[MS-COPYS]:

Copy Web Service Protocol

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation ("this documentation") for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.

- **Copyrights**. This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.

- **No Trade Secrets**. Microsoft does not claim any trade secret rights in this documentation.

- **Patents**. Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft Open Specifications Promise or the Microsoft Community Promise. If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.

- **Trademarks**. The names of companies and products contained in this documentation might be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.

- **Fictitious Names**. The example companies, organizations, products, domain names, email addresses, logos, people, places, and events that are depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

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

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

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	Minor	Updated 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	Editorial	Changed language and formatting in 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	2.04	None	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	2.04	None	No changes to the meaning, language, or formatting of the technical content.
3/18/2011	2.04	None	No changes to the meaning, language, or formatting of the technical content.
6/10/2011	3.0	Major	Significantly changed the technical content.
1/20/2012	3.1	Minor	Clarified the meaning of the technical content.
4/11/2012	3.1	None	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	3.2	Minor	Clarified the meaning of the technical content.
9/12/2012	3.2	None	No changes to the meaning, language, or formatting of the technical content.
10/8/2012	3.2	None	No changes to the meaning, language, or formatting of the technical content.
2/11/2013	3.2	None	No changes to the meaning, language, or formatting of the technical content.
7/30/2013	3.2	None	No changes to the meaning, language, or formatting of the technical content.
11/18/2013	3.2	None	No changes to the meaning, language, or formatting of the technical content.

Date	Revision History	Revision Class	Comments
2/10/2014	3.2	None	No changes to the meaning, language, or formatting of the technical content.
4/30/2014	3.3	Minor	Clarified the meaning of the technical content.
7/31/2014	3.4	Minor	Clarified the meaning of the technical content.
10/30/2014	3.5	Minor	Clarified the meaning of the technical content.
3/16/2015	4.0	Major	Significantly changed the technical content.
6/30/2015	5.0	Major	Significantly changed the technical content.
2/26/2016	6.0	Major	Significantly changed the technical content.
4/14/2016	7.0	Major	Significantly changed the technical content.
7/15/2016	7.0	None	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1   Introduction

The Copy Web Service Protocol is used to copy files, including associated metadata such as creation date and author, to one or more locations on a single protocol server or between different protocol servers.

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:

ASCII: The American Standard Code for Information Interchange (ASCII) is an 8-bit character-encoding scheme based on the English alphabet. ASCII codes represent text in computers, communications equipment, and other devices that work with text. ASCII refers to a single 8-bit ASCII character or an array of 8-bit ASCII characters with the high bit of each character set to zero.

base64 encoding: A binary-to-text encoding scheme whereby an arbitrary sequence of bytes is converted to a sequence of printable **ASCII** characters, as described in [RFC4648].

checked out: A publishing level that indicates that a document has been created and locked for exclusive editing by a user in a version control system.

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).

destination location: An Internationalized Resource Identifier (IRI) to which a file is copied. The file name portion of the destination location does not have to be the same as the file name portion of the source location. A file can be renamed as it is copied.

destination server: A protocol server to which a file is copied.

file: A single, discrete unit of content.

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].

Internationalized Resource Identifier (IRI): A resource identifier that conforms to the rules for Internationalized Resource Identifiers, as defined in [RFC3987].

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.

Meeting Workspace site: A SharePoint site that is based on a Meeting Workspace site template and has a template ID value of "2". A Meeting Workspace site is used for planning, posting, and working together on meeting materials.

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.

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.

source location: A server, disk, file, document, or other collection of information from which a file or data is copied.

Status-Code: A 3-digit integer result code in an HTTP response message, as described in [RFC2616].

Unicode: A character encoding standard developed by the Unicode Consortium that represents almost all of the written languages of the world. The **Unicode** standard [UNICODE5.0.0/2007] provides three forms (UTF-8, UTF-16, and UTF-32) and seven schemes (UTF-8, UTF-16, UTF-16 BE, UTF-16 LE, UTF-32, UTF-32 LE, and UTF-32 BE).

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

Web Services Description Language (WSDL): An XML format for describing network services as a set of endpoints that operate on messages that contain either document-oriented or procedure-oriented information. The operations and messages are described abstractly and are bound to a concrete network protocol and message format in order to define an endpoint. Related concrete endpoints are combined into abstract endpoints, which describe a network service. WSDL is extensible, which allows the description of endpoints and their messages regardless of the message formats or network protocols that are used.

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].

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

1.2 References

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

1.2.1 Normative References

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

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

[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

[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

[RFC3548] Josefsson, S., Ed., "The Base16, Base32, and Base64 Data Encodings", RFC 3548, July 2003, http://www.rfc-editor.org/rfc/rfc3548.txt

[RFC4122] Leach, P., Mealling, M., and Salz, R., "A Universally Unique Identifier (UUID) URN Namespace", RFC 4122, July 2005, http://www.ietf.org/rfc/rfc4122.txt

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

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

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

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

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

[XMLSCHEMA1] Thompson, H., 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/

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

1.2.2 Informative References

None.

1.3   Protocol Overview (Synopsis)

This protocol enables users to copy **files**, and the associated metadata for each of those files, to one or more locations on a single protocol server or different protocol servers by using **SOAP**, as described in [SOAP1.1], [SOAP1.2/1], and [SOAP1.2/2].

This protocol assumes that each file can be accessed on the protocol server by using a **Uniform Resource Locator (URL)**. For a detailed abstract data model, see the section 3.1.1.

The protocol client sends a request to the protocol server that stores the file, and the associated metadata for the file, at the specified absolute **Internationalized Resource Identifier (IRI)**. If the file exists in the **source location**, the protocol server responds with the content and metadata of the file.

The protocol client then sends the content and metadata of the file, with a collection of **destination locations**, to the **destination server**. The destination server creates or overwrites the files in the destination locations, and sets the metadata of each file to the values that were sent. The destination server then responds with a collection of success or failure messages for each location. The protocol client can send the same content and metadata to multiple destination servers as needed.
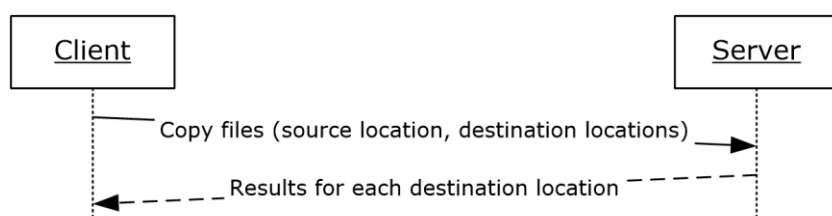

Figure 1: Copying files to a protocol server

This protocol also provides a method for copying a file within the same protocol server without retrieving the content or metadata for the file. The protocol client sends a request to the protocol server with the source file location and the collection of destination locations. The protocol server copies the file and its associated metadata to all of the specified locations. The protocol server then responds with a collection of success or failure messages for each location that was sent.

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].

