
IncludePermissions	If it is **True**, the **EffectivePermMask** attribute 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**.
ExpandUserField	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 "**,,**". The default for this element is **False**.

Element name	Description

	Passing `<ExpandUserField>False</ExpandUserField>` in the **query** options causes the "Author" field to be returned in the following format:
	
	`ows_Author="1;#Admin AdminName"`
	
	Passing `<ExpandUserField>True</ExpandUserField>` returns the preceding result in the following different format:
	
	`ows_Author="1;#Admin AdminName,#login\name,#email@address,#sip@address,#Admin AdminName"`

IncludeAttachmentUrls	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**.
	
	If this option is **True**, all list items returned that have no **attachments** show the **Attachments** field as follows, where … specifies that attributes have been omitted:
	
	`<z:row ows_Attachments='0' … />`
	
	If there are attachments on this list item, the **Attachment** field is returned as follows, where … specifies that attributes have been omitted:
	
	`<z:row ows_Attachments=';#http://siteservice/Lists/TestList Item/Attachments/249/Attachment1.txt;#http://siteservice/Lists/TestListItem/Attachments/249/Attachment2.txt;#' … />`

IncludeAttachmentVersion	This MUST be used only in conjunction with **IncludeAttachmentUrls**. Specifying **True** causes the **GUID** and version number to be returned for attachments that can be used for conflict detection on update. The default value is **False**. 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: `<z:row ows_Attachments=';#http://siteservice/Lists/TestList Item/Attachments/249/Attachment1.txt;#{72c714ab-32b2-46e0-8a49-8c9d2ee6ee8d},1;#http://siteservice/Lists/TestListItem/Attachments/249/Attachment2.txt;#{f196ee67-b9e8-475b-9308-a6e941052321},1;#' … />`
RemoveInvalidXmlCharacters	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.
OptimizeFor	Specifies how the internal query SHOULD be optimized: list items

Element name	Description
	or list folders. The default is to optimize for list items.
ExtraIds	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 identifiers of the folders being synchronized. This MUST only be used with a change token.
OptimizeLookups	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, if 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**.

2.2.4.5 CamlViewFields

The **CamlViewFields** complex type 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 (section 3.1.4.15), 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, the metadata properties related to the list item are returned in the protocol server response with the prefix "ows_MetaInfo_".

2.2.4.6 ContentTypePropertyDefinition

The **ContentTypePropertyDefinition** complex type 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 **document template** content type.

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

The **DataDefinition** complex type specifies items contained within a list.

```
<s:complexType name="DataDefinition" mixed="true">
  <s:sequence>
    <s:any minOccurs="0" maxOccurs="unbounded"/>
  </s:sequence>
  <s:attribute name="ItemCount" type="s:string" use="required" />
  <s:attribute name="ListItemCollectionPositionNext" type ="s:string" use="optional"/>
</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. The **ListItemCollectionPositionNext** attribute is used by protocol server methods that support paging of results. 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

The **DeleteFieldDefinition** complex type 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

The **DeleteFieldsDefinition** complex type 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

The **FieldReferenceDefinition** complex type 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

The **ListDefinitionCT** complex type specifies information about a particular **list**.

```
    <s:complexType name="ListDefinitionCT">
```

```xml
              <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: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:string" />
              <s:attribute name="WebImageWidth" type="s:string" />
              <s:attribute name="WebImageHeight" type="s:string" />
              <s:attribute name="Flags" type="s:int" />
              <s:attribute name="ItemCount" type="s:int" use="required" />
              <s:attribute name="AnonymousPermMask" type="unsignedLong" />
              <s:attribute name="RootFolder" type="s:string" />
              <s:attribute name="ReadSecurity" type="s:int" use="required" />
              <s:attribute name="WriteSecurity" type="s:int" use="required" />
              <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="core:TRUEFALSE" />
              <s:attribute name="NoThrottleListOperations" type="core:TRUEFALSE" />
              <s:attribute name="HasRelatedLists" type="s:string" />
              <s:attribute name="AllowDeletion" type="core:TRUEFALSE" />
              <s:attribute name="AllowMultiResponses" type="core:TRUEFALSE" />
              <s:attribute name="EnableAttachments" type="core:TRUEFALSE" />
              <s:attribute name="EnableModeration" type="core:TRUEFALSE" />
              <s:attribute name="EnableVersioning" type="core:TRUEFALSE" />
              <s:attribute name="HasExternalDataSource" type="core:TRUEFALSE" />
              <s:attribute name="Hidden" type="core:TRUEFALSE" />
              <s:attribute name="MultipleDataList" type="core:TRUEFALSE" />
              <s:attribute name="Ordered" type="core:TRUEFALSE" />
              <s:attribute name="ShowUser" type="core:TRUEFALSE" />
              <s:attribute name="EnablePeopleSelector" type="core:TRUEFALSE" />
              <s:attribute name="EnableResourceSelector" type="core:TRUEFALSE" />
              <s:attribute name="EnableMinorVersion" type="core:TRUEFALSE" />
              <s:attribute name="RequireCheckout" type="core:TRUEFALSE" />
              <s:attribute name="ThrottleListOperations" type="core:TRUEFALSE" />
              <s:attribute name="ExcludeFromOfflineClient" type="core:TRUEFALSE" />
              <s:attribute name="CanOpenFileAsync" type="core:TRUEFALSE" />
              <s:attribute name="EnableFolderCreation" type="core:TRUEFALSE" />
              <s:attribute name="IrmEnabled" type="core:TRUEFALSE" />
              <s:attribute name="IsApplicationList" type="core:TRUEFALSE" />
              <s:attribute name="PreserveEmptyValues" type="core:TRUEFALSE" />
              <s:attribute name="StrictTypeCoercion" type="core:TRUEFALSE" />
              <s:attribute name="EnforceDataValidation" type="core:TRUEFALSE" use="optional"/>
              <s:attribute name="MaxItemsPerThrottledOperation" type="s:int" />

              <s:attribute name="EnableAssignedToEmail" type="core:TRUEFALSE" use="optional"/>
              <s:attribute name="Followable" type="core:TRUEFALSE" />
              <s:attribute name="Acl" type="s:string" use="optional"/>
```

```
        <s:attribute name="Flags2" type="s:int" use="optional"/>
        <s:attribute name="RootFolderId" type="s:string" use="optional"/>
        <s:attribute name="IrmSyncable" type="core:TRUEFALSE" 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: The server-relative path to the page containing the **default mobile list view**.

ListDefinitionCT.MobileDefaultViewUrl: The 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 internal name for the list. The **Name** is equal to **ID**.

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-WSSFO3] section 2.2.1.2.12 for the possible values.

ListDefinitionCT.Created: Specifies the **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 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 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. This element 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.

ListDefinitionCT.WebImageWidth: The display width of an image in pixels. Used by picture libraries.

ListDefinitionCT.WebImageHeight: The display height of an image in pixels. Used by picture libraries.

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 only major versions.

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. Otherwise is "False". When it is returned by **GetListCollection** (section 3.1.4.17) this value will be an empty string.

ListDefinitionCT.AllowDeletion: True, if this list can 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** can 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.CanOpenFileAsync: True, if the client attempts to open files asynchronously from the server. By default, it will return **True** to the client.<8>

ListDefinitionCT.EnableFolderCreation: True, if folder creation is enabled on the list.<9>

ListDefinitionCT.IrmEnabled: True, if **Information Rights Management (IRM)** is enabled on the list.<10>

ListDefinitionCT.IsApplicationList: True, if the list is an application list.<11>

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.<12>

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.<13>

ListDefinitionCT.MaxItemsPerThrottledOperation: Specifies the value of the query size threshold for the **Web application**.<14>

ListDefinitionCT.EnableAssignedToEmail: This attribute 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**, the protocol server SHOULD enforce the **field** properties that limit the data values that can be stored in that field.<15>

ListDefinitionCT.Followable: If **True**, the list can be followed in an activity feed.<16>

ListDefinitionCT.Acl: Specifies the **access control list (ACL)** information.<17>

ListDefinitionCT.Flags2: Additional Bit Flags for the list.<18> This element MUST be one of the following values.

Value	Description
0	Default value.
0x0000000000000001	The document parser is disabled on the list.
0x0000000000000002	The crawler indexes the non-default views of the list.
0x0000000000000004	The list has restricted item.
0x0000000000000008	The files in the list can be processed in asynchronous manner.

