[MS-CDEPLOY]:
Content Deployment Remote Import Web Service Protocol Specification

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.

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

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

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

- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.

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

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

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

Revision Summary

Date	Revision History	Revision Class	Comments
04/04/2008	0.1		Initial Availability
04/25/2008	0.2	Editorial	Revised and edited the technical content
06/27/2008	1.0	Major	Revised and edited the technical content
10/06/2008	1.01	Editorial	Revised and edited the technical content
12/12/2008	1.02	Editorial	Revised and edited the technical content
07/13/2009	1.03	Major	Revised and edited the technical content
08/28/2009	1.04	Editorial	Revised and edited the technical content
11/06/2009	1.05	Editorial	Revised and edited the technical content
02/19/2010	2.0	Minor	Updated the technical content
03/31/2010	2.01	Editorial	Revised and edited the technical content
04/30/2010	2.02	Editorial	Revised and edited the technical content
06/07/2010	2.03	Editorial	Revised and edited the technical content
06/29/2010	2.04	Editorial	Changed language and formatting in the technical content.
07/23/2010	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
09/27/2010	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	2.04	No change	No changes to the meaning, language, or formatting of the technical content.
03/18/2011	3.0	Major	Significantly changed the technical content.
06/10/2011	3.0	No change	No changes to the meaning, language, or formatting of the technical content.
01/20/2012	3.1	Minor	Clarified the meaning of the technical content.
04/11/2012	3.1	No change	No changes to the meaning, language, or formatting of the technical content.
07/16/2012	3.1	No change	No changes to the meaning, language, or formatting of the technical content.
09/12/2012	3.1	No change	No changes to the meaning, language, or formatting of

Date	Revision History	Revision Class	Comments
			the technical content.
10/08/2012	3.1.1	Editorial	Changed language and formatting in the technical content.

Table of Contents

1   Introduction

This document specifies the Content Deployment Remote Import Web Service Protocol, which enables a protocol client to remotely manage the import of a set of content, previously exported from an export server, to a protocol server.

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

1.1   Glossary

The following terms are defined in [MS-GLOS]:

GUID
Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)

The following terms are defined in [MS-OFCGLOS]:

Central Administration site
deployment package
editor
event receiver
farm
HTTP POST
Simple Object Access Protocol (SOAP)
site collection
site-relative URL
SOAP action
SOAP body
SOAP fault
SOAP message
timer job
Uniform Resource Locator (URL)
web application
Web Services Description Language (WSDL)
XML namespace
XML namespace prefix
XML schema
XML schema definition (XSD)

The following terms are specific to this document:

content deployment: The act of exporting content from a source system and importing it to a destination system.

export server: A server that serves as the source of a data export operation.

import job: A timer job that is used to import data from a content migration package to a remote server.

operator account: The account of the user who is managing the import process for a deployment package.

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

1.2 References

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

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information. Please check the archive site, http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624, as an additional source.

[MS-NRTP] Microsoft Corporation, ".NET Remoting: Core Protocol Specification".

[MS-PRIMEPF] Microsoft Corporation, "Deployment Package Format Specification".

[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.ietf.org/rfc/rfc2616.txt

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

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

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

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

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

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

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

1.2.2 Informative References

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

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

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

1.3 Overview

This protocol enables a protocol client to remotely manage the import of a **deployment package (1)** to a protocol server. This protocol consists of nine stateless operations. These operations enable a protocol client to create an **import job**, run the import job, monitor the status of the import job, cancel the import job, and delete the import job or the log files created by the import job on the remote import server. The operations also enable the protocol client to gather information about the remote import server, such as the **URL** of the **Central Administration site**, the names and the URLs of all **Web applications (1)**, and the names of all **site collections** within a specified Web application (1).

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:
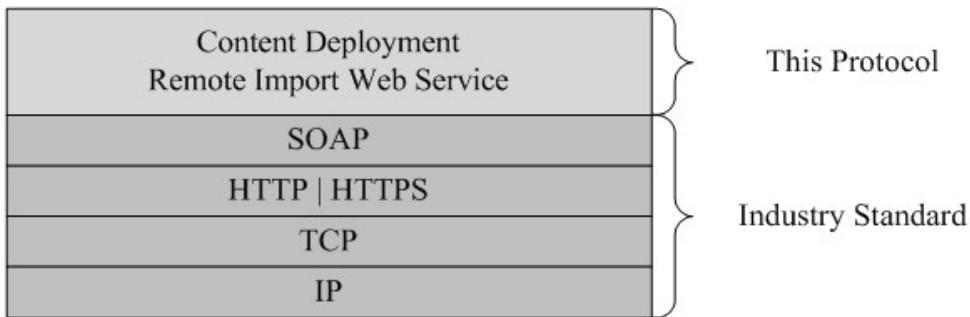
Figure 1: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against the central administration site of a remote import server with a URL that is known to the protocol client. The protocol server endpoint is formed by appending "_vti_adm/ContentDeploymentRemoteImport.asmx" to the URL of the central administration site of the remote import server. For example, if the URL of the central administration site of the remote import server were http://www.contoso.com:8080, the protocol server endpoint would be http://www.contoso.com:8080/_vti_adm/ContentDeploymentRemoteImport.asmx.

This protocol assumes that authentication has been performed by the underlying protocols.

The remote import server has been configured to accept incoming **content deployment** import jobs.

1.6 Applicability Statement

This protocol is not applicable to the scenario of importing content to a site collection while users are editing content in that site collection. The protocol is also not applicable to the scenario of concurrent imports of multiple deployment packages (1) to the same site collection. When used to

query for the status of an import job, this protocol is intended for querying no more frequently than once every 10 seconds.

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 specified in Section 2.1.

- **Localization:** This protocol includes text strings in various messages. Localization considerations for such strings are specified in section 2.2 and section 3.1.4.

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] (Section 4, SOAP Envelope) or in [SOAP1.2/1] SOAP Message Construct section 5. Protocol server faults MUST be returned either using HTTP status codes as specified in [RFC2616] Status Code Definitions section 10 or using **SOAP faults** as specified either in [SOAP1.1] SOAP Fault section 4.4 or in [SOAP1.2/1] SOAP Fault 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 specified in [XMLSCHEMA1] and [XMLSCHEMA2], and **WSDL**, as specified in [WSDL].

2.2.1 Namespaces

This specification defines and references various **XML namespaces** using the mechanisms specified in [XMLNS]. Although this specification associates a specific **XML namespace prefix** to 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
s	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1] [XMLSCHEMA2]
soap12	http://schemas.xmlsoap.org/wsdl/soap12/	[SOAP1.2/1] [SOAP1.2/2]
tns	http://schemas.microsoft.com/sharepoint/soap/	
s1	http://microsoft.com/wsdl/types/	
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
(none)	http://schemas.microsoft.com/sharepoint/soap/	
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]

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

This specification does not define any common XML schema complex type definitions.

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
guid	A globally unique identifier (GUID).

2.2.5.1 guid

A **guid** simple type specifies a GUID. Its structure is specified by the following **XML schema definition (XSD)**:

```
<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 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

Except where specified, the protocol client SHOULD interpret HTTP status codes returned by the protocol server as specified in [RFC2616] Status Code Definitions section 10.

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

This protocol allows protocol servers to perform implementation-specific authorization checks and notify the protocol client of authorization faults either using HTTP status codes or using SOAP faults as specified previously in this section.

3.1   ContentDeploymentRemoteImportSoap Server Details

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.

All operations consist of a basic SOAP request-response pair and the protocol server treats each request as an independent transaction, unrelated to any previous request.

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 specification does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this specification.

The protocol server MUST maintain a mapping of **GUIDs** to import jobs such that, for any GUID, at most one import job is found.

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.

Operations	Description
CancelJob	Cancels the specified import job.
CreateJob	Creates a new import job.
DeleteImportLogs	Deletes log files created by import job(s) associated with the specified content deployment job.

Operations	Description
DeleteJob	Deletes the specified import job.
GetJobStatus	Returns the status of the specified import job.
GetRemoteAdminServerUrl	Returns the URL of the central administration site on the remote import server.
GetSiteCollectionNames	Returns the names of all site collections within the specified Web application (1).
GetVirtualServersInformation	Returns the names and URLs of all Web applications (1) within the current **farm**.
RunJob	Runs the specified import job.

3.1.4.1   CancelJob

This operation cancels the specified import job, as follows.