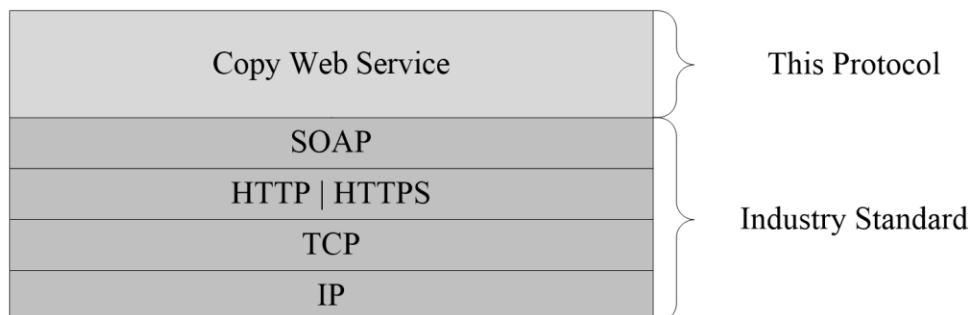The following diagram shows the underlying messaging and transport stack used by the protocol:

Figure 2: This protocol in relation to other protocols

1.5   Prerequisites/Preconditions

This protocol operates against a protocol server that is identified by a URL that is known by protocol clients. The protocol server endpoint is formed by appending "/_vti_bin/copy.asmx" to the URL of the protocol server; for example, http://www.example.com:56737/_vti_bin/copy.asmx.

This protocol assumes that authentication was performed by the underlying protocols.

1.6   Applicability Statement

This protocol is applicable when a protocol server supports storage of IRI-addressable files, SOAP over HTTP or HTTPS, and the destination servers are accessible from the protocol client.

Note the following when using this protocol:

▪   This protocol does not provide a way to control whether overwriting files during a copy operation is allowed.

▪   To copy a file from one protocol server to another, the protocol client downloads the full content of the file as a **base64**-encoded **string**. The amount of network resources and the memory that is used on the protocol client and protocol servers depends on size of the file.

Use this protocol for the following:

▪   Copying files within a single protocol server.

▪   Copying files between protocol servers if those files and the associated metadata for the files contain fewer than 10 megabytes of data.

Consider using different protocols for the following:

▪   Copying only file content between protocol servers because preserving the associated metadata is not required.

▪   Copying files with more than 10 megabytes of content between protocol servers. Although the protocol does not impose a limit on the size of the content to be copied, the practical limit varies based on server resources and network bandwidth.

▪   Copying files when the protocol client needs to control whether overwriting existing files during the copy operation is allowed.

▪   Copying files when the protocol client needs to guarantee that any or all of the associated metadata is copied.

1.7   Versioning and Capability Negotiation

Supported Transports: This protocol uses multiple transports with SOAP as specified 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 to help secure communication with protocol clients. Protocol messages MUST be formatted as specified either in [SOAP1.1] section 4 or in [SOAP1.2/1] section 5. Protocol server faults MUST be returned by using either HTTP **Status-Codes**, as specified in [RFC2616] section 10, or **SOAP faults**, as specified in [SOAP1.1] section 4.4 or [SOAP1.2/1] section 5.4.

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 defined in [XMLSCHEMA1] and [XMLSCHEMA2], and **Web Services Description Language (WSDL)** as defined in [WSDL].

2.2.1 Namespaces

This specification defines and references various **XML namespaces** by using the mechanisms that are specified in [XMLNS]. Although this specification associates a specific **XML namespace prefix** with each XML namespace that is used, the choice of an XML namespace prefix is implementation-specific and is not significant for interoperability.

Prefix	Namespace URI	Reference
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
tns	http://schemas.microsoft.com/sharepoint/soap/	
s1	http://microsoft.com/wsdl/types/	
s	http://www.w3.org/2001/XMLSchema/	[XMLSCHEMA1] [XMLSCHEMA2]
soap12	http://schemas.xmlsoap.org/wsdl/soap12/	[SOAP1.2/1] and [SOAP1.2/2]
(none)	http://schemas.microsoft.com/sharepoint/soap/	
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]

2.2.2 Messages

2.2.2.1 SOAP Fault Message

This protocol enables a protocol server to notify a protocol client with application-level faults by using a SOAP fault response. In a SOAP fault response, the **detail** element contains application-specific error information.

The following schema specifies the structure of the **detail** element in the SOAP fault response that is used by this protocol:

```
    <s:schema xmlns:s="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/sharepoint/soap/">
        <s:complexType name="SOAPFaultDetails">
```

```
            <s:sequence>
                <s:element name="errorstring" type="s:string"/>
                <s:element name="errorcode" type="s:string" minOccurs="0"/>
            </s:sequence>
        </s:complexType>
    </s:schema>
```

errorstring: A description of the error.

errorcode: A hexadecimal representation of a 4-byte result code, which is an operation-specific error code that is defined in the messages subsection for operations that return SOAP faults.

2.2.2.2 SOAP Exception Message

This protocol enables a protocol server to notify a protocol client when a request is not well-formed and cannot be processed using a SOAP exception response.

The following schema specifies the structure of the **detail** element in the SOAP exception response that is used by this protocol:

```
    <s:schema xmlns:s="http://www.w3.org/2001/XMLSchema"
  targetNamespace="http://schemas.microsoft.com/sharepoint/soap/">
        <s:complexType name="SOAPExceptionDetails">
          <s:sequence>
            <s:element name="errorstring" type="s:string"/>
            <s:element name="faultcode" type="s:string" />
            <s:element name="faultstring" type="s:string" />
          </s:sequence>
        </s:complexType>
    </s:schema>
```

errorstring: A description of the error.

faultcode: This MUST be the **string** "soap:Server".

faultstring: This MUST be the **string** "Exception of type 'Microsoft.SharePoint.SoapServer.SoapServerException' was thrown."

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, complex type definitions that are defined in this specification. The complex type definitions that are specific to a particular operation are described with that operation.

Complex Type	Description
DestinationUrlCollection	A collection of destination locations for the copy operation.
CopyResult	The error information and corresponding IRI that are the result of a copy operation to a specific destination.
CopyResultCollection	A collection of copy operation results with one result per destination.

Complex Type	Description
FieldInformation	The value, type, and identity information for a specific metadata field for a file.
FieldInformationCollection	A collection of FieldInformation records for a file.

2.2.4.1 DestinationUrlCollection

The **DestinationUrlCollection** complex type specifies the collection of destination locations for the copy operation.

```
<s:complexType name="DestinationUrlCollection">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
type="s:string" />
  </s:sequence>
</s:complexType>
```

string: A destination location on the destination server. The value of the element MUST be an absolute IRI.

2.2.4.2 CopyResult

The **CopyResult** complex type specifies the result of a copy operation for a single destination location.

```
<s:complexType name="CopyResult">
  <s:attribute name="ErrorCode" type="tns:CopyErrorCode" use="required"/>
  <s:attribute name="ErrorMessage" type="s:string"/>
  <s:attribute name="DestinationUrl" type="s:string" use="required"/>
</s:complexType>
```

ErrorCode: The success or failure status code of the operation for a specific **DestinationUrl**, as defined in section 2.2.5.1.

ErrorMessage: The user-readable message that explains the failure specified by the **ErrorCode** attribute. If the value of **ErrorCode** is "Success", the attribute MUST NOT be present. Otherwise, the **ErrorMessage** attribute MUST be present and the value MUST be a non-empty **Unicode string**.

DestinationUrl: The destination location for which the **CopyResult** element specifies the result of the operation.