ListDefinitionCT.RootFolderId: The GUID of the root folder of the list.<19>

ListDefinitionCT.IrmSyncable: **True**, if Information Rights Management (IRM) can be synchronized on the list.<20>

2.2.4.12   ListDefinitionSchema

The **ListDefinitionSchema** complex type specifies the results from retrieving a specified **list**.

```
<s:complexType name="ListDefinitionSchema">
  <s:complexContent>
        <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: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:extension>
  </s:complexContent>
</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. If present, this element MUST not be longer than 1024 characters.<21>

ListDefinitionSchema.Validation.Message: A message used to indicate that **data validation** was not satisfied. If present, this element MUST not be longer than 1024 characters.<22>

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. Constants MUST be formatted according to the **Locale** attribute of the **RegionalSettings** element. Commas MUST be replaced with the appropriate separator character according to the **Locale** attribute of the **RegionalSettings** element.<23>

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 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 **UTC**.

ListDefinitionSchema.SortOrder: The Regional Settings Collation 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

The **ListItemChangeDefinition** complex type specifies the change to a **list item**. The value of the element specifies the identifier of the item in the **list**.

```
<s:complexType name="ListItemChangeDefinition" mixed="true">
  <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>
```

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.<24>

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.<25>

ServerChangeUnit: Specifies the synchronization version, as described in [MS-XSSK] section 1.3, for the change.<26> This attribute is not returned when the element is contained in the **GetListItemChangesSinceTokenResult** element.

2.2.4.14   UpdateListFieldResults

The **UpdateListFieldResults** complex type 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

The **UpdateListFieldsRequest** complex type specifies the operation to add, update, or delete **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>
```

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**<27> or an empty string. If the parameter is not presented, the protocol server MUST NOT add the fields to any view.

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

2.2.4.16   ViewChangeDefinition

The **ViewChangeDefinition** complex type 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 multiple 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**.
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

The **ChangeTypeEnum** simple type 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 described in the following table, which is indicated in the response.

Change	Description
`<Id ChangeType="InvalidToken" />`	Specifies that the **changeToken** parameter for the **GetListItemChangesSinceToken** operation (section 3.1.4.22) is either invalid or expired, or the **knowledge** parameter for the **GetListItemChangesWithKnowledge** operation (section 3.1.4.23) 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. The protocol client MUST then 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, 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 using the **ExtraIds** query option for

Change Type	Description
	list items. If the change was made to a view, the updated view can be obtained from the server by calling **GetListAndView** operation, 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 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

The **EnumOptimizeFor** simple type 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

The **EnumRecurrencePatternXMLVersion** simple type 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;'
/>
```

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;'
/>
```

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

2.2.5.4 EnumViewAttributes

The **EnumViewAttributes** simple type specifies values to restrict the data returned from **document libraries**. The values here MUST be used by the protocol server to restrict the data returned in 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>
```

Recursive: MUST return only those items one level deep.

RecursiveAll: MUST return a flat view of all documents and folders.

FilesOnly: MUST return only documents, and not the **folders**.

2.2.5.5 MethodCmdEnum

The **MethodCmdEnum** simple type 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

The **OnErrorEnum** simple type 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 further methods from running after the first error is encountered.

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

2.2.5.7 TRUEONLY

The **TRUEONLY** simple type 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.

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 identifier.
AddListFromFeature	Creates a list in the current site based on the specified list name, description, **feature identifier**, and **list template identifier**.

Operation	Description
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 a file from a document library.
CreateContentType	Creates a new content type on a list.
DeleteAttachment	Removes an attachment from s specified list item in a specified list.
DeleteContentType	Removes a content type from a specific list.
DeleteContentTypeXmlDocument	Deletes an **XML document** from an 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 a specified list.
GetListAndView	Returns the properties and fields for a specified list, and the properties for a 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 the **GetListItemChangesSinceToken** (section 3.1.4.22) or **GetListItemChangesWithKnowledge** (section 3.1.4.23) operations 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 a specified field of a list.
UndoCheckOut	Undoes the checkout of a specified file in a document library.
UpdateContentType	Updates a content type on a list.

Operation	Description
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 (section 3.1.4.1.1.1) and the server responds with an **AddAttachmentSoapOut** response message (section 3.1.4.1.1.2), 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** does not correspond to a list from either of these checks, the protocol server SHOULD<28> 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** 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.

5. If the specified **fileName** is not unique for the specified list item's attachment collection, the protocol server MUST return a SOAP fault with error code 0x81020067. This indicates that the specified **fileName** is already in use.

6. 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.

7. 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.

8. 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.

9. 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

The **AddAttachmentSoapIn** message is the request of the **AddAttachment** operation (section 3.1.4.1).

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 (section 3.1.4.1.2.1).

3.1.4.1.1.2   AddAttachmentSoapOut

The **AddAttachmentSoapOut** message is the response of the **AddAttachment** operation (section 3.1.4.1).

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 (section 3.1.4.1.2.2).

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.

Element	Description
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

The **AddAttachment** element specifies the details for adding an **attachment** to a **list item**.

```
<s:element name="AddAttachment">
  <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="fileName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" 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

The **AddAttachmentResponse** 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

The **AddDiscussionBoardItem** operation<29> 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 (section 3.1.4.2.1.1) and the protocol server responds with an **AddDiscussionBoardItemSoapOut** response message (section 3.1.4.2.1.2), 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 the specified **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 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

The **AddDiscussionBoardItemSoapIn** message is the request that contains information required by the **AddDiscussionBoardItem** operation (section 3.1.4.2).

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 (section 3.1.4.2.2.1).

3.1.4.2.1.2   AddDiscussionBoardItemSoapOut

The **AddDiscussionBoardItemSoapOut** message is the response that contains the results returned by the **AddDiscussionBoardItem** operation (section 3.1.4.2).

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 (section 3.1.4.2.2.2).

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

The **AddDiscussionBoardItem** 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 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>
```

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

The **AddDiscussionBoardItemResponse** 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:anyAttribute />
            </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.

The protocol server response included in the **listitems** element is modeled on a persistence format as specified in [MS-PRSTFR], excluding the **<s:schema>** element. The **listitems** element includes attributes describing the namespaces for the 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** (section 2.2.4.7).

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 (section 3.1.4.3.1.1) and the server responds with an **AddListSoapOut** response message (section 3.1.4.3.1.2), as follows:

1. If the **templateID** provided is less than zero, 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. 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, 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**. 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

The **AddListSoapIn** message is the request to create a new list labelled **listName** on the site using the **AddList** operation (section 3.1.4.3).

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 (section 3.1.4.3.2.1).

3.1.4.3.1.2   AddListSoapOut

The **AddListSoapOut** message is the response that confirms the addition of a new list after a call to the **AddList** operation (section 3.1.4.3).

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 (section 3.1.4.3.2.2).

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

The **AddList** 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 [MS-WSSFO2] section 2.2.3.12. The value of **templateID** refers to one of the identifiers in the referenced list.

3.1.4.3.2.2   AddListResponse

The **AddListResponse** element contains the response that is sent to the protocol client after a successful **AddList** call (section 3.1.4.3).

```
<s:element name="AddListResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" 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<31> 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 (section 3.1.4.4.1.1), and the server responds with an **AddListFromFeatureSoapOut** response message (section 3.1.4.4.1.2), as follows:

1. If the **templateID** provided is less than zero, 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 SHOULD<32> return a SOAP fault with the 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 is not 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. 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, 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 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

The **AddListFromFeatureSoapIn** 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 (section 3.1.4.4.2.1).

3.1.4.4.1.2   AddListFromFeatureSoapOut

The **AddListFromFeatureSoapOut** message is the response that confirms the addition of a new list after a call to **AddList** (section 3.1.4.3).

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 (section 3.1.4.4.2.2).

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

The **AddListFromFeature** 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 (section 3.1.4.3) 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

The **AddListFromFeatureResponse** element contains the response that is sent to the protocol client after a successful **AddListFromFeature** call (section 3.1.4.4).

```
<s:element name="AddListFromFeatureResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" 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.

3.1.4.5 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

The **AddWikiPageSoapIn** message MUST NOT be used.

3.1.4.5.1.2   AddWikiPageSoapOut

The **AddWikiPageSoapOut** 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

The **AddWikiPage** 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

The **AddWikiPageResponse** 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>
```

3.1.4.6 ApplyContentTypeToList

The **ApplyContentTypeToList** operation<33> 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 (section 3.1.4.6.1.1) and the protocol server responds with an **ApplyContentTypeToListSoapOut** response message (section 3.1.4.6.1.2), 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<34> 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

The **ApplyContentTypeToListSoapIn** message is the request of the **ApplyContentTypeToList** operation (section 3.1.4.6).

