

[MS-OXWSTASK]: Tasks 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's Open Specification Promise (available here: <http://www.microsoft.com/interop/osp>) or the Community Promise (available here: <http://www.microsoft.com/interop/cp/default.msp>). 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
07/15/2009	1.0	Major	Initial Availability.
11/04/2009	1.1.0	Minor	Updated the technical content.
02/10/2010	1.1.0	None	Version 1.1.0 release
05/05/2010	1.1.1	Editorial	Revised and edited the technical content.
08/04/2010	1.2	Minor	Clarified the meaning of the technical content.

Contents

1 Introduction	5
1.1 Glossary	5
1.2 References	5
1.2.1 Normative References	5
1.2.2 Informative References	6
1.3 Overview	6
1.4 Relationship to Other Protocols	6
1.5 Prerequisites/Preconditions	7
1.6 Applicability Statement	7
1.7 Versioning and Capability Negotiation	7
1.8 Vendor Extensible Fields	7
1.9 Standards Assignments	7
2 Messages	8
2.1 Transport	8
2.2 Common Message Syntax	8
2.2.1 Namespaces	8
2.2.2 Simple Types	8
2.2.2.1 t:TaskDelegateStateType Simple Type	8
2.2.2.2 t:TaskStatusType Simple Type	9
2.2.3 Complex Types	10
2.2.3.1 t:TaskRecurrenceType Complex Type	11
2.2.3.2 t:TasksFolderType Complex Type	11
2.2.3.3 t:TaskType Complex Type	11
2.2.3.4 t:DailyRegeneratingPatternType Complex Type	14
2.2.3.5 t:MonthlyRegeneratingPatternType Complex Type	15
2.2.3.6 t:RegeneratingPatternBaseType Complex Type	15
2.2.3.7 t:WeeklyRegeneratingPatternType Complex Type	15
2.2.3.8 t:YearlyRegeneratingPatternType Complex Type	15
2.2.4 Elements	16
2.2.5 Attributes	16
2.2.6 Groups	16
2.2.6.1 TaskRecurrencePatternTypes Group	16
2.2.7 Attribute Groups	18
2.2.8 Messages	18
3 Protocol Details	19
3.1 ExchangeServicePortType Server Details	19
3.1.1 Abstract Data Model	19
3.1.2 Timers	19
3.1.3 Initialization	19
3.1.4 Message Processing Events and Sequencing	19
3.1.4.1 CopyItem	20
3.1.4.2 CreateItem	20
3.1.4.3 DeleteItem	21
3.1.4.3.1 Simple Types	21
3.1.4.3.1.1 t:AffectedTaskOccurrencesType Simple Type	21
3.1.4.4 GetItem	22
3.1.4.5 MoveItem	22
3.1.4.6 UpdateItem	23

3.1.5	Timer Events	23
3.1.6	Other Local Events	23
3.2	Client Details.....	23
3.2.1	Abstract Data Model	24
3.2.2	Timers	24
3.2.3	Initialization	24
3.2.4	Message Processing Events and Sequencing	24
3.2.5	Timer Events	24
3.2.6	Other Local Events	24
4	Protocol Examples	25
5	Security	26
5.1	Security Considerations for Implementers.....	26
5.2	Index of Security Parameters	26
6	Appendix A: Full WSDL	27
6.1	WSDL.....	27
6.2	Types Schema.....	31
6.3	Messages Schema.....	33
7	Appendix B: Product Behavior	34
8	Change Tracking.....	35
9	Index	37

1 Introduction

This document specifies the Tasks Web Service protocol, which is responsible for creating, updating, and deleting task items on the server. Clients use the SOAP protocol [\[SOAP1.1\]](#) to contact the tasks service.

1.1 Glossary

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

delegate
Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Socket Layers (HTTPS)
mailbox
property
SOAP body
SOAP fault
SOAP header
SOAP message
Web Services Description Language (WSDL)
WSDL message
WSDL port type
XML
XML namespace
XML schema

The following terms are specific to this document:

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

1.2 References

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-OXWSCDATA] Microsoft Corporation, "[Common Web Service Data Types](#)", July 2009.

[MS-OXWSCORE] Microsoft Corporation, "[Core Items Web Service Protocol Specification](#)", July 2009.

[MS-OXWSFOLD] Microsoft Corporation, "[Folders and Folder Permissions Web Service Protocol Specification](#)", July 2009.

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

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", W3C Note, May 2000, <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

[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., Eds., et al., "Namespaces in XML 1.0 (Third Edition)", December 2009, <http://www.w3.org/TR/REC-xml-names/>

[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., and Malhotra, A., Eds., "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-OXGLOS] Microsoft Corporation, "[Exchange Server Protocols Master Glossary](#)", April 2008.

1.3 Overview

The Tasks Web Service protocol provides clients with the ability to create, update, and delete task items on the server. Clients create task items by using the **CreateItem** operation, as specified in [\[MS-OXWSCORE\]](#) section 3.1.4.2, or get properties of an existing task item by using the **GetItem** operation, as specified in [\[MS-OXWSCORE\]](#) section 3.1.4.4. Tasks can also be updated, deleted, or copied on the server by the using **UpdateItem** operation ([\[MS-OXWSCORE\]](#) section 3.1.4.7), the **DeleteItem** operation ([\[MS-OXWSCORE\]](#) section 3.1.4.3), and the **CopyItem** operation ([\[MS-OXWSCORE\]](#) section 3.1.4.1), respectively.

1.4 Relationship to Other Protocols

The Tasks Web Service protocol uses SOAP over **HTTP** and SOAP over **HTTPS**, as shown in the following figures.