2.2.4.3 CopyResultCollection

The **CopyResultCollection** complex type is a collection of results from the copy operation for each destination location that is used in the copy operation. This collection MUST contain exactly one entry for each IRI in the **DestinationUrlCollection** complex type (section 2.2.4.1).

```
<s:complexType name="CopyResultCollection">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="unbounded" name="CopyResult" nillable="true"
type="tns:CopyResult" />
  </s:sequence>
</s:complexType>
```

CopyResult: Specifies the copy status for each destination location, as defined in section 2.2.4.2.

2.2.4.4 FieldInformation

The **FieldInformation** complex type contains the value, type, and identity information for a specific metadata field for a file. Each field has two identifiers, **InternalName** and **Id**.

```
<s:complexType name="FieldInformation">
  <s:attribute name="Type" type="tns:FieldType" use="required"/>
  <s:attribute name="DisplayName" type="s:string" use="required"/>
  <s:attribute name="InternalName" type="s:string" use="required"/>
  <s:attribute name="Id" type="s1:guid" use="required"/>
  <s:attribute name="Value" type="s:string"/>
</s:complexType>
```

Type: The field type that is associated with a file. The value MUST contain one of the values defined in section 2.2.5.2.

DisplayName: The user-readable name of the field. This value MUST be a non-empty Unicode **string** that is fewer than 256 characters.

InternalName: The internal name that identifies the metadata field for a file in the source location. The value MUST be a non-empty **ASCII** **string** that does not contain spaces and is fewer than 256 characters.

Id: The **GUID** that identifies the metadata field for a file in the source location.

Value: The value of the field.

2.2.4.5 FieldInformationCollection

The **FieldInformationCollection** complex type contains a collection of metadata fields and metadata for the file.

The **InternalName** attribute values MUST be unique across all **FieldInformation** elements in the collection. The **Id** attribute values MUST be unique across all **FieldInformation** elements in the collection.

```
<s:complexType name="FieldInformationCollection">
  <s:sequence>
    <s:element name="FieldInformation" type="tns:FieldInformation" minOccurs="0"
maxOccurs="unbounded"/>
  </s:sequence>
</s:complexType>
```

FieldInformation: A single metadata field for a file, as defined in section 2.2.4.4.

2.2.5 Simple Types

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

Simple Type	Description
CopyErrorCode	Enumeration of all possible status codes for the copy operation, including "Success"

Simple Type	Description
FieldType	Enumeration of all metadata field types for a file
guid	GUID value

2.2.5.1 CopyErrorCode

The **CopyErrorCode** simple type specifies the success or failure of the copy operation from the source location to a particular destination location.

```
<s:simpleType name="CopyErrorCode">
  <s:restriction base="s:string">
    <s:enumeration value="Success"/>
    <s:enumeration value="DestinationInvalid"/>
    <s:enumeration value="DestinationMWS"/>
    <s:enumeration value="SourceInvalid"/>
    <s:enumeration value="DestinationCheckedOut"/>
    <s:enumeration value="InvalidUrl"/>
    <s:enumeration value="Unknown"/>
  </s:restriction>
</s:simpleType>
```

The following table lists the allowed values of the **CopyErrorCode** simple type.

Value	Meaning
Success	This value is used when the copy operation succeeds for the specified destination location.
DestinationInvalid	This value is used to indicate an error when the destination location is not the same domain as the source destination or the destination location points to a folder location that is not valid on the destination server.
DestinationMWS	This value is used to indicate a failure to copy the file because the destination location is inside a **Meeting Workspace site**.
SourceInvalid	This value is used to indicate an error when the source location for the copy operation does not reference an existing file in the source location.
DestinationCheckedOut	This value is used to indicate an error when the file on the destination location is **checked out** and cannot be overridden.
InvalidUrl	This value is used to indicate an error when the IRI of a destination location is malformed.
Unknown	This value is used to indicate an error for all other error conditions for a given destination location.

2.2.5.2 FieldType

The **FieldType** simple type specifies the metadata field type of a specified file.

```
<s:simpleType name="FieldType">
  <s:restriction base="s:string">
    <s:enumeration value="Invalid"/>
```

```
         <s:enumeration value="Integer"/>
         <s:enumeration value="Text"/>
         <s:enumeration value="Note"/>
         <s:enumeration value="DateTime"/>
         <s:enumeration value="Counter"/>
         <s:enumeration value="Choice"/>
         <s:enumeration value="Lookup"/>
         <s:enumeration value="Boolean"/>
         <s:enumeration value="Number"/>
         <s:enumeration value="Currency"/>
         <s:enumeration value="URL"/>
         <s:enumeration value="Computed"/>
         <s:enumeration value="Threading"/>
         <s:enumeration value="Guid"/>
         <s:enumeration value="MultiChoice"/>
         <s:enumeration value="GridChoice"/>
         <s:enumeration value="Calculated"/>
         <s:enumeration value="File"/>
         <s:enumeration value="Attachments"/>
         <s:enumeration value="User"/>
         <s:enumeration value="Recurrence"/>
         <s:enumeration value="CrossProjectLink"/>
         <s:enumeration value="ModStat"/>
         <s:enumeration value="AllDayEvent"/>
         <s:enumeration value="Error"/>
      </s:restriction>
   </s:simpleType>
```

This protocol supports the following field types as a subset of the fields that are specified in [MS-WSSTS] section 2.3. Other field types specified in [MS-WSSTS] section 2.3, except the **ContentTypeId**, **LookupMulti** and **UserMulti** field types, MUST map to the Error field type defined by this protocol. The **ContentTypeId** field type SHOULD<1> map to the **Text** field type, the **LookupMulti** field type SHOULD map to the **Lookup** field type, and the **UserMulti** field type SHOULD map to the **User** field type.

Attributes that use **FieldType** MUST use one of the following values:

- Invalid

- Integer

- Text

- Note

- DateTime

- Counter

- Choice

- Lookup

- Boolean

- Number

- Currency

- URL

- Computed

- Threading

- Guid

- MultiChoice

- GridChoice

- Calculated

- File

- Attachments

- User

- Recurrence

- CrossProjectLink

- ModStat

- AllDayEvent

- Error

2.2.5.3 guid

The **guid** simple type specifies a representation of a GUID value, as defined in [RFC4122].

```
<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>
```

2.2.6 Attributes

This specification does not define any common XML schema attributes.

2.2.7 Groups

This specification does not define any common XML schema groups.

2.2.8 Attribute Groups

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

3   Protocol Details

The client side of this protocol is simply a pass-through. That is, no additional timers or other state is required on the 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.

3.1   CopySoap 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 the external behavior is consistent with that described in this document.

The source location of this protocol is an entity that maintains a collection of files and the metadata for each file. The files are identifiable by an absolute IRI on that protocol server.

Metadata for a given file consists of a collection of fields. The fields that are defined for a file can vary based on the location of the file on the protocol server. A field can be read-write or read-only based on the field type, the location of the file on the protocol server, and the **permission** settings for the file.

For files that are created or updated as a result of the **CopyIntoItems** (section 3.1.4.2) or **CopyIntoItemsLocal** (section 3.1.4.3) operations, the protocol server maintains the field for the source location in the metadata. The value of this field is set to the source location that was passed in the operation.<2>

The protocol server provides access control restrictions to files and metadata. If a protocol server implements access control restrictions, all of the operations that are provided by this protocol also take into account those restrictions while accessing the files.