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 (section 3.1.4.6.2.1).

3.1.4.6.1.2   ApplyContentTypeToListSoapOut

The **ApplyContentTypeToListSoapOut** message is the response of the **ApplyContentTypeToList** operation (section 3.1.4.6).

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 (section 3.1.4.6.2.2).

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

The **ApplyContentTypeToList** 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.

3.1.4.6.2.2   ApplyContentTypeToListResponse

The **ApplyContentTypeToListResponse** 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 minOccurs="0" maxOccurs="1" 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

The **CheckInFile** operation<35> 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 (section 3.1.4.7.1.1) and the protocol server responds with a **CheckInFileSoapOut** response message (section 3.1.4.7.1.2), 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.

2. If the **pageUrl** is an invalid **URL**, the protocol server SHOULD<36> 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.

3. If the **pageUrl** does not refer to a **document**, the protocol server MUST return a SOAP fault. There is no error code returned for this fault.

4. If the **CheckinType** element is an empty string, the protocol server MUST return a SOAP fault. There is no error code for this fault.

5. 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.

6. 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

The **CheckInFileSoapIn** 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 (section 3.1.4.7.2.1).

3.1.4.7.1.2   CheckInFileSoapOut

The **CheckInFileSoapOut** 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 (section 3.1.4.7.2.2).

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

The **CheckInFile** 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 minOccurs="0" maxOccurs="1" name="pageUrl" type="s:string" />
      <s:element name="comment" type="s:string" minOccurs="0"/>
      <s:element minOccurs="0" maxOccurs="1" 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

The **CheckInFileResponse** element contains the response that is sent to the protocol client after a successful **CheckInFile** operation (section 3.1.4.7).

```
<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 this element 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

The **CheckOutFile** operation<37> 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>
```

The protocol client sends a **CheckOutFileSoapIn** request message (section 3.1.4.8.1.1) and the protocol server responds with a **CheckOutFileSoapOut** response message (section 3.1.4.8.1.2), 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 SHOULD<38> return a SOAP fault with error string "Invalid URI: The format of the URI could not be determined.". There is no error code for this fault.

3. If the **pageUrl** does not refer to a document, 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** elements, 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

The **CheckOutFileSoapIn** 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 (section 3.1.4.8.2.1).

3.1.4.8.1.2   CheckOutFileSoapOut

The **CheckOutFileSoapOut** 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 (section 3.1.4.8.2.2).

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

The **CheckOutFile** 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 minOccurs="0" maxOccurs="1" name="pageUrl" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" 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: This element 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".<39> 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 date format is passed, this parameter is ignored.

3.1.4.8.2.2   CheckOutFileResponse

The **CheckOutFileResponse** element contains the response that is sent to the protocol client after a successful **CheckOutFile** call (section 3.1.4.8).

```
<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 this element 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

The **CreateContentType** operation<40> 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 (section 3.1.4.9.1.1) and the protocol server responds with a **CreateContentTypeSoapOut** response message (section 3.1.4.9.1.2), 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** 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 **listName** specified does not exist or it could 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.

6. The protocol server adds the given fields to the newly created content type. If an error regarding the fields is encountered during this operation, the protocol server MUST ignore this error. 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. 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

The **CreateContentTypeSoapIn** message is the request of the **CreateContentType** operation (section 3.1.4.9).

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

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

The **SOAP body** contains a **CreateContentType** element (section 3.1.4.9.2.1).

3.1.4.9.1.2   CreateContentTypeSoapOut

The **CreateContentTypeSoapOut** message is the response of the **CreateContentType** operation (section 3.1.4.9).

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 (section 3.1.4.9.2.2).

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

The **CreateContentType** element specifies details for the request to create a **content type** on the list.

```
<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 name="fields" type="tns:AddOrUpdateFieldsDefinition"
                  minOccurs="0"/>
       <s:element minOccurs="0" maxOccurs="1" name="contentTypeProperties">
       <s:complexType mixed="true">
         <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>
```

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 0x000000b7 MUST be returned if it equals the name of an existing content type on the list, ignoring case. 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 is longer than 128 characters.

- It ends with a period, or begins or ends with a 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. This element 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

The **CreateContentTypeResponse** 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 minOccurs="0" maxOccurs="1" name="CreateContentTypeResult" type="s:string"
/>
    </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

The **DeleteAttachment** 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>
```

The protocol client sends a **DeleteAttachmentSoapIn** request message (section 3.1.4.10.1.1) and the protocol server responds with a **DeleteAttachmentSoapOut** response message (section 3.1.4.10.1.2), 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**. There is no error code 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 the specified **listName** does not correspond to a list from either of these checks, the protocol server SHOULD<42> 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

The **DeleteAttachmentSoapIn** message is the request of the **DeleteAttachment** operation (section 3.1.4.10).

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 (section 3.1.4.10.2.1).

3.1.4.10.1.2 DeleteAttachmentSoapOut

The **DeleteAttachmentSoapOut** message is the response of the **DeleteAttachment** operation (section 3.1.4.10).

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 (section 3.1.4.10.2.2).

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

The **DeleteAttachment** 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 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>
```

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 (section 3.1.4.14) to the protocol server with the same **listName** and **listItemID**.

3.1.4.10.2.2 DeleteAttachmentResponse

The **DeleteAttachmentResponse** 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

The **DeleteContentType** operation<43> 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 (section 3.1.4.11.1.1) and the protocol server responds with a **DeleteContentTypeSoapOut** response message (section 3.1.4.11.1.2), 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** does not correspond to a list from either of these checks, the protocol server SHOULD<44> 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

The **DeleteContentTypeSoapIn** message is the request of the **DeleteContentType** operation (section 3.1.4.11).

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

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

The **SOAP body** contains a **DeleteContentType** element (section 3.1.4.11.2.1).

3.1.4.11.1.2 DeleteContentTypeSoapOut

The **DeleteContentTypeSoapOut** message is the response of the **DeleteContentType** operation (section 3.1.4.11).

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

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

The **SOAP body** contains a **DeleteContentTypeResponse** element (section 3.1.4.11.2.2).

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

The **DeleteContentType** element specifies details for the request to delete the **content type** from the list.

```
<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>
```

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

The **DeleteContentTypeResponse** 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>
```

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

The **DeleteContentTypeXmlDocument** operation<45> 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 (section 3.1.4.12.1.1) and the protocol server responds with a **DeleteContentTypeXmlDocumentSoapOut** response message (section 3.1.4.12.1.2), 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** does not correspond to a list from either of these checks, the protocol server SHOULD<46> 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 the **DeleteContentTypeXmlDocumentResult** element (section 3.1.4.12.2.2).

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 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

The **DeleteContentTypeXmlDocumentSoapIn** message is the request of the **DeleteContentTypeXmlDocument** operation (section 3.1.4.12).

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 (section 3.1.4.12.2.1).

3.1.4.12.1.2 DeleteContentTypeXmlDocumentSoapOut

The **DeleteContentTypeXmlDocumentSoapOut** message is the response of the **DeleteContentTypeXmlDocument** operation (section 3.1.4.12).

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 (section 3.1.4.12.2.2).

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

The **DeleteContentTypeXmlDocument** 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 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>
```

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

The **DeleteContentTypeXmlDocumentResponse** 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, this element 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 (section 3.1.4.13.1.1), and the protocol server MUST responds with a **DeleteListSoapOut** response message (section 3.1.4.13.1.2), 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 (section 3.1.4.13.2.2).

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 the specified **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 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

The **DeleteListSoapIn** 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 (section 3.1.4.13.2.1).

3.1.4.13.1.2 DeleteListSoapOut

The **DeleteListSoapOut** message is the response that confirms the deletion of a list by the **DeleteList** operation (section 3.1.4.13).

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 (section 3.1.4.13.2.2).

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

The **DeleteList** element specifies details for the request to delete a list from the site.