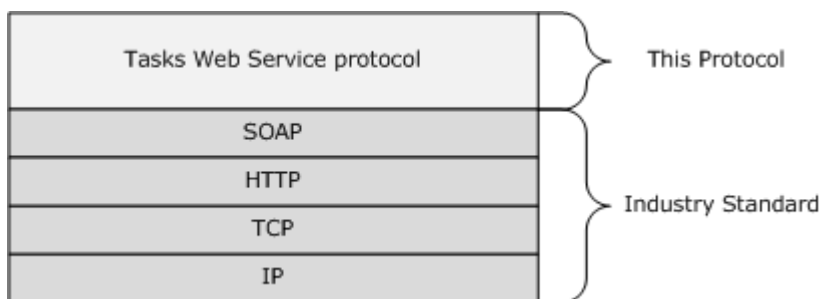


Figure 1: Tasks Web Service protocol HTTP stack

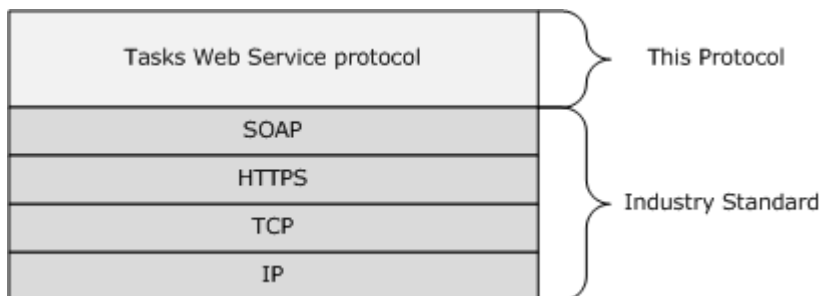


Figure 2: Tasks Web Service protocol HTTPS stack

The task information that is returned by the Tasks Web Service protocol is used when requests are made by using the Core Items Web Service protocol [\[MS-OXWSCORE\]](#).

1.5 Prerequisites/Preconditions

None.

1.6 Applicability Statement

The protocol specified in this document is applicable to environments that create, delete, and update task items.

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- **Supported Transports:** This protocol uses SOAP 1.1, as specified in section [2.1](#).
- **Protocol Versions:** This protocol specifies only one **WSDL port type** version.
- **Security and Authentication Methods:** This protocol relies on the Web server that is hosting it to perform authentication.
- **Localization:** This protocol includes text strings in various messages. Localization considerations for such strings are specified in section [3.1.4](#).
- **Capability Negotiation:** None.

1.8 Vendor Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

The SOAP version supported is SOAP 1.1. For details, see [\[SOAP1.1\]](#).

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 for each XML namespace that is used, the choice of any particular XML namespace prefix is implementation-specific and not significant for interoperability.

Prefix	Namespace URI	Reference
soap	http://schemas.xmlsoap.org/wsdl/soap/	[SOAP1.1]
tns	http://schemas.microsoft.com/exchange/services/2006/messages	[MS-OXWSTASK]
s	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1]
targetNamespace	http://schemas.microsoft.com/exchange/services/2006/messages	[MS-OXWSTASK]
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
t	http://schemas.microsoft.com/exchange/services/2006/types	[MS-OXWSTASK]

2.2.2 Simple Types

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

Simple Type	Description
t:TaskDelegateStateType	Specifies the status types of a delegated task. This enumeration is never set.
t:TaskStatusType	Specifies the status types of a task item.

2.2.2.1 t:TaskDelegateStateType Simple Type

The [TaskDelegateStateType](#) simple type specifies the status types of a delegated task. The values for this simple type are never set.

```
<xs:simpleType name="TaskDelegateStateType">
```



```

<xs:restriction
  base="xs:string"
>
  <xs:enumeration
    value="Accepted"
  />
  <xs:enumeration
    value="Declined"
  />
  <xs:enumeration
    value="Max"
  />
  <xs:enumeration
    value="NoMatch"
  />
  <xs:enumeration
    value="Owned"
  />
  <xs:enumeration
    value="OwnNew"
  />
</xs:restriction>
</xs:simpleType>

```

Enumeration

The following values are defined by the **TaskDelegateStateType** simple type:

Value	Description
Accepted	Specifies that the task has been declined.
Declined	Not used.
Max	Not used.
NoMatch	Specifies that this is not a delegated task or that the task request has been created but not sent. This is also used for a task request message, whether in the owner's Sent Items folder or the delegate's Inbox.
Owned	Specifies that a task has been accepted. This value cannot be in the simple type.
OwnNew	Specifies that this is a new task request that has been sent, but the delegate has not yet responded to the task.

2.2.2.2 t:TaskStatusType Simple Type

The [TaskStatusType](#) simple type specifies the status of a task item.

```

<xs:simpleType name="TaskStatusType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="Completed"
    />
  </xs:restriction>
</xs:simpleType>

```

```

    />
    <xs:enumeration
      value="Deferred"
    />
    <xs:enumeration
      value="InProgress"
    />
    <xs:enumeration
      value="NotStarted"
    />
    <xs:enumeration
      value="WaitingOnOthers"
    />
  </xs:restriction>
</xs:simpleType>

```

Enumeration

The following values are defined by the **TaskStatusType** simple type:

Value	Description
Completed	Specifies that the task is completed.
Deferred	Specifies that the task is deferred.
InProgress	Specifies that the task is in progress.
NotStarted	Specifies that the task is not started.
WaitingOnOthers	Specifies that the task is waiting on others.

2.2.3 Complex Types

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

Complex Type	Description
t:TaskRecurrenceType	Specifies the recurrence pattern for tasks.
t:TasksFolderType	Specifies a Tasks folder that is contained in a mailbox .
t:TaskType	Specifies a task in the server store.
t:DailyRegeneratingPatternType	Specifies the frequency, in days, at which a task is regenerated.
t:MonthlyRegeneratingPatternType	Specifies the interval, in months, at which a task is regenerated.
t:RegeneratingPatternBaseType	Specifies the base type for all regenerating patterns.
t:WeeklyRegeneratingPatternType	Specifies the interval, in weeks, at which a task is regenerated.
t:YearlyRegeneratingPatternType	Specifies the interval, in years, at which a task is regenerated.

