

[MS-OMPWHP]:

Office Mobile PowerPoint Web Handler Protocol

Intellectual Property Rights Notice for Open Specifications Documentation

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

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

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

Revision Summary

Date	Revision History	Revision Class	Comments
7/13/2009	0.1	Major	Initial Availability
8/28/2009	0.2	Editorial	Revised and edited the technical content
11/6/2009	0.3	Editorial	Revised and edited the technical content
2/19/2010	1.0	Major	Updated and revised the technical content
3/31/2010	1.01	Editorial	Revised and edited the technical content
4/30/2010	1.02	Editorial	Revised and edited the technical content
6/7/2010	1.03	Editorial	Revised and edited the technical content
6/29/2010	1.04	Editorial	Changed language and formatting in the technical content.
7/23/2010	1.05	Minor	Clarified the meaning of the technical content.
9/27/2010	1.05	None	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	1.06	Editorial	Changed language and formatting in the technical content.
12/17/2010	1.06	None	No changes to the meaning, language, or formatting of the technical content.
3/18/2011	1.06	None	No changes to the meaning, language, or formatting of the technical content.
6/10/2011	1.06	None	No changes to the meaning, language, or formatting of the technical content.
1/20/2012	2.0	Major	Significantly changed the technical content.
4/11/2012	2.0	None	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	2.1	Minor	Clarified the meaning of the technical content.
10/8/2012	2.1	None	No changes to the meaning, language, or formatting of the technical content.
2/11/2013	2.1	None	No changes to the meaning, language, or formatting of the technical content.
7/30/2013	2.1	None	No changes to the meaning, language, or formatting of the technical content.
11/18/2013	2.1	None	No changes to the meaning, language, or formatting of the technical content.
2/10/2014	2.1	None	No changes to the meaning, language, or formatting of the technical content.
4/30/2014	2.2	Minor	Clarified the meaning of the technical content.
7/31/2014	2.2	None	No changes to the meaning, language, or formatting of the technical content.
10/30/2014	2.2	None	No changes to the meaning, language, or formatting of the

Date	Revision History	Revision Class	Comments
			technical content.
6/23/2016	2.2	None	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1	Introduction	6
1.1	Glossary	6
1.2	References	6
1.2.1	Normative References	7
1.2.2	Informative References	7
1.3	Protocol Overview (Synopsis)	7
1.4	Relationship to Other Protocols	7
1.5	Prerequisites/Preconditions	8
1.6	Applicability Statement	8
1.7	Versioning and Capability Negotiation	8
1.8	Vendor-Extensible Fields	8
1.9	Standards Assignments.....	8
2	Messages.....	9
2.1	Transport	9
2.2	Common Message Syntax	9
2.2.1	Namespaces	9
2.2.2	Messages.....	9
2.2.3	Elements	9
2.2.4	Complex Types.....	9
2.2.5	Simple Types	9
2.2.6	Attributes	9
2.2.7	Groups	9
2.2.8	Attribute Groups.....	9
3	Protocol Details.....	10
3.1	Server Details.....	10
3.1.1	Abstract Data Model.....	10
3.1.2	Timers	10
3.1.3	Initialization	11
3.1.4	Message Processing Events and Sequencing Rules	11
3.1.4.1	MobilePresentation	11
3.1.4.1.1	GetMetadataXml.....	11
3.1.4.1.1.1	Return Values.....	11
3.1.4.1.2	GetLatestDocumentVersion.....	12
3.1.4.1.2.1	Return Values.....	12
3.1.4.2	MobileSlideInfo	13
3.1.4.2.1	GetSlideInfoXml	13
3.1.4.2.1.1	Return Values.....	13
3.1.4.3	MobileSlideImage.....	14
3.1.4.3.1	GetSlideImage	14
3.1.4.3.1.1	Return Values.....	14
3.1.5	Timer Events.....	14
3.1.6	Other Local Events.....	14
4	Protocol Examples	15
4.1	Presentation Information.....	15
4.2	Slide Image.....	16
4.3	Slide Information	16
5	Security.....	18
5.1	Security Considerations for Implementers	18
5.2	Index of Security Parameters	18
6	Appendix A: Full WSDL	19
7	Appendix B: Product Behavior	20

8	Change Tracking	21
9	Index	22

1 Introduction

This document specifies the Office Mobile PowerPoint Web Handler Protocol, which enables a protocol client to obtain information about presentation content on a protocol server.

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

1.1 Glossary

This document uses the following terms:

HTTP GET: An HTTP method for retrieving a resource, as described in [\[RFC2616\]](#).