```
<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>
```

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

3.1.4.13.2.2 DeleteListResponse

The **DeleteListResponse** element contains a single element that MUST be sent by the site if the **DeleteList** operation (section 3.1.4.13) succeeds.

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

3.1.4.14 GetAttachmentCollection

The **GetAttachmentCollection** 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 (section 3.1.4.14.1.1) and the protocol server responds with a **GetAttachmentCollectionSoapOut** response message (section 3.1.4.14.1.2), 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** does not correspond to a list from either of these checks, the protocol server SHOULD<48> 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 SHOULD<49> 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

The **GetAttachmentCollectionSoapIn** message is the request of the **GetAttachmentCollection** operation (section 3.1.4.14).

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 (section 3.1.4.14.2.1).

3.1.4.14.1.2 GetAttachmentCollectionSoapOut

The **GetAttachmentCollectionSoapOut** message is the response of the **GetAttachmentCollection** operation (section 3.1.4.14).

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 (section 3.1.4.14.2.2).

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 list.

3.1.4.14.2.1 GetAttachmentCollection

The **GetAttachmentCollection** 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 minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
```

```
            <s:element minOccurs="0" maxOccurs="1" 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

The **GetAttachmentCollectionResponse** 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" type="s:string" minOccurs="0"
                             maxOccurs="unbounded">
                    </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 (section 3.1.4.15.1.1) and the protocol server MUST respond with a **GetListSoapOut** response message (section 3.1.4.15.1.2), 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 (section 3.1.4.15.2.2).

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 the specified **listName** does not correspond to a list from either of these checks, the protocol server SHOULD<50> 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

The **GetListSoapIn** message is the request that contains information required by the **GetList** operation (section 3.1.4.15).

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 (section 3.1.4.15.2.1).

3.1.4.15.1.2 GetListSoapOut

The **GetListSoapOut** message is the response that contains the results returned by the **GetList** operation (section 3.1.4.15).

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 (section 3.1.4.15.2.1).

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

Element	Description
	for a specified list.

3.1.4.15.2.1 GetList

The **GetList** 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 minOccurs="0" maxOccurs="1" 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

The **GetListResponse** 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

The **GetListAndView** 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 (section 3.1.4.16.1.1) and the server MUST respond with a **GetListAndViewSoapOut** response message (section 3.1.4.16.1.2), 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 **GetListAndViewResponse** element.

3. If the specified **listName** does not correspond to a list from either of these checks, the protocol server SHOULD<51> 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

The **GetListAndViewSoapIn** message is the request that contains the information required by the **GetListAndView** operation (section 3.1.4.16).

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 (section 3.1.4.16.2.1).

3.1.4.16.1.2 GetListAndViewSoapOut

The **GetListAndViewSoapOut** message is the response that contains the results returned by the **GetListAndView** operation (section 3.1.4.16).

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 (section 3.1.4.16.2.2).

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

The **GetListAndView** 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 minOccurs="0" maxOccurs="1" 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

The **GetListAndViewResponse** 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

The **GetListCollection** 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 (section 3.1.4.17.1.1) and the server responds with a **GetListCollectionSoapOut** response message (section 3.1.4.17.1.2).

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

The **GetListCollectionSoapIn** message is the request that contains information required by the **GetListCollection** operation (section 3.1.4.17).

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 (section 3.1.4.17.2.1).

3.1.4.17.1.2 GetListCollectionSoapOut

The **GetListCollectionSoapOut** message is the response that contains the results returned by the **GetListCollection** operation (section 3.1.4.17).

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 (section 3.1.4.17.2.2).

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

The **GetListCollection** 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

The **GetListCollectionResponse** 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>
    </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

The **GetListContentType** operation<52> 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 (section 3.1.4.18.1.1) and the protocol server responds with a **GetListContentTypeSoapOut** response message (section 3.1.4.18.1.2), 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** 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 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** element (section 3.1.4.18.2.2).

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

The **GetListContentTypeSoapIn** 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 (section 3.1.4.18.2.1).

3.1.4.18.1.2 GetListContentTypeSoapOut

The **GetListContentTypeSoapOut** 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 (section 3.1.4.18.2.2).

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

The **GetListContentType** element specifies details for the request to return data for a specific **content type**.

```
<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>
```

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

The **GetListContentTypeResponse** element contains the response to the request to return **content type** data.

```
<s:element name="GetListContentTypeResponse">
  <s:complexType>

    <s:sequence>
```

```
        <s:element minOccurs="0" maxOccurs="1" name="GetListContentTypeResult">
          <s:complexType mixed="true">
             <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

The **GetListContentTypes** operation<54> 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 (section 3.1.4.19.1.1) and the protocol server responds with a **GetListContentTypesSoapOut** message (section 3.1.4.19.1.2).

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

The **GetListContentTypesSoapIn** 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 (section 3.1.4.19.2.1).

3.1.4.19.1.2 GetListContentTypesSoapOut

The **GetListContentTypesSoapOut** 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 (section 3.1.4.19.2.2).

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

The **GetListContentTypes** 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 minOccurs="0" maxOccurs="1" 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<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. 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 (section 3.1.4.19.2.2).

- 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

The **GetListContentTypesResponse** 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: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: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**.<56>

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 one 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, this element 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 (section 3.1.4.19.1.1).

3.1.4.20   GetListContentTypesAndProperties

The **GetListContentTypesAndProperties** 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<57> support this operation.

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

The protocol client sends a **GetListContentTypesAndPropertiesSoapIn** message (section 3.1.4.20.1.1) and the protocol server responds with a **GetListContentTypesAndPropertiesSoapOut** message (section 3.1.4.20.1.2).

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

The **GetListContentTypesAndPropertiesSoapIn** 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 (section 3.1.4.20.2.1).

3.1.4.20.1.2 GetListContentTypesAndPropertiesSoapOut

The **GetListContentTypesAndPropertiesSoapOut** message is the response to the **GetListContentTypesAndPropertiesSoapIn** request (section 3.1.4.20.1.1).

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 (section 3.1.4.20.2.2).

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

The **GetListContentTypesAndProperties** element specifies the details of the **GetListContentTypesAndPropertiesSoapIn** request (section 3.1.4.20.1.1).

```
<s:element name="GetListContentTypesAndProperties">
  <s:complexType>
    <s:sequence>
        <s:element minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
        <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 (section 3.1.4.20.2.2). 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

The **GetListContentTypesAndPropertiesResponse** element provides the **content types** and properties requested by the **GetListContentTypesAndPropertiesSoapIn** message (section 3.1.4.20.1.1).

```
<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>
```

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**.

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, that 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 one 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, 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 3.1.4.19.2.1 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

The **GetListItemChanges** 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 the **GetListItemChangesSinceToken** operation (section 3.1.4.22) 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 (section 3.1.4.21.1.1) and the protocol server responds with a **GetListItemChangesSoapOut** response message (section 3.1.4.21.1.2), 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 SHOULD<58> 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<59> 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 **GetListItemsChangesSoapIn** request (section 3.1.4.21.1.1).

3.1.4.21.1.1   GetListItemChangesSoapIn

The **GetListItemChangesSoapIn** message is the request of the **GetListItemChanges** operation (section 3.1.4.21).

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 (section 3.1.4.21.2.1).

3.1.4.21.1.2   GetListItemChangesSoapOut

The **GetListItemChangesSoapOut** message is the response of the **GetListItemChanges** operation (section 3.1.4.21).

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 (section 3.1.4.21.2.2).

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

Element	Description
	specified filter criteria.
GetListItemChangesResponse	Contains the response to a **GetListItemsChangesSoapIn** request, as specified in section 3.1.4.21.1.1.

3.1.4.21.2.1 GetListItemChanges

The **GetListItemChanges** 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 minOccurs="0" maxOccurs="1" 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 in the **default view** 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, the 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 (section 3.1.4.21). 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 **UTC** format, the date MUST be treated by the protocol server as being in the protocol server's local **time zone** and it MUST be 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

The **GetListItemChangesResponse** element contains the protocol server response for the **GetListItemChanges** request (section 3.1.4.21).

```
<s:element name="GetListItemChangesResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetListItemChangesResult">
        <s:complexType mixed="true">
          <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, that 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** (section 2.2.4.7).

The **TimeStamp** attribute is a string that contains the date in **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** operation (section 3.1.4.15.2.1), and are prefixed by "ows_". Note that the 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

The **GetListItemChangesSinceToken** operation<60> 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 (section 3.1.4.22.1.1) and the protocol server responds with a **GetListItemChangesSinceTokenSoapOut** response message (section 3.1.4.22.1.2), 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** does not correspond to a list from either of these checks, the protocol server SHOULD<61> 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 **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 SHOULD<62> 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<63>. If the value of the **changeToken** parameter is not empty and 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.
GetListItemChangesSinceTokenSoapOut	The response to a **GetListItemChangesSinceTokenSoapIn** request.