2.2.3.1 t:TaskRecurrenceType Complex Type

The [TaskRecurrenceType](#) complex type specifies the recurrence pattern for tasks.

```
<xs:complexType name="TaskRecurrenceType">
  <xs:sequence>
    <xs:group
      ref="t:TaskRecurrencePatternTypes"
    />
    <xs:group
      ref="t:RecurrenceRangeTypes"
    />
  </xs:sequence>
</xs:complexType>
```

2.2.3.2 t:TasksFolderType Complex Type

The **TasksFolderType** complex type specifies a Tasks folder that is contained in a mailbox. The **TasksFolderType** complex type extends the [FolderType](#) complex type ([\[MS-OXWSFOLD\]](#) section 2.2.4.10).

```
<xs:complexType name="TasksFolderType">
  <xs:complexContent>
    <xs:extension
      base="t:FolderType"
    />
  </xs:complexContent>
</xs:complexType>
```

2.2.3.3 t:TaskType Complex Type

The **TaskType** complex type specifies a task in the server store. The **TaskType** complex type extends the [ItemType](#) complex type ([\[MS-OXWSCORE\]](#) section 2.2.3.6).

```
<xs:complexType name="TaskType">
  <xs:complexContent>
    <xs:extension
      base="t:ItemType"
    >
    <xs:sequence>
      <xs:element name="ActualWork"
        type="xs:int"
        minOccurs="0"
      />
      <xs:element name="AssignedTime"
        type="xs:dateTime"
        minOccurs="0"
      />
      <xs:element name="BillingInformation"
        type="xs:string"
        minOccurs="0"
      />
      <xs:element name="ChangeCount"
```

```

    type="xs:int"
    minOccurs="0"
  />
<xs:element name="Companies"
  type="t:ArrayOfStringsType"
  minOccurs="0"
/>
<xs:element name="CompleteDate"
  type="xs:dateTime"
  minOccurs="0"
/>
<xs:element name="Contacts"
  type="t:ArrayOfStringsType"
  minOccurs="0"
/>
<xs:element name="DelegationState"
  type="t:TaskDelegateStateType"
  minOccurs="0"
/>
<xs:element name="Delegator"
  type="xs:string"
  minOccurs="0"
/>
<xs:element name="DueDate"
  type="xs:dateTime"
  minOccurs="0"
/>
<xs:element name="IsAssignmentEditable"
  type="xs:int"
  minOccurs="0"
/>
<xs:element name="IsComplete"
  type="xs:boolean"
  minOccurs="0"
/>
<xs:element name="IsRecurring"
  type="xs:boolean"
  minOccurs="0"
/>
<xs:element name="IsTeamTask"
  type="xs:boolean"
  minOccurs="0"
/>
<xs:element name="Mileage"
  type="xs:string"
  minOccurs="0"
/>
<xs:element name="Owner"
  type="xs:string"
  minOccurs="0"
/>
<xs:element name="PercentComplete"
  type="xs:double"
  minOccurs="0"
/>
<xs:element name="Recurrence"
  type="t:TaskRecurrenceType"
  minOccurs="0"
/>

```

```

<xs:element name="StartDate"
  type="xs:dateTime"
  minOccurs="0"
/>
<xs:element name="Status"
  type="t:TaskStatusType"
  minOccurs="0"
/>
<xs:element name="StatusDescription"
  type="xs:string"
  minOccurs="0"
/>
<xs:element name="TotalWork"
  type="xs:int"
  minOccurs="0"
/>
</xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>

```

Child Elements

Element	Type	Description
ActualWork	xs:int	Specifies an integer value that specifies the actual amount of time that is spent on a task.
AssignedTime	xs:dateTime	Specifies an instance of the DateTime structure that contains the time when a task is assigned to a contact.
BillingInformation	xs:string	Specifies a string value that contains billing information for a task.
ChangeCount	xs:int	Specifies an integer value that specifies the version of a task.
Companies	t:ArrayOfStringsType	Specifies an instance of an array of type string that represents a collection of companies that are associated with a task.
CompleteDate	xs:dateTime	Specifies an instance of the DateTime structure that represents the date on which a task was completed.
Contacts	t:ArrayOfStringsType	Specifies an instance of an array of type string that contains a list of contacts that are associated with a task.
DelegationState	t:TaskDelegateStateType	Specifies one of the valid TaskDelegateStateType simple type (section 2.2.2.1) enumeration values that represent the status of a delegated task.
Delegator	xs:string	Specifies a string value that contains the name of the delegator who assigned the task.

Element	Type	Description
DueDate	xs:dateTime	Specifies an instance of the DateTime structure that represents the date when the task is due.
IsAssignmentEditable	xs:int	Specifies an integer value that represents the task type.
IsComplete	xs:boolean	Specifies a Boolean value that indicates whether the task has been completed.
IsRecurring	xs:boolean	Specifies a Boolean value that indicates whether the task is part of a recurring task.
IsTeamTask	xs:boolean	Specifies a Boolean value that indicates whether the task is owned by a team.
Mileage	xs:string	Specifies a string value that represents the mileage for a task.
Owner	xs:string	Specifies a string value that represents the owner of the task.
PercentComplete	xs:double	Specifies a double value from 0 to 100 that describes the completion status of a task.
Recurrence	t:TaskRecurrenceType	Specifies an instance of the TaskRecurrenceType complex type (section 2.2.2.1) that contains recurrence information for a recurring task.
StartDate	xs:dateTime	Specifies an instance of the DateTime structure that represents the start date of the task.
Status	t:TaskStatusType	Specifies one of the valid TaskStatusType simple type (section 2.2.2.2) enumeration values that represent the status of a task.
StatusDescription	xs:string	Specifies a string value that contains an explanation of the status of the task.
TotalWork	xs:int	Specifies an integer value that represents the total amount of work that is associated with the task.