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

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

presentation: A collection of slides that are intended to be viewed by an audience.

presentation broadcast: A feature that enables users to run a presentation over the web. The presentation is saved in HTML format and can contain audio and video. It can also be recorded and saved for viewing later.

presentation slide: A slide that contains the content that can be displayed during a slide show. A presentation slide can derive formatting and content from a main master slide or a title master slide.

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

XML: The Extensible Markup Language, as described in [\[XML1.0\]](#).

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

XML schema: A description of a type of XML document that is typically expressed in terms of constraints on the structure and content of documents of that type, in addition to the basic syntax constraints that are imposed by **XML** itself. An XML schema provides a view of a document type at a relatively high level of abstraction.

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

1.2 References

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

1.2.1 Normative References

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

[MS-PWBHPS] Microsoft Corporation, "[PowerPoint Web Broadcast Host Protocol](#)".

[MS-PWVRSC] Microsoft Corporation, "[PowerPoint Web Viewer Rendered Static Content Structure](#)".

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

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

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

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

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

1.2.2 Informative References

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

1.3 Protocol Overview (Synopsis)

This protocol enables a protocol client to send a request to retrieve **presentation** content from the protocol server.

To facilitate this, the protocol allows the protocol client to request specific pieces of content from a presentation stored on the protocol server. In a **presentation slide** contained within a presentation, a protocol client can retrieve information describing the presentation slide contents as well as images of the presentation slide contents.

1.4 Relationship to Other Protocols

This protocol uses **Hypertext Transfer Protocol (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:

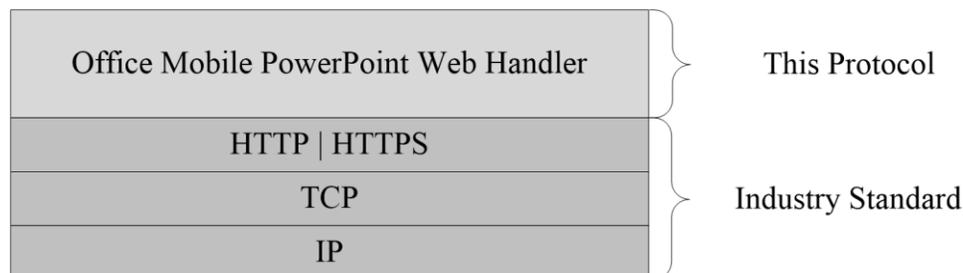


Figure 1: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a site that is identified by a **URL** that is known by protocol clients.

The protocol server endpoint to send requests for the presentation metadata is formed by appending "/_layouts/MobilePresentation.ashx" to the URL of the site, for example:
http://www.contoso.com/Repository/_layouts/MobilePresentation.ashx.

The protocol server endpoint to send requests for the presentation slide information is formed by appending the value of the **url** attribute of the **CT_SlideInfo** element ([\[MS-PWVRSC\]](#) section 2.3.11) contained in the metadata for the requested presentation to the URL of the site, for example:
http://www.contoso.com/Repository/_layouts/MobileSlideInfo.ashx.

The protocol server endpoint to send requests for the presentation slide image is formed by appending the value of the **url** attribute of the **CT_SlideImage** element ([\[MS-PWVRSC\]](#) section 2.3.10) contained in the metadata for the requested presentation to the URL of the site, for example:
http://www.contoso.com/Repository/_layouts/MobileSlideImage.ashx.

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

1.6 Applicability Statement

This protocol is designed to retrieve information about presentations that are stored on the protocol server.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

This protocol uses HTTP(S) version 1.1 (as specified in [\[RFC2616\]](#)) as transport for the **HTTP GET** method.

2.2 Common Message Syntax

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

2.2.1 Namespaces

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

Prefix	Namespace URI	Reference
xsd	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1] [XMLSCHEMA2]

2.2.2 Messages

None.

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

This specification does not define any common XML Schema simple type definitions.

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.

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.

Except where specified, protocol clients SHOULD interpret HTTP status codes returned by the protocol server as specified in [\[RFC2616\]](#) section 10.

This protocol allows protocol servers to perform implementation-specific authorization checks and notify protocol clients of authorization faults using HTTP status codes.

3.1 Server Details

The following high-level sequence diagram illustrates the operation of the protocol.