3.1.4.22.1.1 GetListItemChangesSinceTokenSoapIn

The **GetListItemChangesSinceTokenSoapIn** message is the request of the **GetListItemChangesSinceToken** operation (section 3.1.4.22) 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 (section 3.1.4.22.2.1).

3.1.4.22.1.2 GetListItemChangesSinceTokenSoapOut

The **GetListItemChangesSinceTokenSoapOut** message is the response of the **GetListItemChangesSinceToken** operation (section 3.1.4.22) 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 (section 3.1.4.22.2.2).

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

The **GetListItemChangesSinceToken** 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 minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element name="viewName" type="s:string" minOccurs="0" maxOccurs="1"/>
      <s:element minOccurs="0" maxOccurs="1" name="query">
        <s:complexType mixed="true">
          <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 zero 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 because they are returned in the **Changes** element.

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 the **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** (section 3.1.4.22). 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 paging mechanism specified in the **ListItemCollectionPositionNext** attribute (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 40, only the first 40 items that changed are returned<64>. The **Changes** element in the response MUST include the attribute **MoreChanges** set to "TRUE" to indicate that protocol clients MAY make another call to the **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. If both parameters 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** (section 3.1.4.22.2.2):

```
<?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

The **GetListItemChangesSinceTokenResponse** element contains the protocol server response for the **GetListItemChangesSinceToken** request (section 3.1.4.22).

```
<s:element name="GetListItemChangesSinceTokenResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetListItemChangesSinceTokenResult">
        <s:complexType mixed="true">
          <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:attribute name="LastChangeToken" type="s:string" />
                      <s:attribute name="MoreChanges" type="core:TRUEFALSE" />


                    </s:complexType>
                  </s:element>
                  <s:any />
                </s:sequence>
                <s:attribute name="MinTimeBetweenSyncs" type="s:unsignedInt" />
                <s:attribute name="RecommendedTimeBetweenSyncs" type="s:unsignedInt" />
                <s:attribute name="MaxBulkDocumentSyncSize" type="s:unsignedInt" />
                <s:attribute name="MaxRecommendedEmbeddedFileSize "type="s:unsignedInt" />
                <s:attribute name="AlternateUrls" type="s:string" />
                <s:attribute name="EffectivePermMask" type="s:string" />
              </s:complexType>
            </s:element>
          </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, that MAY be ignored by the protocol client, as well as the **Changes** element. The following attributes are specified for **Changes**.

Attribute	Definition
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

Attribute	Definition
	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 the **GetList** operation (section 3.1.4.15) if no change token was passed in the request or if there have been schema changes made to the list after the change token.<65> 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 the set of fields returned by the method is restricted by the **viewField** or **viewName** parameter.

The **ID** element in the **Changes** element represents the change to a list item, as specified in section 2.2.4.13.

The following attributes are specified for **listitems**.

Attribute	Definition
MinTimeBetweenSyncs	Represents the minimum amount of time between user-initiated 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".
RecommendedTimeBetweenSyncs	The recommended minimum amount of time in seconds 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.
MaxRecommendedEmbeddedFileSize	Client applications that are capable of both embedding a document inside another document or linking between documents, and which are capable of using a server-defined value to control this behavior MUST NOT embed documents larger than this value in other documents. The default is 500 megabytes.
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.

The **listitems** element also contains an inner element named **rs:data**, which is of type **DataDefinition ListItemCollectionPositionNext** (section 2.2.4.7).

The following is an example response from a **GetListItemChangesSinceToken** request (section 3.1.4.22):

```
<?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 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 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

The **GetListItemChangesWithKnowledge** 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.<66>

```
<wsdl:operation name="GetListItemChangesWithKnowledge">
    <wsdl:input message="GetListItemChangesWithKnowledgeSoapIn" />
    <wsdl:output message="GetListItemChangesWithKnowledgeSoapOut" />
</wsdl:operation>
```

The protocol client sends a **GetListItemChangesWithKnowledgeSoapIn** request message (section 3.1.4.23.1.1) and the protocol server responds with a **GetListItemChangesWithKnowledgeSoapOut** response message (section 3.1.4.23.1.2), 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** 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 could have been deleted by another user.

4. 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.

5. 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.

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. 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 (section 3.1.4.23).

3.1.4.23.1.1   GetListItemChangesWithKnowledgeSoapIn

The **GetListItemChangesWithKnowledgeSoapIn** message is the request of the **GetListItemChangesWithKnowledge** operation (section 3.1.4.23) 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/GetListItemChangesWithKnowledge

The **SOAP body** contains a **GetListItemChangesWithKnowledge** element (section 3.1.4.23.2.1).

3.1.4.23.1.2   GetListItemChangesWithKnowledgeSoapOut

The **GetListItemChangesWithKnowledgeSoapOut** message is the response of the **GetListItemChangesWithKnowledge** operation (section 3.1.4.23.2.1) 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 (section 3.1.4.23.2.2).

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

The **GetListItemChanagesWithKnowledge** 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 minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element name="viewName" type="s:string" minOccurs="0" maxOccurs="1"/>
      <s:element minOccurs="0" maxOccurs="1" name="query">
        <s:complexType mixed="true">
          <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" >
       <s:complexType>
          <s:sequence>
           <s:element ref="sync:syncKnowledge" />
          </s:sequence>
       </s:complexType>
      </s:element>
      <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 the **GetListItemChangesSinceToken** element (section 3.1.4.22.2.1).

viewName: See the description of the **viewName** element in **GetListItemChangesSinceToken**.

query: See the description of the **query** element in **GetListItemChangesSinceToken**.

viewFields: See the description of the **viewFields** element in **GetListItemChangesSinceToken**.

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 zero indicates that all rows SHOULD be retrieved, unless the **knowledge** parameter is specified. (See description of **knowledge** parameter for limitations of **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 because they are returned in the **Changes** element.

queryOptions: See the description of the **queryOptions** element in **GetListItemChangesSinceToken**.

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** (section 3.1.4.23) 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 paging mechanism specified in the **ListItemCollectionPositionNext** attribute (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, 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 **GetListItemChangesSinceToken**.

The following example shows a request that uses the knowledge extracted from a previous **GetListItemChangesWithKnowledgeResponse** response (section 3.1.4.23.2.2):