Protocol servers provide the ability to mark some files as checked out. In this case, the copy operations take into account the checked-out status when accessing files at the destination locations.

Some locations on a protocol server can be part of a Meeting Workspace site. Although such locations are valid file locations, attempts to use such a location as a destination for a copy operation will fail. The protocol server can proceed with such an operation. For additional information, see section 3.1.4.2.

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 operations as defined by this specification.

Operation	Description
GetItem	Retrieves a file and metadata for that file from the source location.
CopyIntoItems	Copies a file to a destination server that is different from the source location.
CopyIntoItemsLocal	Copies a file when the destination of the operation is on the same protocol server as the source location.

3.1.4.1 GetItem

The **GetItem** operation retrieves content and metadata for a file that is stored in a source location.

```
<wsdl:operation name="GetItem">
    <wsdl:input message="tns:GetItemSoapIn" />
    <wsdl:output message="tns:GetItemSoapOut" />
</wsdl:operation>
```

The protocol client sends a **GetItemSoapIn** request message (section 3.1.4.1.1.1) and the protocol server responds with a **GetItemSoapOut** response message (section 3.1.4.1.1.2) as follows:

- The protocol client prepares a request with the source location to retrieve content and metadata for the file.

- The protocol server receives the request and provides the content and metadata for the file specified in the source location.

The protocol server returns results based on the following conditions:

- If the source location URL is in an invalid format, the protocol server MUST report a SOAP exception.<3>

- If the source location points to a different server or uses a different HTTP or HTTPS protocol, the protocol server MUST report a SOAP exception.

- If the source location does not point to an existing file on the protocol server, the protocol server MUST omit the **Fields** and **Stream** elements in the **GetItemResponse** element (section 3.1.4.1.2.2).

- If the protocol server implements permission settings for files and the file at the source location cannot be read based on the permission settings for the file, the protocol server MUST report a failure by using HTTP Status-Code 401 Unauthorized, as defined in [RFC2616].

- If the source location points to an existing file in the source location and the file can be read based on the permission settings for the file, the source location MUST return the content and metadata of the file.

3.1.4.1.1 Messages

3.1.4.1.1.1   GetItemSoapIn

The **GetItemSoapIn** message is the request WSDL message for a **GetItem** WSDL operation (section 3.1.4.1).

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

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

The **SOAP body** contains a **GetItem** element (section 3.1.4.1.2.1).

3.1.4.1.1.2   GetItemSoapOut

The **GetItemSoapOut** message is the response WSDL message for a **GetItem** WSDL operation.

The SOAP body contains a **GetItemResponse** element (section 3.1.4.1.2.2).

3.1.4.1.2 Elements

3.1.4.1.2.1   GetItem

The **GetItem** element specifies the request that the protocol client MUST send for the **GetItem** operation (section 3.1.4.1).

```
<s:element name="GetItem">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="Url" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

Url: The location of the file. The value of this element MUST be an absolute IRI for the file in the source location. This value MUST be a valid file location in the source location.

3.1.4.1.2.2   GetItemResponse

The **GetItemResponse** element specifies a protocol server response for the **GetItem** operation (section 3.1.4.1). It contains content and metadata for the requested file.

```
<s:element name="GetItemResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1"
        name="GetItemResult" type="s:unsignedInt" />
      <s:element minOccurs="0" maxOccurs="1"
        name="Fields" type="tns:FieldInformationCollection" />
      <s:element minOccurs="0" maxOccurs="1"
        name="Stream" type="s:base64Binary" />
    </s:sequence>
  </s:complexType>
</s:element>
```

GetItemResult: The result of the operation. The protocol client MUST ignore this value. The protocol server MUST set this value to zero.

Fields: Specifies the collection of **FieldInformation** records that comprise the metadata for the file, as specified in section 2.2.4.5. If the **Fields** element is present, the **Stream** element MUST be present as well.

Stream: Specifies the base64-encoded content of the file. Base64 encoding is specified in [XMLSCHEMA2] section 3.2.16. For more information, see [RFC3548].

3.1.4.1.3 Complex Types

None.

3.1.4.1.4 Simple Types

None.

3.1.4.1.5 Attributes

None.

3.1.4.1.6 Groups

None.

3.1.4.1.7 Attribute Groups

None.

3.1.4.2 CopyIntoItems

The **CopyIntoItems** operation copies a file to the destination server. This operation can be used when the destination server is different from the source location.

```
<wsdl:operation name="CopyIntoItems">
    <wsdl:input message="tns:CopyIntoItemsSoapIn" />
    <wsdl:output message="tns:CopyIntoItemsSoapOut" />
</wsdl:operation>
```

The protocol client sends a **CopyIntoItemsSoapIn** request message (section 3.1.4.2.1.1) and the protocol server responds with a **CopyIntoItemsSoapOut** response message (section 3.1.4.2.1.2) as follows.

The protocol client prepares a request that contains the content of the file, the file metadata as a collection of fields, the source location from which the file is being copied, and the collection of destination locations on the destination server to which the file is to be copied.

The destination server proceeds with the operation by attempting to create new files or to update existing files at all of the locations that are specified in the destination locations collection.

The protocol server MUST report the status of the operation inside the **Results** collection (see section 3.1.4.2.2.2) for each destination location that is passed. The **CopyResult** element in the **Results** collection MUST be in the same order as the items in the destination locations collection. The **DestinationUrl** attribute of the **CopyResult** element (section 2.2.4.2) that corresponds to the destination location MUST be set to the value of the destination location.

If the protocol server detects a **Value** attribute that is not one of the valid values of the **FieldInformation** elements of the **Fields** collection (see section 3.1.4.2.2.1), the protocol server MUST report a failure of the operation by setting the **ErrorCode** attribute to "Unknown" for each **CopyResult** element, and provide a **string** value that denotes the error in the **ErrorMessage** attribute. The validity of the value of a field is derived from the **Type** attribute value and varies according to the base type of the field. Base field types are defined in [MS-WSSTS] section 2.3.1.

- If the value of a field is empty and the base field type is "Integer", "Number", "Boolean", or "DateTime", the **Value** attribute MUST NOT be present.

- If the value of a field is empty and the base field type is something other than "Integer", "Number", "Boolean", or "DateTime", the **Value** attribute MUST either be present and have an empty value, or not be present. A protocol server MUST accept both choices as an empty value.

- If the field is not empty, the **Value** attribute MUST be present and contain a non-empty Unicode **string** that represents the value of the field.

- If the type is "DateTime", the value MUST be a **UTC** value.

The protocol server MUST attempt to copy the file to all destination locations that are specified in the request.

If the file on the protocol server is checked out and cannot be updated, the protocol server MUST report a failure of the copy operation by setting the value of the **ErrorCode** attribute of the corresponding **CopyResult** element to "DestinationCheckedOut", and provide a **string** value that specifies the error in the **ErrorMessage** attribute.

If the destination location is part of a Meeting Workspace site, the protocol server MUST report a failure of the copy operation by setting the value of the **ErrorCode** attribute of the corresponding **CopyResult** element to "DestinationMWS", and provide a **string** value that specifies the error in the **ErrorMessage** attribute. The protocol server MAY proceed with the copy operation to the destination location.

The file that is created or updated as a result of the operation MUST have the same content as specified in the **Stream** element of the **CopyIntoItemsSoapIn** message. If the file cannot be created at the given destination location or the content of the file cannot be set to the value specified in the **Stream** element based on permission settings for example, the protocol server MUST report a failure for this destination location by setting the **ErrorCode** attribute of the corresponding **CopyResult** element to "Unknown" and provide a **string** value that describes the error in the **ErrorMessage** attribute.

The protocol server MUST attempt to update the metadata of the file that is created or updated as a result of the operation to match the metadata that is passed in the **Fields** element of the **CopyIntoItemsSoapIn** message. The field for the source location MUST be set to the value of **SourceUrl** element. Failure to set or update some or all of the fields MUST NOT be treated as a failure to copy the file and MUST be reported as success for the given destination location.<4>

For the **CopyIntoItemsLocal** method, see section 3.1.4.3.

3.1.4.2.1 Messages

3.1.4.2.1.1   CopyIntoItemsSoapIn

The **CopyIntoItemsSoapIn** message is the request WSDL message for a **CopyIntoItems** WSDL operation (section 3.1.4.2).

The SOAP action value of the message is defined as:

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

The SOAP body contains a **CopyIntoItems** element (section 3.1.4.2.2.1).

3.1.4.2.1.2   CopyIntoItemsSoapOut

The **CopyIntoItemsSoapOut** message is the response WSDL message for a **CopyIntoItems** WSDL operation (section 3.1.4.2).

The SOAP body contains a **CopyIntoItemsResponse** element (section 3.1.4.2.2.2).

3.1.4.2.2 Elements

3.1.4.2.2.1   CopyIntoItems

The **CopyIntoItems** element specifies the request that the protocol client MUST send for the **CopyIntoItems** operation (section 3.1.4.2).