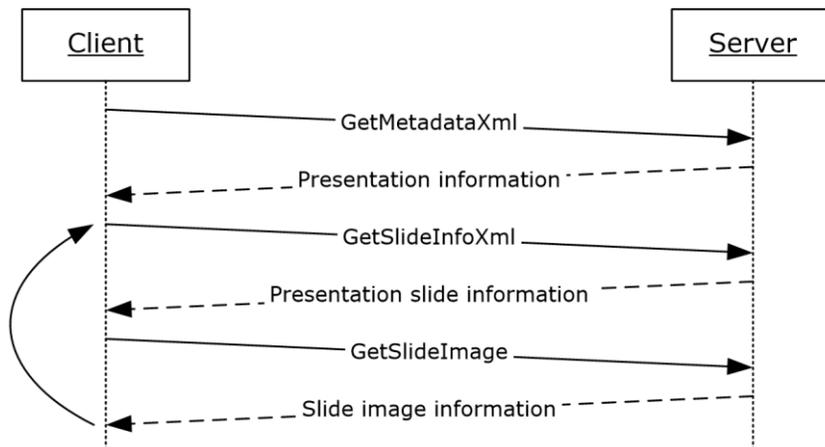


Figure 2: Sample communication between protocol client and protocol server

First, a protocol client calls **GetMetadataXml** and the protocol server responds with the information about the requested presentation.

The protocol client, using the preceding information, makes one or more calls to **GetSlideInfoXml**, to retrieve information about the requested presentation slide, and **GetSlideImage**, to retrieve the image of the requested presentation slide.

3.1.1 Abstract Data Model

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

presentationId: An entity which represents a unique identifier for a presentation.

3.1.2 Timers

None.

3.1.3 Initialization

The protocol server MUST expose its Web methods at a URL, which builds upon a base URL.

The URL MUST be constructed as base URL/_layouts/handler name.

The base URL can be any Web site URL, such as "http://www.contoso.com/Repository". The part of _layouts/handler name can be any of the following:

```
_layouts/MobilePresentation.ashx  
_layouts/MobileSlideImage.ashx  
_layouts/MobileSlideInfo.ashx
```

This is the minimal required structure. Case-sensitivity is specific to the protocol server implementation.

3.1.4 Message Processing Events and Sequencing Rules

The following table describes the supported operations.

Operation	Description
GetMetadataXml	Retrieves the metadata for a presentation.
GetLatestDocumentVersion	Retrieves the current version of a presentation.
GetSlideImage	Retrieves the image of a presentation slide.
GetSlideInfoXml	Retrieves information about content of a presentation slide.

3.1.4.1 MobilePresentation

The methods in this section are supported by MobilePresentation.ashx.

3.1.4.1.1 GetMetadataXml

This method retrieves the metadata for a presentation. In cases other than that of a **presentation broadcast**, the URL to this method is constructed by appending _layouts/MobilePresentation.ashx to the site name. The parameter to be passed while calling this method is:

presentationId: An **xsd:string** element ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the absolute or relative URL to the presentation.

Also, the scheme of the URL MUST match the scheme of the **presentationId** if the **presentationId** is being passed as an absolute URL.

In case of a presentation broadcast the URL to this method is constructed by replacing PowerpointFrame.aspx with MobilePresentation.ashx in the attendee URL. An attendee URL is contained in the **BroadcastGetAttendeeUrlResponse** defined in [\[MS-PWBHPS\]](#) section 3.1.4.2.2.2. No additional parameters need to be passed to the protocol server in this case.

If any of the preceding conditions are not satisfied, the method returns an error.

3.1.4.1.1.1 Return Values

This method MUST send back to the protocol client an HTTP response that MUST be in the following table.

Value	Description
200	<p>If successful, the protocol server returns an XML string specifying the presentation metadata, which contains a mobilePres root element as specified in [MS-PWVRSC] section 2.1.1. An X-DocVersion HTTP header is present (except in the case of a presentation broadcast) which specifies the version of the presentation. An X-DocUrl HTTP header is present (except in the case of a presentation broadcast) which specifies the URL of the presentation.</p> <p>All other cases MUST be treated as failure by the protocol client.</p>
503	<p>Failure. The server also sets an HTTP header X-Error xsd:unsignedInt ([XMLSCHEMA2] section 3.3.22) in the response. If $100 \leq \text{value} < 200$, the protocol client SHOULD retry this call, else this header MUST be ignored.</p>

3.1.4.1.2 GetLatestDocumentVersion

This method retrieves the version of the presentation. In cases other than that of a presentation broadcast, the URL to this method is constructed by appending `_layouts/MobilePresentation.ashx` to the site name. The parameters to be passed while calling this method are as follows:

presentationId: An **xsd:string** element ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the URL to the presentation. This parameter is optional. If the parameter is specified, the URL MUST be absolute or relative. If the parameter is not specified or if it is empty, then it MUST be auto generated.

getCurrentDocVersion: An **xsd:string** element ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies if the version of the document is to be returned. If this parameter is set to **true**, the server MUST return the version. For any other legal string, the server MUST ignore this parameter and treat this call as the **GetMetaDataXml** method (section [3.1.4.1.1](#)) call.