```
<?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

The **GetListItemChangesWithKnowledgeResponse** element contains the protocol server response for the **GetListItemChangesWithKnowledge** request (section 3.1.4.23).

```
<s:element name="GetListItemChangesWithKnowledgeResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetListItemChangesWithKnowledgeResult">
```

```
              <s:complexType mixed="true">
                <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 ref="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:complexType>
                        </s:element>
                        <s:any />
                      </s:sequence>
                      <s:attribute name="MinTimeBetweenSyncs" type="s:unsignedInt" />
                      <s:attribute name="RecommendedTimeBetweenSyncs" type="s:unsignedInt" />
                      <s:attribute name="MaxBulkDocumentSyncSize" type="s:unsignedInt" />
                      <s:attribute name="MaxRecommendedEmbeddedFileSize" type="s:unsignedInt" />
                      <s:attribute name="AlternateUrls" type="s:string" />
                      <s:attribute name="EffectivePermMask" type="s:string" />
                    </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
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 been schema changes made to the list.

Attribute	Definition
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 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, which 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 **ID** element in the **Changes** element represents the change to a list item, as specified in section 2.2.4.13.

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** operation (section 3.1.4.15), and are prefixed by "ows_". Note that the 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 following attributes are specified for **listtems**.

Attribute	Definition
MinTimeBetweenSyncs	See the description of the **MinTimeBetweenSyncs** attribute in **GetListItemChangesSinceTokenResponse** (section 3.1.4.22.2.2).
RecommendedTimeBetweenSyncs	See the description of the **RecommendedTimeBetweenSyncs** attribute in **GetListItemChangesSinceTokenResponse**.
MaxBulkDocumentSyncSize	See the description of the **MaxBulkDocumentSyncSize** attribute in **GetListItemChangesSinceTokenResponse**.
MaxRecommendedEmbeddedFileSize	Client applications that are capable of both embedding a document inside another document or linking between documents, and capable of using a server-defined value to control this behavior MUST NOT embed documents larger than this value in other documents. The default is 500 megabytes.<67>
AlternateUrls	See the description of the **AlternateUrls** attribute in **GetListItemChangesSinceTokenResponse**.
EffectivePermMask	See the description of the **EffectivePermMask** attribute in **GetListItemChangesSinceTokenResponse**.

The **listitems** element also contains an inner element named **rs:data**, which is of type **DataDefinition** (section 2.2.4.7).

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

The **GetListItems** 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 (section 3.1.4.24.1.1) and the protocol server responds with a **GetListItemsSoapOut** response message (section 3.1.4.24.1.2), 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 not empty and does not correspond to a list from either of these checks, the protocol server SHOULD<68> return a **SOAP fault** with error code 0x82000006. This indicates that the list does not exist or could have been deleted by another user.

4. If **listName** and **viewName** are both empty, the protocol server SHOULD<69> 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 with error code 0x82000005. 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

The **GetListItemsSoapIn** message is the request of the **GetListItems** operation (section 3.1.4.24) that the protocol client sends as a request to this Web protocol server.

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

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

The **SOAP body** contains a **GetListItems** element (section 3.1.4.24.2.1).

3.1.4.24.1.2 GetListItemsSoapOut

The **GetListItemsSoapOut** message is the response of the **GetListItems** operation (section 3.1.4.24) 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 (section 3.1.4.24.2.2).

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

The **GetListItems** element specifies details for 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 minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element name="viewName" type="s:string" minOccurs="0"/>
      <s:element minOccurs="0" maxOccurs="1" name="query">
        <s:complexType mixed="true">
          <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-VIEWSS]). If not specified, the **default list view** is used. The **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 will be used. If a field referenced in a **FieldRef** element of **viewFields** has no value filled, this field SHOULD NOT be returned. If the **Properties** attribute of the **ViewFields** structure is set to **TRUE** and the **MetaInfo** field is referenced in a **FieldRef** element, the 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

The **GetListItemsResponse** element contains the protocol server response for the **GetListItems** request (section 3.1.4.24).

```
<s:element name="GetListItemsResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="GetListItemsResult">
        <s:complexType mixed="true">
          <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: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, that MAY be ignored by the protocol client, and contains an inner element named **rs:data**, that is of type **DataDefinition** (section 2.2.4.7). Protocol clients SHOULD make a **GetList** call (section 3.1.4.15) 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 the 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='20.0000000000000'
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

The **GetVersionCollection** operation<70> 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 (section 3.1.4.25.1.1) and the protocol server responds with a **GetVersionCollectionSoapOut** response message (section 3.1.4.25.1.2), 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 the specified **strListID** does not correspond to a list from either of these checks, the protocol server SHOULD<71> 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. When **strListItemID** is a negative value, the error code is 0x81020014. When **strListItemID** is zero or a positive value that does not correspond to a list item in a specified list on the site, there is no error code.<72>

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

The **GetVersionCollectionSoapIn** message is the request of the **GetVersionCollection** operation (section 3.1.4.25).

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 (section 3.1.4.25.2.1).

3.1.4.25.1.2 GetVersionCollectionSoapOut

The **GetVersionCollectionSoapOut** message is the response of the **GetVersionCollection** operation (section 3.1.4.25).

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 (section 3.1.4.25.2.2).

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

The **GetVersionCollection** 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 minOccurs="0" maxOccurs="1" name="strlistID" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="strlistItemID" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" 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

The **GetVersionCollectionResponse** 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:attribute name="Editor" 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 is "Title" and the **Value** of the attribute is the value of the **Title** field for the specified list item.

Version.Modified: The date and time when this field was modified.

Version.Editor: The user that created the item version.

3.1.4.26   UndoCheckOut

The **UndoCheckOut** operation<73> 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>
```

The protocol client sends an **UndoCheckOutSoapIn** request message (section 3.1.4.26.1.1) and the protocol server responds with an **UndoCheckOutSoapOut** response message (section 3.1.4.26.1.2), 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 SHOULD[<74>](#) return a SOAP fault with the error string: "Invalid URI: The format of the URI could not be determined.". This indicates that the parameter **pageUrl** is invalid.

3. If the **pageUrl** does not refer to a document, 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

The **UpdoCheckOutSoapIn** message is the request to undo the 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 (section 3.1.4.26.2.1).

3.1.4.26.1.2 UndoCheckOutSoapOut

The **UpdoCheckOutSoapOut** message is the response that confirms the operation to undo the 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 (section 3.1.4.26.2.2).

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

The **UndoCheckOut** element specifies details for the request to undo the checkout of the document based on the provided **pageUrl**.

```
<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>
```

pageUrl: The **URL** of the document for which to undo the checkout.

3.1.4.26.2.2 UndoCheckOutResponse

The **UndoCheckOutResponse** element contains the response that is sent after a successful **UndoCheckOut** operation (section 3.1.4.26).

```
<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

The **UpdateContentType** operation<75> 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 (section 3.1.4.27.1.1) and the protocol server responds with an **UpdateContentTypeSoapOut** response message (section 3.1.4.27.1.2), 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 **listNam**e corresponds to the list title of a list on the site and, if so, use that list.

3. If the specified **listName** does not correspond to a list from either of these checks, a **SOAP fault** SHOULD<76> 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 the fields to be deleted do not exist in the specified list, the protocol server MUST return a **Response** with error code 0x82000007. This indicates that the fields specified in the **deleteFields** element do not exist in the specified list, and all fields that were deleted before the error was encountered MUST stay deleted.

6. The protocol server adds the appropriate fields to the content type. If the fields to be added do not exist in the specified list, the protocol server MUST return a **Response** with error code 0x82000007. This indicates that the fields specified in the **newFields** element do not exist in the specified list, 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.

7. The protocol server updates the appropriate fields on the content type. If the fields to be updated do not exist in the specified **contentType** or they do not exist in the specified list, the protocol server MUST return a **Response** with error code 0x82000007. This indicates that the fields specified in the **updateFields** element do not exist in the specified **contentType** or do not exist in the specified list, 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.

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. 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

The **UpdateContentTypeSoapIn** message is the request of the **UpdateContentType** operation (section 3.1.4.27).

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 (section 3.1.4.27.2.1).

3.1.4.27.1.2 UpdateContentTypeSoapOut

The **UpdateContentTypeSoapOut** message is the response of the **UpdateContentType** operation (section 3.1.4.27).

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 (section 3.1.4.27.2.2).

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

The **UpdateContentType** element specifies details for the request to update a **content type** on the list.

```
<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: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. This element 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

The **UpdateContentTypeResponse** 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 minOccurs="0" maxOccurs="1" name="UpdateContentTypeResult">
        <s:complexType mixed="true">
          <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"
                               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>
```

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, this element 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, this element 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

The **UpdateContentTypesXmlDocument** operation<77> 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 (section 3.1.4.28.1.1) and the protocol server responds with an **UpdateContentTypesXmlDocumentSoapOut** response message (section 3.1.4.28.1.2), 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** (section 3.1.4.28.2.2).

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<78> 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

The **UpdateContentTypesXmlDocumentSoapIn** message is the request of the **UpdateContentTypesXmlDocument** operation (section 3.1.4.28).

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 (section 3.1.4.28.2.1).

3.1.4.28.1.2 UpdateContentTypesXmlDocumentSoapOut

Th**UpdateContentTypesXmlDocumentSoapOut** message is the response of the **UpdateContentTypesXmlDocument** operation (section 3.1.4.28).

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 (section 3.1.4.28.2.2).

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 **UpdateContentTypesXmlDocument** element specifies the details for 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 minOccurs="0" maxOccurs="1" name="listName" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="newDocument">
        <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: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

The **UpdateContentTypesXmlDocumentResponse** 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

The **UpdateContentTypeXmlDocument** operation<79> 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 (section 3.1.4.29.1.1) and the protocol server responds with an **UpdateContentTypeXmlDocumentSoapOut** response message (section 3.1.4.29.1.2), 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<80> 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** element (section 3.1.4.29.2.2).

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

The **UpdateContentTypeXmlDocumentSoapIn** message is the request of the **UpdateContentTypeXmlDocument** operation (section 3.1.4.29).

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 (section 3.1.4.29.2.1).

3.1.4.29.1.2 UpdateContentTypeXmlDocumentSoapOut

The **UpdateContentTypeXmlDocumentSoapOut** message is the response of the **UpdateContentTypeXmlDocument** operation (section 3.1.4.29).