```
<s:element name="CopyIntoItems">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="SourceUrl" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="DestinationUrls"
type="tns:DestinationUrlCollection" />
      <s:element minOccurs="0" maxOccurs="1" name="Fields"
type="tns:FieldInformationCollection" />
      <s:element minOccurs="0" maxOccurs="1" name="Stream" type="s:base64Binary" />
    </s:sequence>
  </s:complexType>
</s:element>
```

SourceUrl: The absolute IRI of the file in the source location. Handling of this element is defined in section 3.1.4.2.

DestinationUrls: A collection of locations on the destination server, as defined in section 2.2.4.1. Each entry in the collection MUST be a valid file location on the destination server.

Fields: Specifies a collection of **FieldInformation** records (section 2.2.4.4) that comprise the metadata for the file, as specified in section 2.2.4.5. The protocol server MUST support an empty **Fields** collection by copying the destination stream, and by using implementation-specific default values for the metadata.

Stream: Specifies the base64-encoded content of the file. Base64 encoding is specified in [XMLSCHEMA2] section 3.2.16. For more information, see [RFC3548].

3.1.4.2.2.2   CopyIntoItemsResponse

The **CopyIntoItemsResponse** element specifies a protocol server response for the **CopyIntoItems** operation (section 3.1.4.2). It contains a collection of results for each destination location that was passed to the protocol server in the **CopyIntoItems** request.

```
<s:element name="CopyIntoItemsResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="CopyIntoItemsResult" type="s:unsignedInt"
/>
      <s:element minOccurs="0" maxOccurs="1" name="Results" type="tns:CopyResultCollection"
/>
    </s:sequence>
  </s:complexType>
</s:element>
```

CopyIntoItemsResult: The result of the operation. The protocol client MUST ignore this value. The protocol server MUST set this value to zero ("0").

Results: A collection of **CopyResult** records, as specified in section 2.2.4.3. The collection MUST have exactly one record for each destination location that is passed into the request, as specified in section 3.1.4.2.

3.1.4.2.3 Complex Types

None.

3.1.4.2.4 Simple Types

None.

3.1.4.2.5 Attributes

None.

3.1.4.2.6 Groups

None.

3.1.4.2.7 Attribute Groups

None.

3.1.4.3 CopyIntoItemsLocal

The **CopyIntoItemsLocal** operation copies a file, and the associated metadata, from one location to one or more locations on the same protocol server.

The behavior of this operation is similar to that of the **CopyIntoItems** operation (section 3.1.4.2). Specifications for the **CopyIntoItemsLocal** operation refer to specifications for the **CopyIntoItems** operation where the behavior is the same. Differences between the two operations are specified in this section.

The source location and the destination server refer to the same protocol server for this operation.

```
<wsdl:operation name="CopyIntoItemsLocal">
    <wsdl:input message="tns:CopyIntoItemsLocalSoapIn" />
    <wsdl:output message="tns:CopyIntoItemsLocalSoapOut" />
</wsdl:operation>
```

The protocol client sends a **CopyIntoItemsLocalSoapIn** request message (section 3.1.4.3.1.1) and the protocol server responds with a **CopyIntoItemsLocalSoapOut** response message (section 3.1.4.3.1.2) as follows:

- The protocol client prepares the request with the source location and the collection of destination locations to copy the file to.

- If the source location points to a file whose permission setting does not allow access by the protocol client or the source location does not point to an existing file, then

    - if the destination location does not point to an existing folder or file, the protocol server MUST report a failure by returning the **CopyResult** element (section 2.2.4.2) with the **ErrorCode** attribute set to "Unknown" for this destination location.

    - if the destination location points to an existing file, the protocol server MUST report a failure by returning the **CopyResult** element with the **ErrorCode** attribute set to "SourceInvalid" for this destination location.

- The protocol server retrieves the content and metadata for the file in the source location. The protocol server MUST use the content and metadata of the file for the copy operation.

- The protocol server MUST perform the copy operation for the file and construct a response, as specified in section 3.1.4.2.

- If the source location and the destination location refer to different protocol servers, or if the destination location points to a non-existing folder, the protocol server MUST report a failure by returning the **CopyResult** element with the **ErrorCode** attribute set to "DestinationInvalid".<5>

Unless otherwise specified, the method behavior for the source IRI, content, fields, and destination locations for the **CopyItemsIntoLocal** operation is the same as the **CopyIntoItems** operation, as specified in section 3.1.4.2.<u><6></u>

3.1.4.3.1 Messages

3.1.4.3.1.1   CopyIntoItemsLocalSoapIn

The **CopyIntoItemsLocalSoapIn** message is the request WSDL message for a **CopyIntoItemsLocal** WSDL operation (section 3.1.4.3).

The SOAP action value of the message is defined as:

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

The SOAP body contains a **CopyIntoItemsLocal** element (section 3.1.4.3.2.1).

3.1.4.3.1.2   CopyIntoItemsLocalSoapOut

The **CopyIntoItemsLocalSoapOut** message is the response WSDL message for a **CopyIntoItemsLocal** WSDL operation (section 3.1.4.3).

The SOAP body contains a **CopyIntoItemsLocalResponse** element (section 3.1.4.3.2.2).

3.1.4.3.2 Elements

3.1.4.3.2.1   CopyIntoItemsLocal

The **CopyIntoItemsLocal** element specifies the request that the protocol client MUST send for the **CopyIntoItemsLocal** operation (section 3.1.4.3).