2.2.3.4 t:DailyRegeneratingPatternType Complex Type

The **DailyRegeneratingPatternType** complex type specifies the frequency, in days, at which a task is regenerated. The **DailyRegeneratingPatternType** complex type extends the [RegeneratingPatternBaseType](#) complex type (section [2.2.3.6](#)).

```
<xs:complexType name="DailyRegeneratingPatternType">
  <xs:complexContent>
    <xs:extension
      base="t:RegeneratingPatternBaseType"
    />
  </xs:complexContent>
</xs:complexType>
```

2.2.3.5 t:MonthlyRegeneratingPatternType Complex Type

The **MonthlyRegeneratingPatternType** complex type specifies the interval, in months, at which a task is regenerated. The **MonthlyRegeneratingPatternType** complex type extends the [RegeneratingPatternBaseType](#) complex type (section 2.2.3.6).

```
<xs:complexType name="MonthlyRegeneratingPatternType">
  <xs:complexContent>
    <xs:extension
      base="t:RegeneratingPatternBaseType"
    />
  </xs:complexContent>
</xs:complexType>
```

2.2.3.6 t:RegeneratingPatternBaseType Complex Type

The **RegeneratingPatternBaseType** complex type specifies the base type for all regenerating patterns. The **RegeneratingPatternBaseType** complex type extends the [IntervalRecurrencePatternBaseType](#) complex type ([\[MS-OXWSCDATA\]](#) section 2.2.3.33).

```
<xs:complexType name="RegeneratingPatternBaseType"
  abstract="true"
>
  <xs:complexContent>
    <xs:extension
      base="t:IntervalRecurrencePatternBaseType"
    />
  </xs:complexContent>
</xs:complexType>
```

2.2.3.7 t:WeeklyRegeneratingPatternType Complex Type

The **WeeklyRegeneratingPatternType** complex type specifies the interval, in weeks, at which a task is regenerated. The **WeeklyRegeneratingPatternType** complex type extends the [RegeneratingPatternBaseType](#) complex type (section 2.2.3.6).

```
<xs:complexType name="WeeklyRegeneratingPatternType">
  <xs:complexContent>
    <xs:extension
      base="t:RegeneratingPatternBaseType"
    />
  </xs:complexContent>
</xs:complexType>
```

2.2.3.8 t:YearlyRegeneratingPatternType Complex Type

The **YearlyRegeneratingPatternType** complex type specifies the interval, in years, at which a task is regenerated. The **YearlyRegeneratingPatternType** complex type extends the **RegeneratingPatternBaseType** complex type (section 2.2.3.6).

```

<xs:complexType name="YearlyRegeneratingPatternType">
  <xs:complexContent>
    <xs:extension
      base="t:RegeneratingPatternBaseType"
    />
  </xs:complexContent>
</xs:complexType>

```

2.2.4 Elements

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

2.2.5 Attributes

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

2.2.6 Groups

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

Group	Description
t:TaskRecurrencePatternTypes	Specifies recurrence information for recurring tasks.

2.2.6.1 TaskRecurrencePatternTypes Group

The [TaskRecurrencePatternTypes](#) group specifies recurrence information for recurring tasks.

```

<xs:group name="TaskRecurrencePatternTypes">
  <xs:sequence>
    <xs:choice>
      <xs:element name="RelativeYearlyRecurrence"
        type="t:RelativeYearlyRecurrencePatternType"
      />
      <xs:element name="AbsoluteYearlyRecurrence"
        type="t:AbsoluteYearlyRecurrencePatternType"
      />
      <xs:element name="RelativeMonthlyRecurrence"
        type="t:RelativeMonthlyRecurrencePatternType"
      />
      <xs:element name="AbsoluteMonthlyRecurrence"
        type="t:AbsoluteMonthlyRecurrencePatternType"
      />
      <xs:element name="WeeklyRecurrence"
        type="t:WeeklyRecurrencePatternType"
      />
      <xs:element name="DailyRecurrence"
        type="t:DailyRecurrencePatternType"
      />
      <xs:element name="DailyRegeneration"
        type="t:DailyRegeneratingPatternType"
      />
    </xs:choice>
  </xs:sequence>
</xs:group>

```



```

<xs:element name="WeeklyRegeneration"
  type="t:WeeklyRegeneratingPatternType"
  />
<xs:element name="MonthlyRegeneration"
  type="t:MonthlyRegeneratingPatternType"
  />
<xs:element name="YearlyRegeneration"
  type="t:YearlyRegeneratingPatternType"
  />
</xs:choice>
</xs:sequence>
</xs:group>

```

Child Elements

Element	Type	Description
RelativeYearlyRecurrence	t:RelativeYearlyRecurrencePatternType	Specifies a relative yearly recurrence pattern.
AbsoluteYearlyRecurrence	t:AbsoluteYearlyRecurrencePatternType	Specifies a yearly recurrence pattern.
RelativeMonthlyRecurrence	t:RelativeMonthlyRecurrencePatternType	Specifies a relative monthly recurrence pattern for a recurring task item.
AbsoluteMonthlyRecurrence	t:AbsoluteMonthlyRecurrencePatternType	Specifies a monthly recurrence pattern.
WeeklyRecurrence	t:WeeklyRecurrencePatternType	Specifies the frequency, in weeks, at which and the days on which that a task item recurs.
DailyRecurrence	t:DailyRecurrencePatternType	Specifies the frequency, in days, at which a task item recurs.
DailyRegeneration	t:DailyRegeneratingPatternType	Specifies how many days after the completion of the current task the next occurrence will be due.
WeeklyRegeneration	t:WeeklyRegeneratingPatternType	Specifies how many weeks after the completion of the current task the next occurrence will be due.
MonthlyRegeneration	t:MonthlyRegeneratingPatternType	Specifies how many months after the completion of the current task the next occurrence will be due.

Element	Type	Description
YearlyRegeneration	t:YearlyRegeneratingPatternType	Specifies how many years after the completion of the current task the next occurrence will be due.

2.2.7 Attribute Groups

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

2.2.8 Messages

This specification does not define any common **XML schema** message definitions.

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 ExchangeServicePortType Server Details

The Tasks Web Service protocol defines a single port type.

Operation	Description
GetItem	Gets items in the server.
CreateItem	Creates items in the server.
DeleteItem	Deletes items in the server.
UpdateItem	Updates items in the server.
MoveItem	Moves items in the server.
CopyItem	Copies items in the server.