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 (section 3.1.4.29.2.2).

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

The **UpdateContentTypeXmlDocument** 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 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>
```

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 **URIs** [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

The **UpdateContentTypeXmlDocumentResponse** 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:element name="Success" minOccurs="0">
              <s:complexType />
            </s:element>
          </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.

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

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 (section 3.1.4.30.1.1), and the
server MUST respond with an **UpdateListSoapOut** response message (section 3.1.4.30.1.2), 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<81> 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
   UpdateListResponse 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, this
   operation 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

The **UpdateListSoapIn** message is the request that contains required parameters for the **UpdateList** operation (section 3.1.4.30).

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 (section 3.1.4.30.2.1).

3.1.4.30.1.2 UpdateListSoapOut

The **UpdateListSoapOut** message is the response that contains the results returned by the **UpdateList** operation (section 3.1.4.30).

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 (section 3.1.4.30.2.2).

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

The **UpdateList** 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: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:attribute name="EnableModeration" type="core:TRUEFALSE" />
                        <s:attribute name="EnableVersioning" type="core:TRUEFALSE" />
                        <s:attribute name="EnforceDataValidation" type="core:TRUEFALSE"
  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="core:TRUEFALSE" />
                        <s:attribute name="ShowUser" type="core:TRUEFALSE" />
                        <s:attribute name="StrictTypeCoercion" type="core:TRUEFALSE" />
                        <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**. 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
AllowMultiResponses	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.

Attribute	Description
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.
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.<82>
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.<83>
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. The **ListVersion** is set to the value of **Version** retrieved from a previous **GetListItemChangesSinceToken** call (section 3.1.4.22) with no **changeToken** parameter included, or a **GetListItemChangesWithKnowledge** call (section 3.1.4.23) with no **knowledge** parameter included or where the schema on the protocol server changed or from a call to the **GetList** (section 3.1.4.15) or **GetListAndView** (section 3.1.4.16) operations. If the **listVersion** value sent up in the **UpdateList** request does not match the current version, 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

The **UpdateListResponse** 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>
```

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 (section 3.1.4.31.1.1) and the protocol server responds with an **UpdateListItemsSoapOut** response message (section 3.1.4.31.1.2), 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<84> 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 SHOULD<85> return the SOAP fault: "Cannot access a closed Stream".

5. Otherwise, the protocol server MUST process the batched operations on the list, and return success or failure conditions per each 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

The **UpdateListItemsSoapIn** message is the request of the **UpdateListItems** operations (section 3.1.4.31) 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 (section 3.1.4.31.2.1).

3.1.4.31.1.2 UpdateListItemsSoapOut

The **UpdateListItemsSoapOut** message is the response of the **UpdateListItems** operation (section 3.1.4.31) 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 (section 3.1.4.31.2.2).

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.

Element	Description
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

The **UpdateListItems** element specifies the request that MUST be sent for the **UpdateListItems** operation (section 3.1.4.31).

```
<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: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:attribute name="Property" type="s:string"/>
                              </s:extension>
                            </s:simpleContent>
                          </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>
    </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**.

Batch: There are several attributes supported by this 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** is set to the value of **Version** retrieved from a previous **GetListItemChangesSinceToken** call (section 3.1.4.22) with no **changeToken** parameter included, or a **GetListItemChangesWithKnowledge** call (section 3.1.4.23) with no **knowledge** parameter included, or where the schema on the protocol server changed, or from a call to the **GetList** (section 3.1.4.15) or **GetListAndView** (section 3.1.4.16) operations.
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:
	
	`http://siteservice/Lists/TestListItem/TestNewFolder`
	
	Site-relative URL:
	
	`/Lists/TestListItem/TestNewFolder`

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. If specified, `<Field Name="ID">` MUST be ignored, 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: ▪ 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<86>	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** operation (section 3.1.4.15.2.1) 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 request 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.
`<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, 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** (section 3.1.4.24), **GetListItemChanges** (section 3.1.4.21), or **GetListItemChangesSinceToken** (section 3.1.4.22), or **GetListItemChangesWithKnowledge** (section 3.1.4.23) 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 0x81020015, 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="100.000000000000" 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="0x7ffffffffffffff" 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 that includes 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 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="100.000000000000" 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="0x7fffffffffffffff" 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 the **FSObjType** field to 1 and the **BaseName** field to the name of the folder. These two fields are defined in [MS-WSSTS] section 2.4.2.

```
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>
```

The following is an example of updating an existing item. The example updates the **Title** and **Number** fields for the list item whose ID is "1", and 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

The **UpdateListItemsResponse** element contains the protocol server response to the
UpdateListItems request (section 3.1.4.31).

```
<s:element name="UpdateListItemsResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="UpdateListItemsResult">
        <s:complexType mixed="true">
          <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:any minOccurs="0" maxOccurs="unbounded" />
                      </s:sequence>
                      <s:attribute name="ID" type="s:string" />
                      <s:attribute name="List" type="s:string"/>
                      <s:attribute name="Version" 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>
```

UpdateListItemsResult: If the **OnError** attribute of the **Batch** element is set to **Return**, there
MUST be a **Result** element for each **Method** operation. If a **Method** operation fails, 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**, 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 **owshiddenversion** contained in the **UpdateListItems** request plus 1.

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. An empty **ID** element following the **ErrorCode** element is included, which is reserved for future use. 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. <87>

```
<wsdl:operation name="UpdateListItemsWithKnowledge">
    <wsdl:input message="UpdateListItemsWithKnowledgeSoapIn" />
    <wsdl:output message="UpdateListItemsWithKnowledgeSoapOut" />
</wsdl:operation>
```

The protocol client sends an **UpdateListItemsWithKnowledgeSoapIn** request message (section 3.1.4.32.1.1) and the protocol server responds with an **UpdateListItemsWithKnowledgeSoapOut** response message (section 3.1.4.32.1.2), 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 the specified **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

The **UpdateListItemsWithKnowledgeSoapIn** message is the request of the **UpdateListItemsWithKnowledge** operation (section 3.1.4.32) 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 (section 3.1.4.32.2.1).

3.1.4.32.1.2 UpdateListItemsWithKnowledgeSoapOut

The **UpdateListItemsWithKnowledgeSoapOut** message is the response of the **UpdateListItemsWithKnowledge** operation (section 3.1.4.32) 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 (section 3.1.4.32.2.2).

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

The **UpdateListItemsWithKnowledge** element specifies the request that MUST be sent for the **UpdateListItemsWithKnowledge** operation (section 3.1.4.32).

```
<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: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:attribute name="Property" type="s:string"/>
                              </s:extension>
                            </s:simpleContent>
                          </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>
      <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

The **UpdateListItemsWithKnowledgeResponse** element contains the protocol server response to the **UpdateListItemsWithKnowledge** request (section 3.1.4.32).

```
    <s:element name="UpdateListItemsWithKnowledgeResponse">
      <s:complexType>
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="1" name="UpdateListItemsWithKnowledgeResult">
            <s:complexType mixed="true">
              <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:any minOccurs="0" maxOccurs="unbounded"/>
                          </s:sequence>
                          <s:attribute name="ID" type="s:string" />
                          <s:attribute name="List" type="s:string" />
                          <s:attribute name="Version" 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>
```

UpdateListItemsWithKnowledgeResult: See description of the **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 (section 3.1.4.18) 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** (section 3.1.4.17).

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>
      <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[<img align="absmiddle" style="cursor: pointer"
src="/_layouts/images/rbsel.gif" alt="]]>
</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[<img border="0" align="absmiddle" style="cursor: pointer"
src="/ layouts/images/rbunsel.gif"  alt="]]></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="{9F8B4EE0-84B7-42c6-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="{EDECEC70-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>
```

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** operation (section 3.1.4.28) 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 (section 3.1.4.28).

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 (section 3.1.4.12).

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 (section 3.1.4.3) 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** (section 3.1.4.22), 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 (section 3.1.4.31) 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="0x7ffffffffffffff"
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.tailspintoys.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
ffffffffff" 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="0x7fffffffffffffff" ows_FileLeafRef="3;#3_.00
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_Encod
edAbsUrl="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 the **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>
```

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.