```
<wsdl:operation name="CancelJob">
    <wsdl:input message="tns:CancelJobSoapIn" />
    <wsdl:output message="tns:CancelJobSoapOut" />
</wsdl:operation>
```

A protocol client initiates the operation by sending a **CancelJobSoapIn** message to the protocol server, formatted as specified in section 3.1.4.1.1.1.

For this operation the protocol server MUST take no action.

The protocol server MUST respond with a **CancelJobSoapOut** response message, formatted as specified in section 3.1.4.1.1.2.

3.1.4.1.1   Messages

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

Message	Description
CancelJobSoapIn	A request to initiate a **CancelJob** operation on the protocol server.
CancelJobSoapOut	A response from the protocol server at completion of the **CancelJob** operation.

3.1.4.1.1.1   CancelJobSoapIn

This message is sent from the protocol client to the protocol server to initiate a **CancelJob** operation.

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

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

The **SOAP body** MUST contain a **CancelJob** element.

3.1.4.1.1.2 CancelJobSoapOut

This message is sent from the protocol server to the protocol client when the **CancelJob** operation is finished.

The SOAP action value of the message is specified as follows:

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

The SOAP body MUST contain a **CancelJobResponse** element.

3.1.4.1.2 Elements

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

Element	Description
CancelJob	Specifies the body of the **CancelJobSoapIn** message.
CancelJobResponse	Specifies the body of the **CancelJobSoapOut** message.

3.1.4.1.2.1 CancelJob

This element forms the body of the **CancelJobSoapIn** message. It MUST adhere to the following XSD:

```
<s:element name="CancelJob">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="jobId" type="s1:guid" />
    </s:sequence>
  </s:complexType>
</s:element>
```

jobId: The identifier of the import job.

3.1.4.1.2.2 CancelJobResponse

This element forms the body of the **CancelJobSoapOut** message. It MUST adhere to the following XSD:

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

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 CreateJob

This operation creates a new import job, as follows.

```
<wsdl:operation name="CreateJob">
    <wsdl:input message="tns:CreateJobSoapIn" />
    <wsdl:output message="tns:CreateJobSoapOut" />
</wsdl:operation>
```

The protocol client initiates the operation by sending a **CreateJobSoapIn** request message to the server, formatted as specified in section 3.1.4.2.1.1.

If an import job already exists with the specified **sourceId**, the protocol server MUST delete the existing import job. The protocol server SHOULD<1> create a new import job with the specified parameter values.

When the operation is finished, the protocol server MUST respond with a **CreateJobSoapOut** response message, formatted as specified in section 3.1.4.2.1.2.

3.1.4.2.1 Messages

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

Message	Description
CreateJobSoapIn	A request to initiate a **CreateJob** operation on the protocol server.
CreateJobSoapOut	A response from the protocol server at completion of the **CreateJob** operation.

3.1.4.2.1.1 CreateJobSoapIn

This message is sent from the protocol client to the protocol server to initiate a **CreateJob** operation.

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

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

The SOAP body MUST contain a **CreateJob** element.

3.1.4.2.1.2 CreateJobSoapOut

This message is sent from the protocol server to the protocol client when the **CreateJob** operation is finished.

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

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

The SOAP body MUST contain a **CreateJobResponse** element.

3.1.4.2.2 Elements

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

Element	Description
CreateJob	Specifies the body of the **CreateJobSoapIn** message.
CreateJobResponse	Specifies the body of the **CreateJobSoapOut** message.

3.1.4.2.2.1 CreateJob

This element forms the body of the **CreateJobSoapIn** message. It MUST adhere to the following XSD<2>:

```
<s:element name="CreateJob">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="description" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="sourceId" type="s1:guid" />
      <s:element minOccurs="1" maxOccurs="1" name="destinationServerUrl" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="destinationSiteCollection" type="s:string"
/>
      <s:element minOccurs="1" maxOccurs="1" name="includeSecurity"
type="tns:SPIncludeSecurity" />
      <s:element minOccurs="1" maxOccurs="1" name="includeUserInfoDateTime"
type="tns:SPImportUserInfoDateTimeOption" />
      <s:element minOccurs="1" maxOccurs="1" name="enableEventReceivers" type="s:boolean" />
      <s:element minOccurs="1" maxOccurs="1" name="fileCompression" type="s:boolean" />
      </s:sequence>
  </s:complexType>
</s:element>
```

description: The description of the new import job**.** It MUST be less than 256 characters in length.

sourceId: The identifier of the content deployment job that associates it with the import job that is to be created.

destinationServerUrl: The URL of the remote import server for the new import job. It MUST be less than 255 characters in length and MUST be the URL of an existing Web application (1).

destinationSiteCollection: The **site-relative URL** of the destination site collection for the new import job. It MUST be less than 256 characters in length and MUST be the URL of an existing site collection within the Web application (1) specified by **destinationServerUrl**.

includeSecurity: The user and group information to import. This MUST be set to the same value as was specified for **IncludeSecurity** in the ExportSettings.xml file (specified in [MS-PRIMEPF] section 2.2) when the deployment package (1) was created.

includeUserInfoDateTime: The level of user and creation date information to retain during the import. The default value of this element is None.

enableEventReceivers: Whether **event receivers** are notified while content is being imported.

fileCompression: Whether the deployment package (1) is in a compressed format.

3.1.4.2.2   CreateJobResponse

This element forms the body of the **CreateJobSoapOut** message. It MUST adhere to the following XSD:

```
<s:element name="CreateJobResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="CreateJobResult" type="s1:guid" />
    </s:sequence>
  </s:complexType>
</s:element>
```

CreateJobResult: The identifier of the newly created import job.

3.1.4.2.3   Complex Types

None.

3.1.4.2.4   Simple Types

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

Simple type	Description
SPIncludeSecurity	Specifies user and group information included for imported content.
SPImportUserInfoDateTimeOption	Specifies level of user and creation date retained for imported content.

3.1.4.2.4.1   SPIncludeSecurity

The **SPIncludeSecurity** simple type specifies the user and group information to include when the content is imported. It is specified in [MS-PRIMEPF] section 2.2.4.4.

3.1.4.2.4.2 SPImportUserInfoDateTimeOption

The **SPImportUserInfoDateTimeOption** simple type specifies the level of user and creation date information to retain when the content is imported. It is specified by the following XSD:

```
<s:simpleType name="SPImportUserInfoDateTimeOption">
  <s:restriction base="s:string">
    <s:enumeration value="None"/>
    <s:enumeration value="ReplaceUserWithSystemAccount"/>
    <s:enumeration value="ImportAll"/>
  </s:restriction>
</s:simpleType>
```

The following table defines the valid values.

Value	Meaning
None	Sets the creation date/time values for all content to the date/time of the import, replaces all authors and **editors** with the **operator account**, and replaces all references to users with the empty string.
ReplaceUserWithSystemAccount	Retains the original creation date/time values for all content, but replaces all authors and editors with the system account, and replaces all references to users with the empty string.
ImportAll	Retains all original creation date/time values, all original author and editor information, and all references to users.

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 DeleteImportLogs

This operation deletes log files created by import job(s) associated with the specified content deployment job, as follows.<3>