3.1.1 Abstract Data Model

The Tasks Web Service protocol is a stateless protocol.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing

This protocol includes the operations that are listed in the following table.

Operation	Description
GetItem	Gets items on the server.
CreateItem	Creates items on the server.
DeleteItem	Deletes items on the server.
UpdateItem	Updates items on the server.
MoveItem	Moves items on the server.
CopyItem	Copies items on the server.

3.1.4.1 CopyItem

This protocol uses the **CopyItem** operation ([\[MS-OXWSCORE\]](#) section 3.1.4.1) to copy task item elements.

```
<wsdl:operation name="CopyItem">
  <wsdl:input message="tns:CopyItemSoapIn" />
  <wsdl:output message="tns:CopyItemSoapOut" />
</wsdl:operation>
```

Request

Message Format	Description
tns:CopyItemSoapIn	Specifies the SOAP message that defines the task item to copy. The t:NonEmptyArrayOfAllItemsType complex type ([MS-OXWSCORE] section 2.2.3.38) of the CopyItem operation ([MS-OXWSCORE] section 3.1.4.1) that specifies the XML request MUST contain the t:TargetFolderIdType ([MS-OXWSFOLD] section 2.2.4.14) and t:ItemIdType ([MS-OXWSCORE] section 2.2.3.13) complex types. All other elements MUST be empty.

Response

Message Format	Description
tns:CopyItemSoapOut	Specifies the SOAP message that is returned by the server in response.

3.1.4.2 CreateItem

This protocol uses the **CreateItem** operation ([\[MS-OXWSCORE\]](#) section 3.1.4.2) to create task item elements.

```
<wsdl:operation name="CreateItem">
  <wsdl:input message="tns:CreateItemSoapIn" />
  <wsdl:output message="tns:CreateItemSoapOut" />
</wsdl:operation>
```

Request

Message Format	Description
tns:CreateItemSoapIn	Specifies the SOAP message that defines the task item to create. The t:NonEmptyArrayOfAllItemsType complex type ([MS-OXWSCORE] section 2.2.3.38) of the CreateItem operation ([MS-OXWSCORE] section 3.1.4.2) that specifies the XML request MUST contain one or more t:TaskType complex types (section 2.2.3.3). All other elements MUST be empty.

Response

Message Format	Description
tns:CreateItemSoapOut	Specifies the SOAP message that is returned by the server in response.

3.1.4.3 DeleteItem

This protocol uses the **DeleteItem** operation ([\[MS-OXWSCORE\]](#) section 3.1.4.3) to delete task item elements.

```
<wsdl:operation name="DeleteItem">
  <wsdl:input message="tns:DeleteItemSoapIn" />
  <wsdl:output message="tns:DeleteItemSoapOut" />
</wsdl:operation>
```

Request

Message Format	Description
tns:DeleteItemSoapIn	Specifies the SOAP message that defines the task item to delete. The t:NonEmptyArrayOfAllItemsType complex type ([MS-OXWSCDATA] section) of the DeleteItem operation ([MS-OXWSCORE] section 3.1.4.3) that specifies the XML request MUST contain one or more t:ItemIdType complex type ([MS-OXWSCORE] section 2.2.3.13) elements. All other elements MUST be empty.

Response

Message Format	Description
tns:DeleteItemSoapOut	Specifies the SOAP message that is returned by the server in response.

3.1.4.3.1 Simple Types

The following **XML schema** simple type definitions are specific to this operation.

3.1.4.3.1.1 t:AffectedTaskOccurrencesType Simple Type

The [AffectedTaskOccurrencesType](#) simple type specifies whether an occurrence of a task or a master task is deleted.

```
<xs:simpleType name="AffectedTaskOccurrencesType">
  <xs:restriction
    base="xs:string"
  >
    <xs:enumeration
      value="AllOccurrences"
    />
    <xs:enumeration
      value="SpecifiedOccurrenceOnly"
    />
  </xs:restriction>
</xs:simpleType>
```

Enumeration

The following values are defined by the **AffectedTaskOccurrencesType** simple type:

Value	Description
AllOccurrences	Specifies that a DeleteItem operation ([MS-OXWSCORE] section 3.1.4.3) request deletes the master task, and therefore all recurring tasks that are associated with the master task.
SpecifiedOccurrenceOnly	Specifies that a DeleteItem operation ([MS-OXWSCORE] section 3.1.4.3) request deletes only the current occurrence of a task.

3.1.4.4 GetItem

This protocol uses the **GetItem** operation ([\[MS-OXWSCORE\]](#) section 3.1.4.4) to get task item elements.

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

Request

Message Format	Description
tns:GetItemSoapIn	Specifies the SOAP message that defines the task item to get. The t:NonEmptyArrayOfAllItemsType complex type ([MS-OXWSCDATA] section 2.2.3.38) of the GetItem operation ([MS-OXWSCORE] section 3.1.4.4) that specifies the XML request MUST contain the t:ItemResponseShapeType complex type ([MS-OXWSCDATA] section 2.2.3.35) and t:ItemIdType complex type ([MS-OXWSCORE] section 2.2.3.13) elements.

Response

Message Format	Description
tns:GetItemSoapOut	Specifies the SOAP message that is returned by the server in response. The server returns a t:ItemResponseShapeType complex type ([MS-OXWSCDATA] section 2.2.3.35) element that contains properties that are associated with the task item.

3.1.4.5 MoveItem

This protocol uses the **MoveItem** operation ([\[MS-OXWSCORE\]](#) section 3.1.4.5) to move task item elements.

```
<wsdl:operation name="MoveItem">
  <wsdl:input message="tns:MoveItemSoapIn" />
  <wsdl:output message="tns:MoveItemSoapOut" />
</wsdl:operation>
```

Request

Message Format	Description
tns:MoveItemSoapIn	Specifies the SOAP message that defines the task item to move. The t:NonEmptyArrayOfAllItemsType complex type ([MS-OXWSCDATA] section) of the MoveItem operation ([MS-OXWSCORE] section 3.1.4.5) that specifies the XML request MUST contain the t:TargetFolderIdType complex type ([MS-OXWSFOLD] section 2.2.4.14) and t:ItemIdType complex type ([MS-OXWSCORE] section 2.2.3.13) elements. All other elements MUST be empty.