```
<s:element name="CopyIntoItemsLocal">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="0" maxOccurs="1" name="SourceUrl" type="s:string" />
      <s:element minOccurs="0" maxOccurs="1" name="DestinationUrls"
type="tns:DestinationUrlCollection" />
    </s:sequence>
  </s:complexType>
</s:element>
```

SourceUrl: The location of the file in the source location. The value of the element MUST be an absolute IRI, and it MUST be a valid file location in the source location.

DestinationUrls: A collection of destination locations, as specified in section 2.2.4.1. Each entry in the collection MUST be a valid file location on the destination server.

3.1.4.3.2.2   CopyIntoItemsLocalResponse

The **CopyIntoItemsLocalResponse** element specifies a protocol server response for the **CopyIntoItemsLocal** operation (section 3.1.4.3). It contains a collection of results for each destination location that was passed to the protocol server in the **CopyIntoItemsLocal** request.

```
<s:element name="CopyIntoItemsLocalResponse">
  <s:complexType>
    <s:sequence>
```

```
        <s:element minOccurs="1" maxOccurs="1" name="CopyIntoItemsLocalResult"
type="s:unsignedInt" />
        <s:element minOccurs="0" maxOccurs="1" name="Results" type="tns:CopyResultCollection"
/>
    </s:sequence>
  </s:complexType>
</s:element>
```

CopyIntoItemsLocalResult: The result of the **CopyIntoItemsLocal** operation. The client MUST ignore this value. The protocol server MUST set this value to zero ("0").

Results: The **CopyResultCollection**, as specified in section 2.2.4.3. The collection MUST have exactly one entry for each destination location that is passed in the request, as specified in section 3.1.4.2.

3.1.4.3.3 Complex Types

None.

3.1.4.3.4 Simple Types

None.

3.1.4.3.5 Attributes

None.

3.1.4.3.6 Groups

None.

3.1.4.3.7 Attribute Groups

None.

3.1.5   Timer Events

None.

3.1.6   Other Local Events

None.

4 Protocol Examples

4.1 Successful CopyIntoItemsLocal Operation

In this example, the protocol client uses a **CopyIntoItemsLocal** operation (section 3.1.4.3) to copy a file from one location to another location on the same protocol server. The name of the source location is "contoso". The source location is http://contoso/CopySrc/Bitmap2.bmp and the destination location is http://contoso/CopyDst/CopyBitmap2.bmp.

4.1.1 Request

The following is the request message that is sent by the protocol client.

```
<?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>
    <CopyIntoItemsLocal xmlns="http://schemas.microsoft.com/sharepoint/soap/">
        <SourceUrl>http://contoso/CopySrc/Bitmap2.bmp</SourceUrl>

<DestinationUrls><string>http://contoso/CopyDst/CopyBitmap2.bmp</string></DestinationUrls>
    </CopyIntoItemsLocal>
</soap:Body>
</soap:Envelope>
```

4.1.2 Response

The following is the response message that is sent by 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>
    <CopyIntoItemsLocalResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
        <CopyIntoItemsLocalResult>0</CopyIntoItemsLocalResult>
        <Results>
            <CopyResult ErrorCode="Success"
DestinationUrl="http://contoso/Copydst/CopyBitmap2.bmp" />
        </Results>
    </CopyIntoItemsLocalResponse>
</soap:Body>
</soap:Envelope>
```

4.2 Unsuccessful Invocation of GetItem Operation

In this example, the protocol client cannot get information about a file in a specific location. The **GetItem** operation (section 3.1.4.1) is unsuccessful because the file does not exist on the protocol server. Therefore, the response is a **GetItemResponse** element (section 3.1.4.1.2.2) that contains only a **GetItemResult** element.

In this example, the name of the source location is "contoso". The location of the file is http://contoso/Copysrcdl/NonExistent.txt.

4.2.1 Request

The following is the request message that is sent by the protocol client.

```
<?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>
 <GetItem xmlns="http://schemas.microsoft.com/sharepoint/soap/">
 <Url>http://contoso/Copysrcdl/NonExistent.txt</Url>
 </GetItem>
</soap:Body>
</soap:Envelope>
```

4.2.2 Response

The following is the response message that is sent by 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>
 <GetItemResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
  <GetItemResult>0</GetItemResult>
 </GetItemResponse>
</soap:Body>
</soap:Envelope>
```

4.3   Successful GetItem Operation

In this example, the protocol client successfully gets information about a file in a specific location by using a **GetItem** operation (section 3.1.4.1), and copies that file to a location on another protocol server. The name of the source location is "contoso". The source location of the file is http://contoso/Shared%20Documents/sample.txt. The content of the file is: sample text.

4.3.1 Request

A request is sent to the protocol server with the host name "contoso", and the absolute IRI to the file is passed in the **Url** element of the **GetItem** element (section 3.1.4.1.2.1).

```
<?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>
    <GetItem xmlns="http://schemas.microsoft.com/sharepoint/soap/">
    <Url>http://contoso/Shared%20Documents/sample.txt</Url>
    </GetItem>
</soap:Body>
</soap:Envelope>
```

4.3.2 Response

The **GetItemResponse** (section 3.1.4.1.2.2) contains the stream for the file in the **Stream** element, and the collection of the fields of the file in the **Fields** element. The **GetItemsResult** element contains the value zero ("0"), which is ignored by the protocol client.

```
<?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>
  <GetItemResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
   <GetItemResult>0</GetItemResult>
   <Fields>
    <FieldInformation Type="Error" DisplayName="Content Type ID" InternalName="ContentTypeId"
Id="03e45e84-1992-4d42-9116-26f756012634" Value="0x0101001FCFE93C8BB928468474DAB020EFC264" />
    <FieldInformation Type="Note" DisplayName="Approver Comments"
InternalName=" ModerationComments" Id="34ad21eb-75bd-4544-8c73-0e08330291fe" />
    <FieldInformation Type="File" DisplayName="Name" InternalName="FileLeafRef" Id="8553196d-
ec8d-4564-9861-3dbe931050c8" Value="sample.txt" />
    <FieldInformation Type="DateTime" DisplayName="Created" InternalName="Created"
Id="8c06beca-0777-48f7-91c7-6da68bc07b69" Value="2/25/2008 3:21:18 PM" />
    <FieldInformation Type="User" DisplayName="Created By" InternalName="Author"
Id="1df5e554-ec7e-46a6-901d-d85a3881cb18" Value="84;#Syed Abbas" />
    <FieldInformation Type="DateTime" DisplayName="Modified" InternalName="Modified"
Id="28cf69c5-fa48-462a-b5cd-27b6f9d2bd5f" Value="2/25/2008 3:21:25 PM" />
    <FieldInformation Type="User" DisplayName="Modified By" InternalName="Editor"
Id="d31655d1-1d5b-4511-95a1-7a09e9b75bf2" Value="84;#Syed Abbas" />
    <FieldInformation Type="Text" DisplayName="Copy Source" InternalName=" CopySource"
Id="6b4e226d-3d88-4a36-808d-a129bf52bccf" />
    <FieldInformation Type="DateTime" DisplayName="Modified"
InternalName="Last_x0020_Modified" Id="173f76c8-aebd-446a-9bc9-769a2bd2c18f" Value="2/25/2008
3:21:25 PM" />
   </Fields>
   <Stream>c2FtcGxlIHRleHQuDQo=</Stream>
  </GetItemResponse>
 </soap:Body>
</soap:Envelope>
```

4.4 CopyIntoItems Operation

In this example, the protocol client uses the **CopyIntoItems** operation (section 3.1.4.2) to copy a file, which was retrieved in the Successful GetItem Operation example in section 4.3, to another protocol server.

The name of the source location is "contoso". The name of the destination server is "contoso2". The location of the source file is http://contoso/Shared%20Documents/sample.txt. The values for the **Fields** element derive from the response outlined in the Successful GetItem Operation example in section 4.3. The content of the file is: sample text.

4.4.1 Request

The following is the request message that is sent by the protocol client.