```
<wsdl:operation name="DeleteImportLogs">
  <wsdl:input message="tns:DeleteImportLogsSoapIn" />
  <wsdl:output message="tns:DeleteImportLogsSoapOut" />
</wsdl:operation>
```

A protocol client initiates the operation by sending a **DeleteImportLogsSoapIn** message to the protocol server, formatted as specified in section 3.1.4.3.1.1.

The protocol server MUST delete any existing log files created by import job(s) associated with the specified content deployment job. If no such log files exist, the protocol server MUST take no action.

When the operation is finished, the protocol server MUST respond with a **DeleteImportLogsSoapOut** response message, formatted as specified in section 3.1.4.3.1.2.

3.1.4.3.1 Messages

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

Message	Description
DeleteImportLogsSoapIn	A request to initiate a **DeleteImportLogs** operation on the protocol server.
DeleteImportLogsSoapOut	A response from the protocol server at completion of the **DeleteImportLogs** operation.

3.1.4.3.1.1 DeleteImportLogsSoapIn

This message is sent from the protocol client to the protocol server to initiate a **DeleteImportLogs** operation.

The SOAP action value of the message is specified as follows:

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

The SOAP body MUST contain a **DeleteImportLogs** element.

3.1.4.3.1.2 DeleteImportLogsSoapOut

This message is sent from the protocol server to the protocol client when the **DeleteImportLogs** operation is finished.

The SOAP action value of the message is specified as follows:

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

The SOAP body MUST contain a **DeleteImportLogsResponse** element.

3.1.4.3.2 Elements

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

Element	Description
DeleteImportLogs	Specifies the body of the **DeleteImportLogsSoapIn** message.
DeleteImportLogsReponse	Specifies the body of the **DeleteImportLogsSoapOut** message.

3.1.4.3.2.1 DeleteImportLogs

This element forms the body of the **DeleteImportLogsSoapIn** message. It MUST adhere to the following XSD:

```
<s:element name="DeleteImportLogs">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="sourceJobId" type="s1:guid" />
    </s:sequence>
  </s:complexType>
</s:element>
```

sourceJobId: The identifier of the content deployment job.

3.1.4.3.2.2 DeleteImportLogsResponse

This element forms the body of the **DeleteImportLogsResponse** message. It MUST adhere to the following XSD:

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

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.

3.1.4.4 DeleteJob

This operation deletes the specified import job, as follows.

```
<wsdl:operation name="DeleteJob">
    <wsdl:input message="tns:DeleteJobSoapIn" />
    <wsdl:output message="tns:DeleteJobSoapOut" />
</wsdl:operation>
```

The protocol client initiates the operation by sending a **DeleteJobSoapIn** request message to the protocol server, formatted as specified in section 3.1.4.4.1.1.

If the specified import job exists the protocol server MUST delete the import job. Otherwise, the protocol server MUST take no action.

When the operation is finished, the protocol server MUST respond with a **DeleteJobSoapOut** response message, formatted as specified in section 3.1.4.4.1.2.

3.1.4.4.1 Messages

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

Message	Description
DeleteJobSoapIn	A request to initiate a **DeleteJob** operation on the protocol server.
DeleteJobSoapOut	A response from the protocol server at completion of the **DeleteJob** operation.

3.1.4.4.1.1 DeleteJobSoapIn

This message is sent from the protocol client to the protocol server to initiate a **DeleteJob** operation.

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

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

The SOAP body MUST contain a **DeleteJob** element.

3.1.4.4.1.2 DeleteJobSoapOut

This message is sent from the protocol server to the protocol client when the **DeleteJob** operation is finished.

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

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

The SOAP body MUST contain a **DeleteJobResponse** element.

3.1.4.4.2 Elements

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

Element	Description
DeleteJob	Specifies the body of the **DeleteJobSoapIn** message.
DeleteJobResponse	Specifies the body of the **DeleteJobSoapOut** message.

3.1.4.4.2.1 DeleteJob

This element forms the body of the **DeleteJobSoapIn** message. It MUST adhere to the following XSD:

```
<s:element name="DeleteJob">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="jobId" type="s1:guid" />
    </s:sequence>
  </s:complexType>
</s:element>
```

jobId: The identifier of the import job.

3.1.4.4.2.2 DeleteJobResponse

This element forms the body of the **DeleteJobSoapOut** message. It MUST adhere to the following XSD:

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

3.1.4.4.3 Complex Types

None.

3.1.4.4.4 Simple Types

None.

3.1.4.4.5 Attributes

None.

3.1.4.4.6 Groups

None.

3.1.4.4.7 Attribute Groups

None.

3.1.4.5 GetJobStatus

This operation returns the status and properties of the specified import job, as follows.

```
<wsdl:operation name="GetJobStatus">
    <wsdl:input message="tns:GetJobStatusSoapIn" />
    <wsdl:output message="tns:GetJobStatusSoapOut" />
</wsdl:operation>
```

The protocol client initiates the operation by sending a **GetJobStatusSoapIn** request message to the protocol server, formatted as specified in section 3.1.4.5.1.1.

If the **GetJobStatus** operation is called for a specified import job before the **RunJob** operation is called for that import job, the protocol server MUST respond with a SOAP fault. Otherwise, the protocol server MUST respond with a **GetJobStatusSoapOut** response message, formatted as specified in section 3.1.4.5.1.2.

3.1.4.5.1 Messages

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

Message	Description
GetJobStatusSoapIn	A request to initiate a **GetJobStatus** operation on the protocol server.
GetJobStatusSoapOut	A response from the protocol server at completion of the **GetJobStatus** operation.

3.1.4.5.1.1 GetJobStatusSoapIn

This message is sent from the protocol client to the protocol server to initiate a **GetJobStatus** operation.

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

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

The SOAP body MUST contain a **GetJobStatus** element.

3.1.4.5.1.2 GetJobStatusSoapOut

This message is sent from the protocol server to the protocol client when the **GetJobStatus** operation is finished.

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

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

The SOAP MUST contain a **GetJobStatusResponse** element.

3.1.4.5.2 Elements

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

Element	Description
GetJobStatus	Specifies the body of the **GetJobStatusSoapIn** message.
GetJobStatusResponse	Specifies the body of the **GetJobStatusSoapOut** message.

3.1.4.5.2.1 GetJobStatus

This element forms the body of the **GetJobStatusSoapIn** message. It MUST adhere to the following XSD:

```
<s:element name="GetJobStatus">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="jobId" type="s1:guid" />
    </s:sequence>
  </s:complexType>
</s:element>
```

jobId: The identifier of the import job.

3.1.4.5.2.2 GetJobStatusResponse

This element forms the body of the **GetJobStatusSoapOut** message. It MUST adhere to the following XSD:

```
<s:element name="GetJobStatusResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="GetJobStatusResult"
type="tns:ContentDeploymentRemoteJobStatus" />
    </s:sequence>
  </s:complexType>
</s:element>
```

GetJobStatusResult: The status and properties of the specified import job.

3.1.4.5.3 Complex Types

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

Complex type	Description
ContentDeploymentRemoteJobStatus	Specifies the status and properties of an import job.
ArrayOfReportMessage	Specifies a sequence of **ReportMessages.**
ReportMessage	Specifies a single error, warning, or informational message

3.1.4.5.3.1 ContentDeploymentRemoteJobStatus

A **ContentDeploymentRemoteJobStatus** object specifies the status and properties of an import job. Its structure is specified by the following XSD:<4>

```
<s:complexType name="ContentDeploymentRemoteJobStatus">
  <s:sequence>
    <s:element minOccurs="1" maxOccurs="1" name="Status"
type="tns:ContentDeploymentJobStatus" />
```

```
      <s:element minOccurs="1" maxOccurs="1" name="StatusMessage" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="LastModifiedTime" type="s:dateTime" />
      <s:element minOccurs="1" maxOccurs="1" name="StartTime" type="s:dateTime" />
      <s:element minOccurs="1" maxOccurs="1" name="EndTime" type="s:dateTime" />
      <s:element minOccurs="1" maxOccurs="1" name="ImportedObjectCount" type="s:int" />
      <s:element minOccurs="1" maxOccurs="1" name="ContentSize" type="s:long" />
      <s:element minOccurs="1" maxOccurs="1" name="ImportLogFileMachineName" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="ImportLogFilePath" type="s:string" />
      <s:element minOccurs="1" maxOccurs="1" name="ReportMessages"
  type="tns:ArrayOfReportMessage" />
    </s:sequence>
  </s:complexType>
```

Status: The status of the import job.

StatusMessage: The status message of the import job.

LastModifiedTime: The date and time at which the import job was last modified.

StartTime: The date and time at which the import job started.

EndTime: The date and time at which the import job ended.

ImportedObjectCount: The current count of imported objects. This parameter MUST be a non-negative value.

ContentSize: This parameter MUST be 0.

ImportLogFileMachineName: If not an empty string, the name of the computer containing the import job log file. If empty, the protocol server did not have this information available at the time the message was sent.

ImportLogFilePath: If not an empty string, the absolute path to the import job log file. If empty, the protocol server did not have this information available at the time the message was sent.

ReportMessages: The sequence of error, warning, and informational messages generated by the import job during its run.

3.1.4.5.3.2   ArrayOfReportMessage

An **ArrayOfReportMessage** is a sequence of **ReportMessages** and is specified by the following XSD:

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

ReportMessage: A single error, warning, or informational message.

3.1.4.5.3.3   ReportMessage

The **ReportMessage** type specifies a single error, warning, or informational message. Its structure is specified by the following XSD:

```
<s:complexType name="ReportMessage">
  <s:attribute name="Title" type="s:string"/>
  <s:attribute name="Time" type="s:dateTime" use="required"/>
  <s:attribute name="Severity" type="tns:MessageSeverity" use="required"/>
  <s:attribute name="Description" type="s:string"/>
  <s:attribute name="Recommendation" type="s:string"/>
  <s:attribute name="ObjectName" type="s:string"/>
  <s:attribute name="Phase" type="s:string"/>
</s:complexType>
```

Title: The title of the message.

Time: The date and time at which the message was generated.

Severity: The severity of the message as specified in section 3.1.4.5.4.2.

Description: The text of the message.

Recommendation: The recommended action in response to the message.

ObjectName: The URL or name of the object to which the message applies.

Phase: The phase of the import job when the message was generated.

3.1.4.5.4   Simple Types

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

Simple type	Description
ContentDeploymentJobStatus	Specifies the status of an import job.
MessageSeverity	Specifies the severity level of a message.

3.1.4.5.4.1   ContentDeploymentJobStatus

A **ContentDeploymentJobStatus** object specifies the status of an import job. Its structure is specified by the following XSD:

```
<s:simpleType name="ContentDeploymentJobStatus">
  <s:restriction base="s:string">
    <s:enumeration value="Success"/>
    <s:enumeration value="Failure"/>
    <s:enumeration value="Cancelled"/>
    <s:enumeration value="ImportInProgress"/>
    <s:enumeration value="CancelInProgress"/>
    <s:enumeration value="ImportPreparing"/>
    <s:enumeration value="ImportTimedOut"/>
  </s:restriction>
</s:simpleType>
```

The following table specifies possible values.

Value	Meaning
Success	Indicates that the import job succeeded.
Failure	Indicates that the import job failed.
Cancelled	Indicates that the import job has been cancelled.
ImportInProgress	Indicates that the import job is currently in progress.
CancelInProgress	Indicates that a request to cancel the import job is currently being processed.
ImportPreparing	Indicates that the import job is preparing to run.
ImportTimedOut	Indicates that the import job has timed out.

3.1.4.5.4.2 MessageSeverity

A **MessageSeverity** object specifies the severity level of a message. Its structure is specified by the following XSD:

```
<s:simpleType name="MessageSeverity">
  <s:restriction base="s:string">
    <s:enumeration value="Informational"/>
    <s:enumeration value="Warning"/>
    <s:enumeration value="Error"/>
  </s:restriction>
</s:simpleType>
```

The following table specifies possible values.

Value	Meaning
Informational	The message is informational.
Warning	The message is a warning. A Warning is more severe than an Informational message.
Error	The message describes an error. An Error is more severe than a Warning.

3.1.4.5.5 Attributes

None.

3.1.4.5.6 Groups

None.

3.1.4.5.7 Attribute Groups

None.

3.1.4.6 GetRemoteAdminServerUrl

This operation returns the URL of the central administration site on the remote import server as follows.

```
    <wsdl:operation name="GetRemoteAdminServerUrl">
        <wsdl:input message="tns:GetRemoteAdminServerUrlSoapIn" />
        <wsdl:output message="tns:GetRemoteAdminServerUrlSoapOut" />
    </wsdl:operation>
```

The protocol client initiates the operation by sending a **GetRemoteAdminServerUrlSoapIn** request message to the protocol server, formatted as specified in section 3.1.4.6.1.1.

The protocol server MUST respond with a **GetRemoteAdminServerUrlSoapOut** response message, formatted as specified in section 3.1.4.6.1.2.

3.1.4.6.1 Messages

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

Message	Description
GetRemoteAdminServerUrlSoapIn	A request to initiate a **GetRemoteAdminServerUrl** operation on the protocol server.
GetRemoteAdminServerUrlSoapOut	A response from the protocol server at completion of the **GetRemoteAdminServerUrl** operation.

3.1.4.6.1.1 GetRemoteAdminServerUrlSoapIn

This message is sent from the protocol client to the protocol server to initiate a **GetRemoteAdminServerUrl** operation.

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

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

The SOAP body MUST contain a **GetRemoteAdminServerUrl** element.

3.1.4.6.1.2 GetRemoteAdminServerUrlSoapOut

This message is sent from the protocol server to the protocol client when the **GetRemoteAdminServerUrl** operation is finished.

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

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

The SOAP body MUST contain a **GetRemoteAdminServerUrlResponse** element.

3.1.4.6.2 Elements

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

Element	Description
GetRemoteAdminServerUrl	A request to initiate a **GetRemoteAdminServerUrlSoapIn** operation on the protocol server.
GetRemoteAdminServerUrlResponse	A response from the protocol server at completion of the **GetRemoteAdminServerUrlSoapOut** operation.

3.1.4.6.2.1 GetRemoteAdminServerUrl

This element forms the body of the **GetRemoteAdminServerUrlSoapIn** message. It MUST adhere to the following XSD:

```
<s:element name="GetRemoteAdminServerUrl">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="clientVersion" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

clientVersion: Version of the library which implements the protocol client. The value MUST be formatted as specified by **VersionValue** in [MS-NRTP] section 2.2.1.3. This element MUST be ignored by the protocol server.

3.1.4.6.2.2 GetRemoteAdminServerUrlResponse

This element forms the body of the **GetRemoteAdminServerUrlSoapOut** message. It MUST adhere to the following XSD:

```
<s:element name="GetRemoteAdminServerUrlResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="GetRemoteAdminServerUrlResult"
type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

GetRemoteAdminServerUrlResult: The URL of the central administration site on the remote import server.

3.1.4.6.3 Complex Types

None.

3.1.4.6.4 Simple Types

None.

3.1.4.6.5 Attributes

None.

3.1.4.6.6   Groups

None.

3.1.4.6.7   Attribute Groups

None.

3.1.4.7   GetSiteCollectionNames

This operation returns the names of all site collections within the specified Web application (1) as follows.

```
<wsdl:operation name="GetSiteCollectionNames">
    <wsdl:input message="tns:GetSiteCollectionNamesSoapIn" />
    <wsdl:output message="tns:GetSiteCollectionNamesSoapOut" />
</wsdl:operation>
```

The protocol client initiates the operation by sending a **GetSiteCollectionNamesSoapIn** request message to the protocol server, formatted as specified in section 3.1.4.7.1.1.

The protocol server MUST respond with a **GetSiteCollectionNamesSoapOut** response message, formatted as specified in section 3.1.4.7.1.2.

3.1.4.7.1   Messages

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

Message	Description
GetSiteCollectionNamesSoapIn	A request to initiate a **GetSiteCollectionNames** operation on the protocol server.
GetSiteCollectionNamesSoapOut	A response from the protocol server at completion of the **GetSiteCollectionNames** operation.

3.1.4.7.1.1   GetSiteCollectionNamesSoapIn

This message is sent from the protocol client to the protocol server to initiate a **GetSiteCollectionNames** operation.

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

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

The SOAP body MUST contain a **GetSiteCollectionNames** element.

3.1.4.7.1.2   GetSiteCollectionNamesSoapOut

This message is sent from the protocol server to the protocol client when the **GetSiteCollectionNames** operation is finished.

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

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

The SOAP body MUST contain a **GetSiteCollectionNamesResponse** element.

3.1.4.7.2   Elements

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

Element	Description
GetSiteCollectionNames	Specifies the body of the **GetSiteCollectionNamesSoapIn** message.
GetSiteCollectionNamesResponse	Specifies the body of the **GetSiteCollectionNamesSoapOut** message.

3.1.4.7.2.1   GetSiteCollectionNames

This element forms the body of the **GetSiteCollectionNamesSoapIn** message. It MUST adhere to the following XSD:

```
<s:element name="GetSiteCollectionNames">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="virtualServerUrl" type="s:string" />
    </s:sequence>
  </s:complexType>
</s:element>
```

virtualServerUrl: The URL of an existing Web application (1) on the farm.

3.1.4.7.2.2   GetSiteCollectionNamesResponse

This element forms the body of the **GetSiteCollectionNamesSoapOut** message. It MUST adhere to the following XSD:

```
<s:element name="GetSiteCollectionNamesResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="GetSiteCollectionNamesResult"
type="tns:ArrayOfString" />
    </s:sequence>
  </s:complexType>
</s:element>
```

GetSiteCollectionNamesResult: The names of all site collections within the specified Web application (1).

3.1.4.7.3   Complex Types

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

Complex type	Description
ArrayOfString	Specifies a sequence of strings.

3.1.4.7.3.1   ArrayOfString

A sequence of strings as specified by the following XSD:

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

string: A single string within the **ArrayOfString**.

3.1.4.7.4   Simple Types

None.

3.1.4.7.5   Attributes

None.

3.1.4.7.6   Groups

None.

3.1.4.7.7   Attribute Groups

None.

3.1.4.8   GetVirtualServersInformation

This operation returns the names and URLs of all Web applications (1) within the farm, as follows.

```
<wsdl:operation name="GetVirtualServersInformation">
    <wsdl:input message="tns:GetVirtualServersInformationSoapIn" />
    <wsdl:output message="tns:GetVirtualServersInformationSoapOut" />
</wsdl:operation>
```

The protocol client initiates the operation by sending a **GetVirtualServersInformationSoapIn** request message to the protocol server, formatted as specified in section 3.1.4.8.1.1.

The protocol server MUST respond with a **GetVirtualServersInformationSoapOut** response message, formatted as specified in section 3.1.4.8.1.2.

3.1.4.8.1   Messages

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

Message	Description
GetVirtualServersInformationSoapIn	A request to initiate a **GetVirtualServersInformation** operation on the protocol server.
GetVirtualServersInformationSoapOut	A response from the protocol server at completion of the **GetVirtualServersInformation** operation.

3.1.4.8.1.1 GetVirtualServersInformationSoapIn

This message is sent from the protocol client to the protocol server to initiate a **GetVirtualServersInformation** operation.

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

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

The SOAP body MUST contain a **GetVirtualServersInformation** element.

3.1.4.8.1.2 GetVirtualServersInformationSoapOut

This message is sent from the protocol server to the protocol client when the **GetVirtualServersInformation** operation is finished.

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

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

The SOAP body MUST contain a **GetVirtualServersInformationResponse** element.

3.1.4.8.2 Elements

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

Element	Description
GetVirtualServersInformation	Specifies the body of the **GetVirtualServersInformationSoapIn** message.
GetVirtualServersInformationResponse	Specifies the body of the **GetVirtualServersInformationSoapOut** message.

3.1.4.8.2.1 GetVirtualServersInformation

This element forms the body of the **GetVirtualServersInformationSoapIn** message. It MUST adhere to the following XSD:

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

3.1.4.8.2.2 GetVirtualServersInformationResponse

This element forms the body of the **GetVirtualServersInformationSoapOut** message. It MUST adhere to the following XSD:

```
<s:element name="GetVirtualServersInformationResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="GetVirtualServersInformationResult"
type="tns:ArrayOfVirtualServerInformation" />
    </s:sequence>
  </s:complexType>
</s:element>
```

GetVirtualServersInformationResult: The names and URLs of all Web applications (1) within the farm.

3.1.4.8.3 Complex Types

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

Complex type	Description
ArrayOfVirtualServerInformation	Specifies a sequence of **VirtualServerInformation** elements.
VirtualServerInformation	Specifies the name and URL of a Web application (1).

3.1.4.8.3.1 ArrayOfVirtualServerInformation

A sequence of **VirtualServerInformation** elements as specified by the following XSD:

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

VirtualServerInformation: The name and URL of a single Web application (1) within the farm.

3.1.4.8.3.2 VirtualServerInformation

The name and URL of a Web application (1) as specified by the following XSD:

```
<s:complexType name="VirtualServerInformation">
  <s:sequence>
    <s:element minOccurs="1" maxOccurs="1" name="VirtualServerName" type="s:string" />
    <s:element minOccurs="1" maxOccurs="1" name="VirtualServerUrl" type="s:string" />
  </s:sequence>
</s:complexType>
```

VirtualServerName: The name of the Web application (1).

VirtualServerUrl: The URL of the Web application (1).

3.1.4.8.4 Simple Types

None.

3.1.4.8.5 Attributes

None.

3.1.4.8.6 Groups

None.

3.1.4.8.7 Attribute Groups

None.

3.1.4.9 RunJob

This operation runs the specified import job, as follows.

```
<wsdl:operation name="RunJob">
    <wsdl:input message="tns:RunJobSoapIn" />
    <wsdl:output message="tns:RunJobSoapOut" />
</wsdl:operation>
```

The protocol client initiates the operation by sending a **RunJobSoapIn** request message to the protocol server, formatted as specified in section 3.1.4.9.1.1.

If the **jobId** parameter of the **RunJobSoapIn** message identifies an existing import job that is not in any of the following states—**ImportPreparing**, **ImportInProgress**, **ImportTimeOut**, or **CancelInProgress**—the protocol server MUST execute the import job. Otherwise, the protocol server MUST take no action.

When the operation is finished, the protocol server MUST respond with a **RunJobSoapOut** response message, formatted as specified in section 3.1.4.9.1.2.

3.1.4.9.1 Messages

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

Message	Description
RunJobSoapIn	A request to initiate a **RunJob** operation on the protocol server.
RunJobSoapOut	A response from the protocol server at completion of the **RunJob** operation.

3.1.4.9.1.1 RunJobSoapIn

This message is sent from the protocol client to the protocol server to initiate a **RunJob** operation.

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

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

The SOAP body MUST contain a **RunJob** element.

3.1.4.9.1.2 RunJobSoapOut

This message is sent from the protocol server to the protocol client when the **RunJob** operation is finished.

The SOAP action value of the message is specified as follows:

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

The SOAP body MUST contain a **RunJobResponse** element.

3.1.4.9.2 Elements

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

Element	Description
RunJob	Specifies the body of the **RunJobSoapIn** message.
RunJobResponse	Specifies the body of the **RunJobSoapOut** message.

3.1.4.9.2.1 RunJob

This element forms the body of the **RunJobSoapIn** message. It MUST adhere to the following XSD:

```
<s:element name="RunJob">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="jobId" type="s1:guid" />
    </s:sequence>
  </s:complexType>
</s:element>
```

jobId: The identifier of the import job.

3.1.4.9.2.2 RunJobResponse

This element forms the body of the **RunJobSoapOut** message. It MUST adhere to the following XSD:

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

3.1.4.9.3   Complex Types

None.

3.1.4.9.4   Simple Types

None.

3.1.4.9.5   Attributes

None.

3.1.4.9.6   Groups

None.

3.1.4.9.7   Attribute Groups

None.

3.1.5   Timer Events

None.

3.1.6   Other Local Events

None.

3.2   ContentDeploymentRemoteImport Client Details

3.2.1   Abstract Data Model

None.

3.2.2   Timers

None.

3.2.3   Initialization

None.

3.2.4   Message Processing Events and Sequencing Rules

To perform a successful remote import operation, the protocol client MUST have farm administrator rights on the remote import server and MUST perform the following operations in order:

1. Export content from the **export server** and store it in a local deployment package (1).

   The deployment package (1)<5> MUST adhere to the format specified in [MS-PRIMEPF].

   The names of the deployment package (1) files MUST be as specified in [MS-PRIMEPF] section 2. For a compressed deployment package (1) (consisting of one or more cabinet files) names of the cabinet files MUST be **BaseFileName**.cab for the first file and **BaseFileName***n*.cab for

subsequent files, where **BaseFileName** equals ExportedFiles<6>, and *n* is an integer starting with 1 and incrementing monotonically.

The deployment package (1) MUST contain the file ExportSettings.xml specified in [MS-PRIMEPF] section 2.2 with the following settings:

IncludeSecurity: MUST be set to one of the values specified in [MS-PRIMEPF] section 2.2.4.4.

IncludeVersions: MUST be set to **SPIncludeVersions.LastMajorAndMinor** as specified in [MS-PRIMEPF] section 2.2.4.5.

ExcludeDependencies: MUST be set to false.

The mechanism by which the protocol client performs this export operation is outside the scope of this protocol.

2. Call the **CreateJob** operation to create an import job.

   The protocol client MUST save the identifier of the newly created import job that is returned by the protocol server. This identifier will be needed in step 4.

3. Call the **GetRemoteAdminServerUrl** operation to retrieve the base URL for uploading the deployment package (1).

4. Use the **HTTP POST** operation to upload the deployment package (1) to the following URL:

   AdminServerUrl/_admin/Content Deployment/DeploymentUpload.aspx?filename=*FileName*&remoteJobId=*JobId*

   where *AdminServerUrl* MUST be the URL retrieved in step 3, *FileName* MUST be the name of the file being uploaded, and *JobId* MUST be the identifier of the import job returned by the call to **CreateJob** in step 2.

   If there are multiple files in the deployment package (1), the protocol client MUST use the HTTP POST operation repeatedly to upload each file.

5. Call the **RunJob** operation to import the deployment package (1).

6. Optionally, call the **GetJobStatus** operation to retrieve the status of the import job.

3.2.5 Timer Events

None.

3.2.6 Other Local Events

None.

4 Protocol Examples

4.1 GetRemoteAdminServerUrl

If a protocol client wants to retrieve the URL of the central administration site of the remote import server, the protocol client sends a **SOAP message** such as 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>
    <GetRemoteAdminServerUrl
      xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <clientVersion>
        12.0.6211.1000
      </clientVersion>
    </GetRemoteAdminServerUrl>
  </soap:Body>
</soap:Envelope>
```

The protocol server responds with a SOAP message such as 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>
    <GetRemoteAdminServerUrlResponse
     xmlns="http://schemas.microsoft.com/sharepoint/soap/">
     <GetRemoteAdminServerUrlResult>
        http://www.contoso.com:8000
      </GetRemoteAdminServerUrlResult>
    </GetRemoteAdminServerUrlResponse>
  </soap:Body>
</soap:Envelope>
```

4.2 CreateJob

If a protocol client wants to create a new import job, the protocol client sends a SOAP message such as 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>
    <CreateJob
      xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <description>
        Update Product Sites
      </description>
```

```
            <sourceId>
              6faeca33-f332-488b-93b8-2ecfb858ca4e
            </sourceId>
            <destinationServerUrl>
              http://contoso.com:90
            </destinationServerUrl>
            <destinationSiteCollection>
              /Product Sites
            </destinationSiteCollection>
            <includeSecurity>
              All
            </includeSecurity>
            <includeUserInfoDateTime>
              ImportAll
            </includeUserInfoDateTime>
            <enableEventReceivers>
              true
            </enableEventReceivers>
            <fileCompression>
              true
            </fileCompression>
          </CreateJob>
      </soap:Body>
    </soap:Envelope>
```

The protocol server responds with a SOAP message such as 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>
    <CreateJobResponse
      xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <CreateJobResult>
        0fdac8a8-87ca-45d7-8eb6-ba5c976f5b57
      </CreateJobResult>
    </CreateJobResponse>
  </soap:Body>
</soap:Envelope>
```

4.3 GetJobStatus

If a protocol client wants to obtain the status of an existing import job, the protocol client sends a SOAP message, such as 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>
    <GetJobStatus
      xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <jobId>
```

```
        0fdac8a8-87ca-45d7-8eb6-ba5c976f5b57
      </jobId>
    </GetJobStatus>
  </soap:Body>
</soap:Envelope>
```

The protocol server responds with a SOAP message, such as 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>
    <GetJobStatusResponse
       xmlns="http://schemas.microsoft.com/sharepoint/soap/">
      <GetJobStatusResult>
        <Status>
           ImportInProgress
        </Status>
        <StatusMessage>
           Content deployment job 'Remote import job with sourceID = CDJob001': Import in
progress
        </StatusMessage>
        <LastModifiedTime>
           2008-02-15T19:09:09
        </LastModifiedTime>
        <StartTime>
           2008-02-15T19:08:31
        </StartTime>
        <EndTime>
           2008-02-15T19:09:09
        </EndTime>
        <ImportedObjectCount>
           324
        </ImportedObjectCount>
        <ContentSize>
           0
        </ContentSize>
        <ImportLogFileMachineName>cox64-112</ImportLogFileMachineName>
        <ImportLogFilePath>
D:\ProgramData\ContentDeployment\ImportExportLogs\398052fb-86ca-4d3d-914b-
2e952cebcff2\ExportLog_20091210_082455.zip
        </ImportLogFilePath>
        <ReportMessages/>
      </GetJobStatusResult>
    </GetJobStatusResponse>
  </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 version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:s="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:tns="http://schemas.microsoft.com/sharepoint/soap/"
xmlns:s1="http://microsoft.com/wsdl/types/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
targetNamespace="http://schemas.microsoft.com/sharepoint/soap/">
  <wsdl:types>
    <s:schema elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/sharepoint/soap/">
      <s:import namespace="http://microsoft.com/wsdl/types/" />
      <s:element name="GetRemoteAdminServerUrl">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="clientVersion" type="s:string" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetRemoteAdminServerUrlResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="GetRemoteAdminServerUrlResult"
type="s:string" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetVirtualServersInformation">
        <s:complexType />
      </s:element>
      <s:element name="GetVirtualServersInformationResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="GetVirtualServersInformationResult"
type="tns:ArrayOfVirtualServerInformation" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:complexType name="ArrayOfVirtualServerInformation">
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="unbounded" name="VirtualServerInformation"
nillable="true" type="tns:VirtualServerInformation" />
        </s:sequence>
      </s:complexType>
      <s:complexType name="VirtualServerInformation">
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="VirtualServerName" type="s:string" />
          <s:element minOccurs="1" maxOccurs="1" name="VirtualServerUrl" type="s:string" />
        </s:sequence>
      </s:complexType>
      <s:element name="GetSiteCollectionNames">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="virtualServerUrl" type="s:string" />
          </s:sequence>
```

```
          </s:complexType>
        </s:element>
        <s:element name="GetSiteCollectionNamesResponse">
          <s:complexType>
            <s:sequence>
              <s:element minOccurs="1" maxOccurs="1" name="GetSiteCollectionNamesResult"
  type="tns:ArrayOfString" />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:complexType name="ArrayOfString">
          <s:sequence>
            <s:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
  type="s:string" />
          </s:sequence>
        </s:complexType>
        <s:element name="CreateJob">
          <s:complexType>
            <s:sequence>
              <s:element minOccurs="1" maxOccurs="1" name="description" type="s:string" />
              <s:element minOccurs="1" maxOccurs="1" name="sourceId" type="s1:guid" />
              <s:element minOccurs="1" maxOccurs="1" name="destinationServerUrl"
  type="s:string" />
              <s:element minOccurs="1" maxOccurs="1" name="destinationSiteCollection"
  type="s:string" />
              <s:element minOccurs="1" maxOccurs="1" name="includeSecurity"
  type="tns:SPIncludeSecurity" />
              <s:element minOccurs="1" maxOccurs="1" name="includeUserInfoDateTime"
  type="tns:SPImportUserInfoDateTimeOption" />
              <s:element minOccurs="1" maxOccurs="1" name="enableEventReceivers"
  type="s:boolean" />
              <s:element minOccurs="1" maxOccurs="1" name="fileCompression" type="s:boolean" />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:simpleType name="SPIncludeSecurity">
          <s:restriction base="s:string">
            <s:enumeration value="None" />
            <s:enumeration value="WssOnly" />
            <s:enumeration value="All" />
          </s:restriction>
        </s:simpleType>
        <s:simpleType name="SPImportUserInfoDateTimeOption">
          <s:restriction base="s:string">
            <s:enumeration value="None" />
            <s:enumeration value="ReplaceUserWithSystemAccount" />
            <s:enumeration value="ImportAll" />
          </s:restriction>
        </s:simpleType>
        <s:element name="CreateJobResponse">
          <s:complexType>
            <s:sequence>
              <s:element minOccurs="1" maxOccurs="1" name="CreateJobResult" type="s1:guid" />
            </s:sequence>
          </s:complexType>
        </s:element>
        <s:element name="GetJobStatus">
          <s:complexType>
            <s:sequence>
```