Response

Message Format	Description
tns:MoveItemSoapOut	Specifies the SOAP message that is returned by the server in response.

3.1.4.6 UpdateItem

This protocol uses the **UpdateItem** operation ([MS-OXWSCORE] section 3.1.4.7) to update task item elements.

```
<wsdl:operation name="UpdateItem">
  <wsdl:input message="tns:UpdateItemSoapIn" />
  <wsdl:output message="tns:UpdateItemSoapOut" />
</wsdl:operation>
```

Request

Message Format	Description
tns:UpdateItemSoapIn	Specifies the SOAP message that defines the task item to update.

Response

Message Format	Description
tns:UpdateItemSoapOut	Specifies the SOAP message that is returned by the server in response.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

3.2 Client 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 that are returned by the transport are passed directly back to the higher-layer protocol or application.

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

None.

3.2.5 Timer Events

None.

3.2.6 Other Local Events

None.

4 Protocol Examples

None.

5 Security

5.1 Security Considerations for Implementers

The Tasks Web Service protocol does not use any additional security mechanisms.

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

The following table lists the XML files that are required in order to implement the functionality that is specified in this document. The contents of each file are included in this section.

File name	Description	Section
MS-OXWSTASK.wsdl	Contains the WSDL for the implementation of this protocol.	6.1
MS-OXWSTASK-types.xsd	Contains the XML schema type definitions that are used in this protocol.	6.2
MS-OXWSTASK-messages.xsd	Contains the XML schema message definitions that are used in this protocol.	6.3

These files have to be placed in a common folder in order for the WSDL to validate and operate. Also, any schema files that are included in or imported into the MS-OXWSTASK-types.xsd or MS-OXWSTASK-messages.xsd schemas have to be placed in the common folder along with the files.

6.1 WSDL

This section contains the contents of the MS-OXWSTASK.wsdl file.

```
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:s="http://www.w3.org/2001/XMLSchema" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages">
  <wsdl:types>
    <xs:schema id="messages" elementFormDefault="qualified" version="Exchange2010"
xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns="http://schemas.microsoft.com/exchange/services/2006/messages">
      <xs:import
namespace="http://schemas.microsoft.com/exchange/services/2006/types" schemaLocation="MS-
OXWSCORE-types.xsd"/>
      <xs:include schemaLocation="MS-OXWSCORE-messages.xsd"/>
      <!-- Add global elements and types from messages.xsd -->
    </xs:schema>
    <xs:schema id="types" elementFormDefault="qualified" version="Exchange2010"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
      <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
      <!-- Add global elements and types from types.xsd -->
    </xs:schema>
  </wsdl:types>
  <wsdl:portType name="ExchangeServicePortType">
    <wsdl:operation name="GetItem">
      <wsdl:input message="tns:GetItemSoapIn"/>
      <wsdl:output message="tns:GetItemSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="CreateItem">
```

```

        <wsdl:input message="tns:CreateItemSoapIn"/>
        <wsdl:output message="tns:CreateItemSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="DeleteItem">
        <wsdl:input message="tns:DeleteItemSoapIn"/>
        <wsdl:output message="tns:DeleteItemSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="UpdateItem">
        <wsdl:input message="tns:UpdateItemSoapIn"/>
        <wsdl:output message="tns:UpdateItemSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="MoveItem">
        <wsdl:input message="tns:MoveItemSoapIn"/>
        <wsdl:output message="tns:MoveItemSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="CopyItem">
        <wsdl:input message="tns:CopyItemSoapIn"/>
        <wsdl:output message="tns:CopyItemSoapOut"/>
    </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="ExchangeServiceBinding" type="tns:ExchangeServicePortType">
    <wsdl:documentation>
        <wsi:Claim conformsTo="http://ws-i.org/profiles/basic/1.0"
xmlns:wsi="http://ws-i.org/schemas/conformanceClaim/" />
    </wsdl:documentation>
    <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="GetItem">
        <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/GetItem"/>
        <wsdl:input>
            <soap:header message="tns:GetItemSoapIn" part="Impersonation"
use="literal"/>
            <soap:header message="tns:GetItemSoapIn" part="MailboxCulture"
use="literal"/>
            <soap:header message="tns:GetItemSoapIn" part="RequestVersion"
use="literal"/>
            <soap:header message="tns:GetItemSoapIn" part="TimeZoneContext"
use="literal"/>
            <soap:body parts="request" use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body parts="GetItemResult" use="literal"/>
            <soap:header message="tns:GetItemSoapOut" part="ServerVersion"
use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="CreateItem">
        <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/CreateItem"/>
        <wsdl:input>
            <soap:header message="tns:CreateItemSoapIn" part="Impersonation"
use="literal"/>
            <soap:header message="tns:CreateItemSoapIn" part="MailboxCulture"
use="literal"/>
            <soap:header message="tns:CreateItemSoapIn" part="RequestVersion"
use="literal"/>
            <soap:header message="tns:CreateItemSoapIn" part="TimeZoneContext"
use="literal"/>
            <soap:body parts="request" use="literal"/>
        </wsdl:input>