```
<?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>
  <CopyIntoItems xmlns="http://schemas.microsoft.com/sharepoint/soap/">
   <SourceUrl>http://contoso/Shared%20Documents/sample.txt</SourceUrl>
   <DestinationUrls>
    <string>http://contoso2/Shared%20Documents/sample1.txt</string>
```

```
      <string>http://contoso2/Shared%20Documents/sample2.txt</string>
      <string>http://contoso2/NonExistent/sample.txt</string>
      <string>http://contoso2/mws/Document%20Library/sample.txt</string>
    </DestinationUrls>
    <Fields>
      <FieldInformation Type="Error" DisplayName="Content Type ID" InternalName="ContentTypeId"
Id="03e45e84-1992-4d42-9116-26f756012634" Value="0x0101001FCFE93C8BB928468474DAB020EFC264" />
      <FieldInformation Type="Note" DisplayName="Approver Comments"
InternalName="_ModerationComments" Id="34ad21eb-75bd-4544-8c73-0e08330291fe" />
      <FieldInformation Type="File" DisplayName="Name" InternalName="FileLeafRef" Id="8553196d-
ec8d-4564-9861-3dbe931050c8" Value="sample.txt" />
      <FieldInformation Type="DateTime" DisplayName="Created" InternalName="Created"
Id="8c06beca-0777-48f7-91c7-6da68bc07b69" Value="2/25/2008 3:21:18 PM" />
      <FieldInformation Type="User" DisplayName="Created By" InternalName="Author"
Id="1df5e554-ec7e-46a6-901d-d85a3881cb18" Value="84;#Syed Abbas" />
      <FieldInformation Type="DateTime" DisplayName="Modified" InternalName="Modified"
Id="28cf69c5-fa48-462a-b5cd-27b6f9d2bd5f" Value="2/25/2008 3:21:25 PM" />
      <FieldInformation Type="User" DisplayName="Modified By" InternalName="Editor"
Id="d31655d1-1d5b-4511-95a1-7a09e9b75bf2" Value="84;#Syed Abbas" />
      <FieldInformation Type="Text" DisplayName="Copy Source" InternalName="_CopySource"
Id="6b4e226d-3d88-4a36-808d-a129bf52bccf" />
      <FieldInformation Type="DateTime" DisplayName="Modified"
InternalName="Last x0020 Modified" Id="173f76c8-aebd-446a-9bc9-769a2bd2c18f" Value="2/25/2008
3:21:25 PM" />
    </Fields>
    <Stream>c2FtcGxlIHRleHQuDQo=</Stream>
  </CopyIntoItems>
 </soap:Body>
</soap:Envelope>
```

4.4.2 Response

The protocol server sends a **CopyResult** response (section 2.2.4.2) indicating success for the
following destination locations:

- http://contoso2/Shared%20Documents/sample1.txt

- http://contoso2/Shared%20Documents/sample2.txt

For the destination URL http://contoso2/NonExistent/sample.txt, the operation generates the following
error:

```
    Unknown failure: Object reference not set to an instance of an object.
```

For the destination URL http://contoso2/mws/Document%20Library/sample.txt, the operation
generates the following error:

```
    DestinationMWS failure: The destination is in a Meeting Workspace site, which does not
    support this action.

    <?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>
      <CopyIntoItemsResponse xmlns="http://schemas.microsoft.com/sharepoint/soap/">
       <CopyIntoItemsResult>0</CopyIntoItemsResult>
       <Results>
        <CopyResult ErrorCode="Success"
    DestinationUrl="http://contoso2/Shared%20Documents/sample1.txt" />
        <CopyResult ErrorCode="Success"
    DestinationUrl="http://contoso2/Shared%20Documents/sample2.txt" />
```

```
        <CopyResult ErrorCode="Unknown" ErrorMessage="Object reference not set to an instance of
an object." DestinationUrl="http://contoso2/NonExistent/sample.txt" />
        <CopyResult ErrorCode="DestinationMWS" ErrorMessage="The destination is in a Meeting
Workspace site, which does not support this action."
DestinationUrl="http://contoso2/mws/Document%20Library/sample.txt" />
      </Results>
    </CopyIntoItemsResponse>
  </soap:Body>
</soap:Envelope>
```

5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.

6   Appendix A: Full WSDL

For ease of implementation, the full WSDL is provided in this section:

```
<?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://microsoft.com/wsdl/types/" />
      <s:element name="CopyIntoItemsLocal">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="1" name="SourceUrl" type="s:string" />
            <s:element minOccurs="0" maxOccurs="1" name="DestinationUrls"
type="tns:DestinationUrlCollection" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:complexType name="DestinationUrlCollection">
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
type="s:string" />
        </s:sequence>
      </s:complexType>
      <s:element name="CopyIntoItemsLocalResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="CopyIntoItemsLocalResult"
type="s:unsignedInt" />
            <s:element minOccurs="0" maxOccurs="1" name="Results"
type="tns:CopyResultCollection" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:complexType name="CopyResultCollection">
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="unbounded" name="CopyResult" nillable="true"
type="tns:CopyResult" />
        </s:sequence>
      </s:complexType>
      <s:complexType name="CopyResult">
        <s:attribute name="ErrorCode" type="tns:CopyErrorCode" use="required" />
        <s:attribute name="ErrorMessage" type="s:string" />
        <s:attribute name="DestinationUrl" type="s:string" use="required"  />
      </s:complexType>
      <s:simpleType name="CopyErrorCode">
        <s:restriction base="s:string">
          <s:enumeration value="Success" />
          <s:enumeration value="DestinationInvalid" />
          <s:enumeration value="DestinationMWS" />
          <s:enumeration value="SourceInvalid" />
          <s:enumeration value="DestinationCheckedOut" />
          <s:enumeration value="InvalidUrl" />
          <s:enumeration value="Unknown" />
        </s:restriction>
      </s:simpleType>
      <s:element name="CopyIntoItems">
```