```
            <s:element minOccurs="1" maxOccurs="1" name="jobId" type="s1:guid" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element name="GetJobStatusResponse">
        <s:complexType>
          <s:sequence>
            <s:element minOccurs="1" maxOccurs="1" name="GetJobStatusResult"
  type="tns:ContentDeploymentRemoteJobStatus" />
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:complexType name="ContentDeploymentRemoteJobStatus">
        <s:sequence>
          <s:element minOccurs="1" maxOccurs="1" name="Status"
  type="tns:ContentDeploymentJobStatus" />
          <s:element minOccurs="1" maxOccurs="1" name="StatusMessage" type="s:string" />
          <s:element minOccurs="1" maxOccurs="1" name="LastModifiedTime" type="s:dateTime" />
          <s:element minOccurs="1" maxOccurs="1" name="StartTime" type="s:dateTime" />
          <s:element minOccurs="1" maxOccurs="1" name="EndTime" type="s:dateTime" />
          <s:element minOccurs="1" maxOccurs="1" name="ImportedObjectCount" type="s:int" />
          <s:element minOccurs="1" maxOccurs="1" name="ContentSize" type="s:long" />
          <s:element minOccurs="1" maxOccurs="1" name="ImportLogFileMachineName"
  type="s:string" />
          <s:element minOccurs="1" maxOccurs="1" name="ImportLogFilePath" type="s:string" />
          <s:element minOccurs="1" maxOccurs="1" name="ReportMessages"
  type="tns:ArrayOfReportMessage" />
        </s:sequence>
      </s:complexType>
      <s:simpleType name="ContentDeploymentJobStatus">
        <s:restriction base="s:string">
          <s:enumeration value="Success" />
          <s:enumeration value="Failure" />
          <s:enumeration value="Cancelled" />
          <s:enumeration value="ImportInProgress" />
          <s:enumeration value="CancelInProgress" />
          <s:enumeration value="ImportPreparing" />
          <s:enumeration value="ImportTimedOut" />
        </s:restriction>
      </s:simpleType>
      <s:complexType name="ArrayOfReportMessage">
        <s:sequence>
          <s:element minOccurs="0" maxOccurs="unbounded" name="ReportMessage" nillable="true"
  type="tns:ReportMessage" />
        </s:sequence>
      </s:complexType>
      <s:complexType name="ReportMessage">
        <s:attribute name="Title" type="s:string" />
        <s:attribute name="Time" type="s:dateTime" use="required" />
        <s:attribute name="Severity" type="tns:MessageSeverity" use="required" />
        <s:attribute name="Description" type="s:string" />
        <s:attribute name="Recommendation" type="s:string" />
        <s:attribute name="ObjectName" type="s:string" />
        <s:attribute name="Phase" type="s:string" />
      </s:complexType>
      <s:simpleType name="MessageSeverity">
        <s:restriction base="s:string">
          <s:enumeration value="Informational" />
          <s:enumeration value="Warning" />
```

```
              <s:enumeration value="Error" />
            </s:restriction>
          </s:simpleType>
          <s:element name="RunJob">
            <s:complexType>
              <s:sequence>
                <s:element minOccurs="1" maxOccurs="1" name="jobId" type="s1:guid" />
              </s:sequence>
            </s:complexType>
          </s:element>
          <s:element name="RunJobResponse">
            <s:complexType />
          </s:element>
          <s:element name="DeleteJob">
            <s:complexType>
              <s:sequence>
                <s:element minOccurs="1" maxOccurs="1" name="jobId" type="s1:guid" />
              </s:sequence>
            </s:complexType>
          </s:element>
          <s:element name="DeleteJobResponse">
            <s:complexType />
          </s:element>
          <s:element name="CancelJob">
            <s:complexType>
              <s:sequence>
                <s:element minOccurs="1" maxOccurs="1" name="jobId" type="s1:guid" />
              </s:sequence>
            </s:complexType>
          </s:element>
          <s:element name="CancelJobResponse">
            <s:complexType />
          </s:element>
          <s:element name="DeleteImportLogs">
            <s:complexType>
              <s:sequence>
                <s:element minOccurs="1" maxOccurs="1" name="sourceJobId" type="s1:guid" />
              </s:sequence>
            </s:complexType>
          </s:element>
          <s:element name="DeleteImportLogsResponse">
            <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="GetRemoteAdminServerUrlSoapIn">
        <wsdl:part name="parameters" element="tns:GetRemoteAdminServerUrl" />
      </wsdl:message>
      <wsdl:message name="GetRemoteAdminServerUrlSoapOut">
        <wsdl:part name="parameters" element="tns:GetRemoteAdminServerUrlResponse" />
```

```
    </wsdl:message>
    <wsdl:message name="GetVirtualServersInformationSoapIn">
      <wsdl:part name="parameters" element="tns:GetVirtualServersInformation" />
    </wsdl:message>
    <wsdl:message name="GetVirtualServersInformationSoapOut">
      <wsdl:part name="parameters" element="tns:GetVirtualServersInformationResponse" />
    </wsdl:message>
    <wsdl:message name="GetSiteCollectionNamesSoapIn">
      <wsdl:part name="parameters" element="tns:GetSiteCollectionNames" />
    </wsdl:message>
    <wsdl:message name="GetSiteCollectionNamesSoapOut">
      <wsdl:part name="parameters" element="tns:GetSiteCollectionNamesResponse" />
    </wsdl:message>
    <wsdl:message name="CreateJobSoapIn">
      <wsdl:part name="parameters" element="tns:CreateJob" />
    </wsdl:message>
    <wsdl:message name="CreateJobSoapOut">
      <wsdl:part name="parameters" element="tns:CreateJobResponse" />
    </wsdl:message>
    <wsdl:message name="GetJobStatusSoapIn">
      <wsdl:part name="parameters" element="tns:GetJobStatus" />
    </wsdl:message>
    <wsdl:message name="GetJobStatusSoapOut">
      <wsdl:part name="parameters" element="tns:GetJobStatusResponse" />
    </wsdl:message>
    <wsdl:message name="RunJobSoapIn">
      <wsdl:part name="parameters" element="tns:RunJob" />
    </wsdl:message>
    <wsdl:message name="RunJobSoapOut">
      <wsdl:part name="parameters" element="tns:RunJobResponse" />
    </wsdl:message>
    <wsdl:message name="DeleteJobSoapIn">
      <wsdl:part name="parameters" element="tns:DeleteJob" />
    </wsdl:message>
    <wsdl:message name="DeleteJobSoapOut">
      <wsdl:part name="parameters" element="tns:DeleteJobResponse" />
    </wsdl:message>
    <wsdl:message name="CancelJobSoapIn">
      <wsdl:part name="parameters" element="tns:CancelJob" />
    </wsdl:message>
    <wsdl:message name="CancelJobSoapOut">
      <wsdl:part name="parameters" element="tns:CancelJobResponse" />
    </wsdl:message>
    <wsdl:message name="DeleteImportLogsSoapIn">
      <wsdl:part name="parameters" element="tns:DeleteImportLogs" />
    </wsdl:message>
    <wsdl:message name="DeleteImportLogsSoapOut">
      <wsdl:part name="parameters" element="tns:DeleteImportLogsResponse" />
    </wsdl:message>
    <wsdl:portType name="ContentDeploymentRemoteImportSoap">
      <wsdl:operation name="GetRemoteAdminServerUrl">
        <wsdl:input message="tns:GetRemoteAdminServerUrlSoapIn" />
        <wsdl:output message="tns:GetRemoteAdminServerUrlSoapOut" />
      </wsdl:operation>
      <wsdl:operation name="GetVirtualServersInformation">
        <wsdl:input message="tns:GetVirtualServersInformationSoapIn" />
        <wsdl:output message="tns:GetVirtualServersInformationSoapOut" />
      </wsdl:operation>
      <wsdl:operation name="GetSiteCollectionNames">
```