```

```

        <wsdl:output>
            <soap:body parts="CreateItemResult" use="literal"/>
            <soap:header message="tns:CreateItemSoapOut" part="ServerVersion"
use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="DeleteItem">
        <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/DeleteItem"/>
        <wsdl:input>
            <soap:header message="tns:DeleteItemSoapIn" part="Impersonation"
use="literal"/>
            <soap:header message="tns:DeleteItemSoapIn" part="MailboxCulture"
use="literal"/>
            <soap:header message="tns:DeleteItemSoapIn" part="RequestVersion"
use="literal"/>
            <soap:body parts="request" use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body parts="DeleteItemResult" use="literal"/>
            <soap:header message="tns:DeleteItemSoapOut" part="ServerVersion"
use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="UpdateItem">
        <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/UpdateItem"/>
        <wsdl:input>
            <soap:header message="tns:UpdateItemSoapIn" part="Impersonation"
use="literal"/>
            <soap:header message="tns:UpdateItemSoapIn" part="MailboxCulture"
use="literal"/>
            <soap:header message="tns:UpdateItemSoapIn" part="RequestVersion"
use="literal"/>
            <soap:header message="tns:UpdateItemSoapIn" part="TimeZoneContext"
use="literal"/>
            <soap:body parts="request" use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body parts="UpdateItemResult" use="literal"/>
            <soap:header message="tns:UpdateItemSoapOut" part="ServerVersion"
use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="MoveItem">
        <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/MoveItem"/>
        <wsdl:input>
            <soap:header message="tns:MoveItemSoapIn" part="Impersonation"
use="literal"/>
            <soap:header message="tns:MoveItemSoapIn" part="MailboxCulture"
use="literal"/>
            <soap:header message="tns:MoveItemSoapIn" part="RequestVersion"
use="literal"/>
            <soap:body parts="request" use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body parts="MoveItemResult" use="literal"/>
            <soap:header message="tns:MoveItemSoapOut" part="ServerVersion"
use="literal"/>
        </wsdl:output>
    </wsdl:operation>

```

```

        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="CopyItem">
        <soap:operation
soapAction="http://schemas.microsoft.com/exchange/services/2006/messages/CopyItem"/>
        <wsdl:input>
            <soap:header message="tns:CopyItemSoapIn" part="Impersonation"
use="literal"/>
            <soap:header message="tns:CopyItemSoapIn" part="MailboxCulture"
use="literal"/>
            <soap:header message="tns:CopyItemSoapIn" part="RequestVersion"
use="literal"/>
            <soap:body parts="request" use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body parts="CopyItemResult" use="literal"/>
            <soap:header message="tns:CopyItemSoapOut" part="ServerVersion"
use="literal"/>
        </wsdl:output>
    </wsdl:operation>
</wsdl:binding>
<wsdl:message name="GetItemSoapIn">
    <wsdl:part name="request" element="tns:GetItem"/>
    <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
    <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
    <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
    <wsdl:part name="TimeZoneContext" element="t:TimeZoneContext"/>
</wsdl:message>
<wsdl:message name="GetItemSoapOut">
    <wsdl:part name="GetItemResult" element="tns:GetItemResponse"/>
    <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
<wsdl:message name="CreateItemSoapIn">
    <wsdl:part name="request" element="tns:CreateItem"/>
    <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
    <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
    <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
    <wsdl:part name="TimeZoneContext" element="t:TimeZoneContext"/>
</wsdl:message>
<wsdl:message name="CreateItemSoapOut">
    <wsdl:part name="CreateItemResult" element="tns:CreateItemResponse"/>
    <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
<wsdl:message name="DeleteItemSoapIn">
    <wsdl:part name="request" element="tns>DeleteItem"/>
    <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
    <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
    <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
</wsdl:message>
<wsdl:message name="DeleteItemSoapOut">
    <wsdl:part name="DeleteItemResult" element="tns>DeleteItemResponse"/>
    <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
<wsdl:message name="UpdateItemSoapIn">
    <wsdl:part name="request" element="tns:UpdateItem"/>
    <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
    <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
    <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
    <wsdl:part name="TimeZoneContext" element="t:TimeZoneContext"/>

```

```

</wsdl:message>
<wsdl:message name="UpdateItemSoapOut">
  <wsdl:part name="UpdateItemResult" element="tns:UpdateItemResponse"/>
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
<wsdl:message name="MoveItemSoapIn">
  <wsdl:part name="request" element="tns:MoveItem"/>
  <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
  <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
  <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
</wsdl:message>
<wsdl:message name="MoveItemSoapOut">
  <wsdl:part name="MoveItemResult" element="tns:MoveItemResponse"/>
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
<wsdl:message name="CopyItemSoapIn">
  <wsdl:part name="request" element="tns:CopyItem"/>
  <wsdl:part name="Impersonation" element="t:ExchangeImpersonation"/>
  <wsdl:part name="MailboxCulture" element="t:MailboxCulture"/>
  <wsdl:part name="RequestVersion" element="t:RequestServerVersion"/>
</wsdl:message>
<wsdl:message name="CopyItemSoapOut">
  <wsdl:part name="CopyItemResult" element="tns:CopyItemResponse"/>
  <wsdl:part name="ServerVersion" element="t:ServerVersionInfo"/>
</wsdl:message>
</wsdl:definitions>

```

6.2 Types Schema

This section contains the contents of the MS-OXWSTASK-types.xsd file and information about additional files that this schema file requires in order to operate correctly.

MS-OXWSTASK-types.xsd includes the file listed in the following table. To operate correctly, this file has to be present in the folder that contains the WSDL, types, and messages schema files for this protocol.

File name	Defining specification
MS-OXWSCFOLD-types.xsd	[MS-OXWSFOLD] section 6.2

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/types"
elementFormDefault="qualified" version="Exchange2010" id="types">
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"/>
  <xs:include schemaLocation="MS-OXWSCFOLD-types.xsd"/>
  <xs:simpleType name="AffectedTaskOccurrencesType">
    <xs:restriction base="xs:string">
      <xs:enumeration value="AllOccurrences"/>
      <xs:enumeration value="SpecifiedOccurrenceOnly"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:complexType name="RegeneratingPatternBaseType" abstract="true">

```