```
            <s:complexType>
              <s:sequence>
                <s:element minOccurs="0" maxOccurs="1" name="SourceUrl" type="s:string" />
                <s:element minOccurs="0" maxOccurs="1" name="DestinationUrls"
type="tns:DestinationUrlCollection" />
                <s:element minOccurs="0" maxOccurs="1" name="Fields"
type="tns:FieldInformationCollection" />
                <s:element minOccurs="0" maxOccurs="1" name="Stream" type="s:base64Binary" />
              </s:sequence>
            </s:complexType>
          </s:element>
          <s:complexType name="FieldInformationCollection">
            <s:sequence>
              <s:element minOccurs="0" maxOccurs="unbounded" name="FieldInformation"
type="tns:FieldInformation" />
            </s:sequence>
          </s:complexType>
          <s:complexType name="FieldInformation">
            <s:attribute name="Type" type="tns:FieldType" use="required" />
            <s:attribute name="DisplayName" type="s:string" use="required"/>
            <s:attribute name="InternalName" type="s:string" use="required"/>
            <s:attribute name="Id" type="s1:guid" use="required"/>
            <s:attribute name="Value" type="s:string" />
          </s:complexType>
          <s:simpleType name="FieldType">
            <s:restriction base="s:string">
              <s:enumeration value="Invalid" />
              <s:enumeration value="Integer" />
              <s:enumeration value="Text" />
              <s:enumeration value="Note" />
              <s:enumeration value="DateTime" />
              <s:enumeration value="Counter" />
              <s:enumeration value="Choice" />
              <s:enumeration value="Lookup" />
              <s:enumeration value="Boolean" />
              <s:enumeration value="Number" />
              <s:enumeration value="Currency" />
              <s:enumeration value="URL" />
              <s:enumeration value="Computed" />
              <s:enumeration value="Threading" />
              <s:enumeration value="Guid" />
              <s:enumeration value="MultiChoice" />
              <s:enumeration value="GridChoice" />
              <s:enumeration value="Calculated" />
              <s:enumeration value="File" />
              <s:enumeration value="Attachments" />
              <s:enumeration value="User" />
              <s:enumeration value="Recurrence" />
              <s:enumeration value="CrossProjectLink" />
              <s:enumeration value="ModStat" />
              <s:enumeration value="AllDayEvent" />
              <s:enumeration value="Error" />
            </s:restriction>
          </s:simpleType>
          <s:element name="CopyIntoItemsResponse">
            <s:complexType>
              <s:sequence>
                <s:element minOccurs="1" maxOccurs="1" name="CopyIntoItemsResult"
type="s:unsignedInt" />
                <s:element minOccurs="0" maxOccurs="1" name="Results"
type="tns:CopyResultCollection" />
              </s:sequence>
            </s:complexType>
          </s:element>
          <s:element name="GetItem">
            <s:complexType>
              <s:sequence>
                <s:element minOccurs="0" maxOccurs="1" name="Url" type="s:string" />
              </s:sequence>
```

```
          </s:complexType>
        </s:element>
        <s:element name="GetItemResponse">
          <s:complexType>
            <s:sequence>
              <s:element minOccurs="1" maxOccurs="1" name="GetItemResult" type="s:unsignedInt"
/>
              <s:element minOccurs="0" maxOccurs="1" name="Fields"
type="tns:FieldInformationCollection" />
              <s:element minOccurs="0" maxOccurs="1" name="Stream" type="s:base64Binary" />
            </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="CopyIntoItemsLocalSoapIn">
    <wsdl:part name="parameters" element="tns:CopyIntoItemsLocal" />
  </wsdl:message>
  <wsdl:message name="CopyIntoItemsLocalSoapOut">
    <wsdl:part name="parameters" element="tns:CopyIntoItemsLocalResponse" />
  </wsdl:message>
  <wsdl:message name="CopyIntoItemsSoapIn">
    <wsdl:part name="parameters" element="tns:CopyIntoItems" />
  </wsdl:message>
  <wsdl:message name="CopyIntoItemsSoapOut">
    <wsdl:part name="parameters" element="tns:CopyIntoItemsResponse" />
  </wsdl:message>
  <wsdl:message name="GetItemSoapIn">
    <wsdl:part name="parameters" element="tns:GetItem" />
  </wsdl:message>
  <wsdl:message name="GetItemSoapOut">
    <wsdl:part name="parameters" element="tns:GetItemResponse" />
  </wsdl:message>
  <wsdl:portType name="CopySoap">
    <wsdl:operation name="CopyIntoItemsLocal">
      <wsdl:input message="tns:CopyIntoItemsLocalSoapIn" />
      <wsdl:output message="tns:CopyIntoItemsLocalSoapOut" />
    </wsdl:operation>
    <wsdl:operation name="CopyIntoItems">
      <wsdl:input message="tns:CopyIntoItemsSoapIn" />
      <wsdl:output message="tns:CopyIntoItemsSoapOut" />
    </wsdl:operation>
    <wsdl:operation name="GetItem">
      <wsdl:input message="tns:GetItemSoapIn" />
      <wsdl:output message="tns:GetItemSoapOut" />
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="CopySoap" type="tns:CopySoap">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="CopyIntoItemsLocal">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/CopyIntoItemsLocal" style="document"
/>
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
```

```
    <wsdl:operation name="CopyIntoItems">
      <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/CopyIntoItems"
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetItem">
      <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/GetItem"
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="CopySoap12" type="tns:CopySoap">
    <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="CopyIntoItemsLocal">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/CopyIntoItemsLocal" style="document"
/>
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="CopyIntoItems">
      <soap12:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/CopyIntoItems" style="document" />
      <wsdl:input>
        <soap12:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetItem">
      <soap12:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/GetItem"
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 released service packs.

- Windows SharePoint Services 3.0

- Microsoft SharePoint Foundation 2010

- Microsoft SharePoint Foundation 2013

- The 2007 Microsoft Office system

- Microsoft Office 2010 suites

- Microsoft Office 2013

- Windows 8.1

- Microsoft Office 2016

- Windows 10 operating system

- Microsoft SharePoint Server 2016

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

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

<1> Section 2.2.5.2: 1 The **ContentTypeID** field type in Windows SharePoint Services 3.0 and SharePoint Foundation 2010 map to the error field type. The **ContentTypeID** field type in SharePoint Foundation 2010 SP1 maps to the **Text** field type.

<2> Section 3.1.1: Windows SharePoint Services 3.0, SharePoint Foundation 2010 and SharePoint Foundation 2013 maintain a field with the internal name **_CopySource** for all of the files that are stored in a list. The initial value of this field is empty. During the **CopyIntoItems** or **CopyIntoItemsLocal** operations the protocol server sets the value of this field to the source location that is passed in the operation for each file that is created or updated during the operation.

<3> Section 3.1.4.1: Windows SharePoint Services 3.0 does not return a SOAP exception.

<4> Section 3.1.4.2: Windows SharePoint Services 3.0, SharePoint Foundation 2010 and SharePoint Foundation 2013 set as many fields of the file, identified by **Display Name**, in the destination location as possible based on the collection of fields that are passed into the **CopyIntoItems** operation or are available on the file at the source location for a **CopyIntoItemsLocal** operation. Copy operations ignore the following fields:

- Fields that are computed based on the destination location

- Read-only fields on the destination location

- Fields that are not allowed in the metadata of the files on the destination location

<5> Section 3.1.4.3: The server returns empty results when the source location and the destination location refer to different protocol servers.

<6> Section 3.1.4.3: When a destination location in the **CopyIntoItemsLocal** operation (section 3.1.4.3) is a malformed IRI, Windows SharePoint Services 3.0 and SharePoint Foundation 2010 returns an **ErrorCode** of "Unknown" with text indicating that the URL is not valid instead of returning an **ErrorCode** of "InvalidUrl".

8   Change Tracking

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

9 Index