```
      <wsdl:input message="tns:GetSiteCollectionNamesSoapIn" />
      <wsdl:output message="tns:GetSiteCollectionNamesSoapOut" />
    </wsdl:operation>
    <wsdl:operation name="CreateJob">
      <wsdl:input message="tns:CreateJobSoapIn" />
      <wsdl:output message="tns:CreateJobSoapOut" />
    </wsdl:operation>
    <wsdl:operation name="GetJobStatus">
      <wsdl:input message="tns:GetJobStatusSoapIn" />
      <wsdl:output message="tns:GetJobStatusSoapOut" />
    </wsdl:operation>
    <wsdl:operation name="RunJob">
      <wsdl:input message="tns:RunJobSoapIn" />
      <wsdl:output message="tns:RunJobSoapOut" />
    </wsdl:operation>
    <wsdl:operation name="DeleteJob">
      <wsdl:input message="tns:DeleteJobSoapIn" />
      <wsdl:output message="tns:DeleteJobSoapOut" />
    </wsdl:operation>
    <wsdl:operation name="CancelJob">
      <wsdl:input message="tns:CancelJobSoapIn" />
      <wsdl:output message="tns:CancelJobSoapOut" />
    </wsdl:operation>
    <wsdl:operation name="DeleteImportLogs">
      <wsdl:input message="tns:DeleteImportLogsSoapIn" />
      <wsdl:output message="tns:DeleteImportLogsSoapOut" />
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="ContentDeploymentRemoteImportSoap"
type="tns:ContentDeploymentRemoteImportSoap">
    <soap:binding transport="http://schemas.xmlsoap.org/soap/http" />
    <wsdl:operation name="GetRemoteAdminServerUrl">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetRemoteAdminServerUrl"
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetVirtualServersInformation">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetVirtualServersInformation"
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal" />
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetSiteCollectionNames">
      <soap:operation
soapAction="http://schemas.microsoft.com/sharepoint/soap/GetSiteCollectionNames"
style="document" />
      <wsdl:input>
        <soap:body use="literal" />
```

```
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="CreateJob">
        <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/CreateJob"
  style="document" />
        <wsdl:input>
          <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="GetJobStatus">
        <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/GetJobStatus"
  style="document" />
        <wsdl:input>
          <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="RunJob">
        <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/RunJob"
  style="document" />
        <wsdl:input>
          <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="DeleteJob">
        <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteJob"
  style="document" />
        <wsdl:input>
          <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="CancelJob">
        <soap:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/CancelJob"
  style="document" />
        <wsdl:input>
          <soap:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="DeleteImportLogs">
        <soap:operation
  soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteImportLogs" 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="ContentDeploymentRemoteImportSoap12"
  type="tns:ContentDeploymentRemoteImportSoap">
        <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" />
        <wsdl:operation name="GetRemoteAdminServerUrl">
          <soap12:operation
  soapAction="http://schemas.microsoft.com/sharepoint/soap/GetRemoteAdminServerUrl"
  style="document" />
          <wsdl:input>
            <soap12:body use="literal" />
          </wsdl:input>
          <wsdl:output>
            <soap12:body use="literal" />
          </wsdl:output>
        </wsdl:operation>
        <wsdl:operation name="GetVirtualServersInformation">
          <soap12:operation
  soapAction="http://schemas.microsoft.com/sharepoint/soap/GetVirtualServersInformation"
  style="document" />
          <wsdl:input>
            <soap12:body use="literal" />
          </wsdl:input>
          <wsdl:output>
            <soap12:body use="literal" />
          </wsdl:output>
        </wsdl:operation>
        <wsdl:operation name="GetSiteCollectionNames">
          <soap12:operation
  soapAction="http://schemas.microsoft.com/sharepoint/soap/GetSiteCollectionNames"
  style="document" />
          <wsdl:input>
            <soap12:body use="literal" />
          </wsdl:input>
          <wsdl:output>
            <soap12:body use="literal" />
          </wsdl:output>
        </wsdl:operation>
        <wsdl:operation name="CreateJob">
          <soap12:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/CreateJob"
  style="document" />
          <wsdl:input>
            <soap12:body use="literal" />
          </wsdl:input>
          <wsdl:output>
            <soap12:body use="literal" />
          </wsdl:output>
        </wsdl:operation>
        <wsdl:operation name="GetJobStatus">
          <soap12:operation
  soapAction="http://schemas.microsoft.com/sharepoint/soap/GetJobStatus" style="document" />
          <wsdl:input>
            <soap12:body use="literal" />
          </wsdl:input>
```

```
        <wsdl:output>
          <soap12:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="RunJob">
        <soap12:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/RunJob"
  style="document" />
        <wsdl:input>
          <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap12:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="DeleteJob">
        <soap12:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteJob"
  style="document" />
        <wsdl:input>
          <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap12:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="CancelJob">
        <soap12:operation soapAction="http://schemas.microsoft.com/sharepoint/soap/CancelJob"
  style="document" />
        <wsdl:input>
          <soap12:body use="literal" />
        </wsdl:input>
        <wsdl:output>
          <soap12:body use="literal" />
        </wsdl:output>
      </wsdl:operation>
      <wsdl:operation name="DeleteImportLogs">
        <soap12:operation
  soapAction="http://schemas.microsoft.com/sharepoint/soap/DeleteImportLogs" 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:

- Microsoft® Office SharePoint® Server 2007

- Microsoft® SharePoint® Server 2010

- Microsoft® SharePoint® Server 2013

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 3.1.4.2: Office SharePoint Server 2007 does not create a new import job if the protocol server finds no existing import job with the same **sourceId** parameter to delete but a **timer job** exists with the following name: "ContentDeploymentJobDefinition: Remote import job for job with sourceID = *sourceId*" where *sourceId* is the value of the **sourceId** parameter.

<2> Section 3.1.4.2.2.1: For versions previous to SharePoint Server 2010 the **sourceId** parameter is declared with the type set to "s:string". It MUST adhere to the format specified in section 2.2.5.1.

<3> Section 3.1.4.3: The **DeleteImportLogs** operation is not included in Office SharePoint Server 2007 before update KB2345212 is installed.

<4> Section 3.1.4.5.3.1: The **ImportLogFileMachineName** and **ImportLogFilePath** elements in the **ContentDeploymentRemoteJobStatus** complex type are not included in Office SharePoint Server 2007 before update KB2345212 is installed.

<5> Section 3.2.4: The deployment package (1) generated by Office SharePoint Server 2007 always contains the file ContentDeploymentInformation.xml. This file is reserved and MUST be ignored by import jobs.

<6> Section 3.2.4: For remote import servers running Office SharePoint Server 2007 **BaseFileName** MUST be either ExportedFiles.cab or ExportedFilesRetryNNN_.cab where NNN is a three-digit integer between 001 and 999.

8   Change Tracking

This section identifies changes that were made to the [MS-CDEPLOY] protocol document between the September 2012 and October 2012 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

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 or functionality.

- An extensive rewrite, addition, or deletion of major portions of content.

- The removal of a document from the documentation set.

- Changes made for template compliance.

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 **Editorial** means that the language and formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical or language changes were introduced. The technical content of the document is identical to the last released version, but minor editorial and formatting changes, as well as updates to the header and footer information, and to the revision summary, may have been made.

Major and minor changes can be described further using the following change types:

- New content added.

- Content updated.

- Content removed.

- New product behavior note added.

- Product behavior note updated.

- Product behavior note removed.

- New protocol syntax added.

- Protocol syntax updated.

- Protocol syntax removed.

- New content added due to protocol revision.

- Content updated due to protocol revision.

- Content removed due to protocol revision.

- New protocol syntax added due to protocol revision.

- Protocol syntax updated due to protocol revision.

- Protocol syntax removed due to protocol revision.

- New content added for template compliance.

- Content updated for template compliance.

- Content removed for template compliance.

- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated.**

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.

- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

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

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
1.3 Overview	Changed the name from 'Protocol Overview (Synopsis)' to 'Overview'.	N	Content updated for template compliance.

9 Index

Release: October 8, 2012