6 Appendix A: Full WSDL

For ease of implementation, the full WSDL and schema are provided in this appendix.

```xml
<?xml version="1.0" encoding="UTF-8" ?>
<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: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: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: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: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: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>
```

```xml
            <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:element name="GetVersionCollection">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="strlistID" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="strlistItemID" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="strFieldName" type="s:string" />
```

```
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetVersionCollectionResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetVersionCollectionResult">
              <s:complexType mixed="true">
                <s:sequence>
                  <s:any />
                </s:sequence>
              </s:complexType>
            </s:element>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="AddAttachment">
        <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="fileName" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="attachment" type="s:base64Binary" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="AddAttachmentResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="AddAttachmentResult" type="s:string"
  />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetAttachmentCollection">
        <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:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetAttachmentCollectionResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="GetAttachmentCollectionResult">
              <s:complexType mixed="true">
                <s:sequence>
                  <s:any />
                </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>
```

```xml
          <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: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 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:any />
                </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>
```

```xml
              <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: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 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:definitions>
```

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 updates to those products.

- Microsoft Office 2003

- The 2007 Microsoft Office system

- Microsoft Office 2010 suites

- Microsoft Office 2013

- Windows SharePoint Services 2.0

- Windows SharePoint Services 3.0

- Microsoft SharePoint Foundation 2010

- Microsoft SharePoint Foundation 2013

- Windows 8.1 Update

- Microsoft Office 2016

- Windows 10 operating system

- Microsoft SharePoint Server 2016

- Microsoft Office 2019

- Microsoft SharePoint Server 2019

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

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

<1> Section 2.2.4.11: The **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: The **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: The **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, Windows SharePoint Services 3.0 and SharePoint Foundation 2010.

<9> Section 2.2.4.11: This attribute is not returned by Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 servers.

<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 2.2.4.11: This attribute is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<16> Section 2.2.4.11: This attribute is not returned in Windows SharePoint Services 2.0, Windows SharePoint Services 3.0 and SharePoint Foundation 2010.

<17> Section 2.2.4.11: This attribute is not returned in Windows SharePoint Services 2.0, Windows SharePoint Services 3.0, SharePoint Foundation 2010 and SharePoint Foundation 2013.

<18> Section 2.2.4.11: This attribute is not returned in Windows SharePoint Services 2.0, Windows SharePoint Services 3.0, SharePoint Foundation 2010 and SharePoint Foundation 2013.

<19> Section 2.2.4.11: This attribute is not returned in Windows SharePoint Services 2.0, Windows SharePoint Services 3.0, SharePoint Foundation 2010 and SharePoint Foundation 2013.

<20> Section 2.2.4.11: This attribute is not returned in Windows SharePoint Services 2.0, Windows SharePoint Services 3.0, SharePoint Foundation 2010 and SharePoint Foundation 2013.

<21> Section 2.2.4.12: This element is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<22> Section 2.2.4.12: This attribute is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<23> Section 2.2.4.12: This element is not returned in Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0.

<24> Section 2.2.4.13: This attribute is not returned in Windows SharePoint Services 3.0.

<25> Section 2.2.4.13: This attribute is not returned in Windows SharePoint Services 3.0.

<26> Section 2.2.4.13: This attribute is not returned in Windows SharePoint Services 3.0.

<27> Section 2.2.4.15: Windows SharePoint Services 3.0 does not support the **UpdateList** operation when the attribute **AddToView** is set and the value of the attribute **AddToView** is not a **GUID**.

<28> Section 3.1.4.1: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following **SOAP fault** with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<29> Section 3.1.4.2: Windows SharePoint Services 2.0 does not support this operation.

<30> Section 3.1.4.2: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<31> Section 3.1.4.4: Windows SharePoint Services 2.0 does not support this operation.

<32> Section 3.1.4.4: Windows SharePoint Services 3.0 returns a SOAP fault with error code 0x81072101.

<33> Section 3.1.4.6: Windows SharePoint Services 2.0 does not support this operation.

<34> Section 3.1.4.6: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<35> Section 3.1.4.7: Windows SharePoint Services 2.0 does not support this operation.

<36> Section 3.1.4.7: Windows SharePoint Services 3.0 does not return a SOAP fault.

<37> Section 3.1.4.8: Windows SharePoint Services 2.0 does not support this operation.

<38> Section 3.1.4.8: Windows SharePoint Services 3.0 does not return a SOAP fault.

<39> Section 3.1.4.8.2.1: Windows SharePoint Services 3.0 does not support this format.

<40> Section 3.1.4.9: Windows SharePoint Services 2.0 does not support this operation.

<41> Section 3.1.4.9: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<42> Section 3.1.4.10: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<43> Section 3.1.4.11: Windows SharePoint Services 2.0 does not support this operation.

<44> Section 3.1.4.11: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<45> Section 3.1.4.12: Windows SharePoint Services 2.0 does not support this operation.

<46> Section 3.1.4.12: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<47> Section 3.1.4.13: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<48> Section 3.1.4.14: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<49> Section 3.1.4.14: Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "Value does not fall within the expected range."

<50> Section 3.1.4.15: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<51> Section 3.1.4.16: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<52> Section 3.1.4.18: Windows SharePoint Services 2.0 does not support this operation.

<53> Section 3.1.4.18: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<54> Section 3.1.4.19: Windows SharePoint Services 2.0 does not support this operation.

<55> Section 3.1.4.19.2.1: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<56> Section 3.1.4.19.2.2: The attribute is not returned in Windows SharePoint Services 3.0.

<57> Section 3.1.4.20: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 do not support the **GetListContentTypesAndProperties** operation (section 3.1.4.20).

<58> Section 3.1.4.21: In Windows SharePoint Services 2.0, Windows SharePoint Services 3.0, and SharePoint Foundation 2010, this error code is not returned in the SOAP fault.

<59> Section 3.1.4.21: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<60> Section 3.1.4.22: Windows SharePoint Services 2.0 does not support this operation.

<61> Section 3.1.4.22: In Windows SharePoint Services 3.0, this error code is not returned in the SOAP fault.

<62> Section 3.1.4.22: Windows SharePoint Services 3.0 returns the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<63> Section 3.1.4.22: 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.

<64> Section 3.1.4.22.2.1: In Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0, if the **changeToken** parameter is specified and the total number of **list items** that have been inserted or updated is greater than 100, only the first 100 items that changed are returned.

<65> Section 3.1.4.22.2.2: In Windows SharePoint Services 3.0, the **FileFragment** element is not returned.

<66> Section 3.1.4.23: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 do not support this operation.

<67> Section 3.1.4.23.2.2: This attribute is not returned by Windows SharePoint Services 2.0, Windows SharePoint Services 3.0 and SharePoint Foundation 2010.

<68> Section 3.1.4.24: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<69> Section 3.1.4.24: In Windows SharePoint Services 2.0, Windows SharePoint Services 3.0, and SharePoint Foundation 2010, this error code is not returned in the SOAP fault.

<70> Section 3.1.4.25: Windows SharePoint Services 2.0 does not support this operation.

<71> Section 3.1.4.25: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<72> Section 3.1.4.25: When **strListItemID** is a negative value, is zero, or is a positive value that does not correspond to a list item in a specified list on the site, Windows SharePoint Services 3.0 returns a SOAP fault with no error code and the error string "Value does not fall within the expected range.".

<73> Section 3.1.4.26: Windows SharePoint Services 2.0 does not support this operation.

<74> Section 3.1.4.26: Windows SharePoint Services 3.0 does not return a SOAP fault.

<75> Section 3.1.4.27: Windows SharePoint Services 2.0 does not support this operation.

<76> Section 3.1.4.27: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<77> Section 3.1.4.28: Windows SharePoint Services 2.0 does not support this operation.

<78> Section 3.1.4.28: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<79> Section 3.1.4.29: Windows SharePoint Services 2.0 does not support this operation.

<80> Section 3.1.4.29: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<81> Section 3.1.4.30: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<82> 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.

<83> 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.

<84> Section 3.1.4.31: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 return the following SOAP fault with no error code: "GUID should contain 32 digits with 4 dashes (xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx)".

<85> Section 3.1.4.31: Windows SharePoint Services 3.0 returns the SOAP fault: "Object reference not set to an instance of an object".

<86> Section 3.1.4.31.2.1: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 do not support this behavior.

<87> Section 3.1.4.32: Windows SharePoint Services 2.0 and Windows SharePoint Services 3.0 do not support this method.

8   Change Tracking

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

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

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

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

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

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

Section	Description	Revision class
7 Appendix B: Product Behavior	Updated list of supported products.	Major

9 Index

T

U

V

W