```

        <xs:complexContent>
            <xs:extension base="t:IntervalRecurrencePatternBaseType"/>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="DailyRegeneratingPatternType">
        <xs:complexContent>
            <xs:extension base="t:RegeneratingPatternBaseType"/>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="WeeklyRegeneratingPatternType">
        <xs:complexContent>
            <xs:extension base="t:RegeneratingPatternBaseType"/>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="MonthlyRegeneratingPatternType">
        <xs:complexContent>
            <xs:extension base="t:RegeneratingPatternBaseType"/>
        </xs:complexContent>
    </xs:complexType>
    <xs:complexType name="YearlyRegeneratingPatternType">
        <xs:complexContent>
            <xs:extension base="t:RegeneratingPatternBaseType"/>
        </xs:complexContent>
    </xs:complexType>
    <xs:simpleType name="TaskStatusType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="NotStarted"/>
            <xs:enumeration value="InProgress"/>
            <xs:enumeration value="Completed"/>
            <xs:enumeration value="WaitingOnOthers"/>
            <xs:enumeration value="Deferred"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:simpleType name="TaskDelegateStateType">
        <xs:restriction base="xs:string">
            <xs:enumeration value="NoMatch"/>
            <xs:enumeration value="OwnNew"/>
            <xs:enumeration value="Owned"/>
            <xs:enumeration value="Accepted"/>
            <xs:enumeration value="Declined"/>
            <xs:enumeration value="Max"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:complexType name="TaskType">
        <xs:complexContent>
            <xs:extension base="t:ItemType">
                <xs:sequence>
                    <xs:element name="ActualWork" type="xs:int" minOccurs="0"/>
                    <xs:element name="AssignedTime" type="xs:dateTime" minOccurs="0"/>
                    <xs:element name="BillingInformation" type="xs:string"
minOccurs="0"/>
                    <xs:element name="ChangeCount" type="xs:int" minOccurs="0"/>
                    <xs:element name="Companies" type="t:ArrayOfStringsType"
minOccurs="0"/>
                    <xs:element name="CompleteDate" type="xs:dateTime" minOccurs="0"/>
                    <xs:element name="Contacts" type="t:ArrayOfStringsType"
minOccurs="0"/>
                    <xs:element name="DelegationState" type="t:TaskDelegateStateType"
minOccurs="0"/>
                </xs:sequence>
            </xs:extension>
        </xs:complexContent>
    </xs:complexType>

```



```

        <xs:element name="Delegator" type="xs:string" minOccurs="0"/>
        <xs:element name="DueDate" type="xs:dateTime" minOccurs="0"/>
        <xs:element name="IsAssignmentEditable" type="xs:int"
minOccurs="0"/>
        <xs:element name="IsComplete" type="xs:boolean" minOccurs="0"/>
        <xs:element name="IsRecurring" type="xs:boolean" minOccurs="0"/>
        <xs:element name="IsTeamTask" type="xs:boolean" minOccurs="0"/>
        <xs:element name="Mileage" type="xs:string" minOccurs="0"/>
        <xs:element name="Owner" type="xs:string" minOccurs="0"/>
        <xs:element name="PercentComplete" type="xs:double" minOccurs="0"/>
        <xs:element name="Recurrence" type="t:TaskRecurrenceType"
minOccurs="0"/>
        <xs:element name="StartDate" type="xs:dateTime" minOccurs="0"/>
        <xs:element name="Status" type="t:TaskStatusType" minOccurs="0"/>
        <xs:element name="StatusDescription" type="xs:string"
minOccurs="0"/>
        <xs:element name="TotalWork" type="xs:int" minOccurs="0"/>
    </xs:sequence>
</xs:extension>
</xs:complexContent>
</xs:complexType>
<xs:complexType name="TaskRecurrenceType">
    <xs:sequence>
        <xs:group ref="t:TaskRecurrencePatternTypes"/>
        <xs:group ref="t:RecurrenceRangeTypes"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="TasksFolderType">
    <xs:complexContent>
        <xs:extension base="t:FolderType"/>
    </xs:complexContent>
</xs:complexType>
</xs:schema>

```

6.3 Messages Schema

This section contains the contents of the MS-OXWSTASK-messages.xsd file.

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:m="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:tns="http://schemas.microsoft.com/exchange/services/2006/messages"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
targetNamespace="http://schemas.microsoft.com/exchange/services/2006/messages"
elementFormDefault="qualified" version="Exchange2010" id="messages">
    <!-- CHANGE THE SCHEMA LOCATION TO REFLECT THE TYPES XSD ASSOCIATED WITH THIS
DOCUMENT
    <xs:import namespace="http://schemas.microsoft.com/exchange/services/2006/types"
schemaLocation="types.xsd"/>
    <xs:include/>
-->
</xs:schema>

```

7 Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products:

- Microsoft® Exchange Server 2010

Exceptions, if any, are noted below. If a service pack number appears with the product version, behavior changed in that service pack. 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 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 product does not follow the prescription.

8 Change Tracking

This section identifies changes that were made to the [MS-OXWSTASK] protocol document between the May 2010 and August 2010 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
1.2.1 Normative References	55751 Moved [MS-OXGLOS] from the Normative References section to the Informative References section.	N	Content update.
1.2.1 Normative References	Removed the references [RFC2396], [RFC2616], [RFC2818], and [RFC3066].	N	Content update.

9 Index

A

[Applicability](#) 7

C

[Capability negotiation](#) 7

[Change tracking](#) 35

Client

[abstract data model](#) 24

[initialization](#) 24

[local events](#) 24

[message processing](#) 24

[overview](#) 23

[sequencing rules](#) 24

[timer events](#) 24

[timers](#) 24

F

[Full WSDL](#) 27

G

[Glossary](#) 5

I

[Introduction](#) 5

M

Messages

[overview](#) 8

[syntax](#) 8

[transport](#) 8

O

[Overview \(synopsis\)](#) 6

P

[Preconditions](#) 7

[Prerequisites](#) 7

[Product behavior](#) 34

R

References

[informative](#) 6

[normative](#) 5

[Relationship to other protocols](#) 6

S

Security

[implementer considerations](#) 26

[parameter index](#) 26

Server

[abstract data model](#) 19

[initialization](#) 19

[local events](#) 23

[message processing](#) 19

overview ([section 3.1](#) 19, [section 5](#) 26)

[sequencing rules](#) 19

[timer events](#) 23

[timers](#) 19

[Standards assignments](#) 7

T

[Tracking changes](#) 35

V

[Vendor-extensible fields](#) 7

[Versioning](#) 7