Also, the scheme of the URL MUST match the scheme of the **presentationId** if the **presentationId** is being passed as an absolute URL.

In case of a presentation broadcast, the URL to this method is constructed by replacing `PowerpointFrame.aspx` with `MobilePresentation.ashx` in the attendee URL. An attendee URL is contained in the **BroadcastGetAttendeeUrlResponse** defined in [\[MS-PWBHPS\]](#) section 3.1.4.3.2.2. Only the **getCurrentDocVersion** parameter needs to be passed to the protocol server in this case.

If any of the preceding conditions about the parameters are not satisfied, the method returns an error.

3.1.4.1.2.1 Return Values

This method MUST send back to the protocol client an HTTP response that MUST be in the following table.

Value	Description
200	If successful, the protocol server sets an X-DocVersion HTTP header which is an xsd:string ([XMLSCHEMA2] section 3.2.1) and specifies the version of the presentation. An X-DocUrl HTTP header is present (except in the case of a presentation broadcast) which specifies the URL of the presentation. All other cases MUST be treated as failure by the protocol client.
503	Failure. The server also sets an HTTP header X-Error xsd:unsignedInt ([XMLSCHEMA2] section 3.3.22) in the response. If $100 \leq \text{value} < 200$, the protocol client SHOULD retry this call, else this header MUST be ignored.

3.1.4.2 MobileSlideInfo

The methods in this section are supported by MobileSlideInfo.ashx.

3.1.4.2.1 GetSlideInfoXml

This method retrieves the information for a particular presentation slide. The URL to this method is constructed by appending "_layouts/" and the value of the **url** attribute of the **slideInfo** element in the **mobilePres** root element returned by the **GetMetadataXml** method, as specified in [MS-PWVRSC] section 2.3.11, to the URL of the site. The parameters to be passed while calling this function are as follows:

infoId: An **xsd:string** element ([XMLSCHEMA2] section 3.2.1) that specifies the presentation slide for which information is to be retrieved. It MUST be set to the value of the **sldInfo** attribute of a **CT_SlideEntry** element specified in [MS-PWVRSC] section 2.3.1.

If any of the preceding conditions about the parameters are not satisfied, the method returns an error.

3.1.4.2.1.1 Return Values

This method MUST send back to the protocol client an HTTP response that MUST be in the following table.

Value	Description
200	If successful, the protocol server returns an XML string specifying information about the requested presentation slide, which contains a sld root element as specified in [MS-PWVRSC] section 2.1.3. All other cases MUST be treated as failure by the protocol client.
503	Failure. The server also sets an HTTP header X-Error xsd:unsignedInt ([XMLSCHEMA2] section 3.3.22) in the response. If $100 \leq \text{value} < 200$, the protocol client SHOULD retry this call, else this header MUST be ignored.

3.1.4.3 MobileSlideImage

The methods in this section are supported by MobileSlideImage.ashx.

3.1.4.3.1 GetSlideImage

This method retrieves the image for a particular presentation slide in a presentation. The URL to this method is constructed by appending "_layouts/" and the value of the **url** attribute of **slideImage** element in the **mobilePres** root element returned by the **GetMetadataXml** method, as specified in [\[MS-PWVRSC\]](#) section 2.3.10, to the URL of the site. The parameters to be passed while calling this function are as follows:

imageId: An **xsd:string** element ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the presentation slide for which information is to be retrieved. It MUST be set to the value of the **sldImg** attribute of a **CT_SlideEntry** element specified in [\[MS-PWVRSC\]](#) section 2.3.1.

width: An **xsd:unsignedInt** element ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the width of the requested image in pixels. The maximum permissible value of this parameter is 1300. If the specified value is 0, a value of 1 MUST be used.

height: An **xsd:unsignedInt** element ([\[XMLSCHEMA2\]](#) section 3.3.22) that specifies the height of the requested image in pixels. The maximum permissible value of this parameter is 1300. If the specified value is 0, a value of 1 MUST be used.

format: An **xsd:string** element ([\[XMLSCHEMA2\]](#) section 3.2.1) that specifies the image format of the requested image. The only acceptable values of this parameter are **png** and **jpeg**. This parameter is optional. The default value of this parameter is **jpeg**. This default value of **jpeg** will be used if any other values other than the acceptable values are used.

If any of the preceding conditions about the parameters are not satisfied, the method returns an error.

3.1.4.3.1.1 Return Values

This method MUST send back to the protocol client an HTTP response that MUST be in the following table.

Value	Description
200	If successful, the protocol server returns an image of the requested page in the specified format. All other cases MUST be treated as failure by the protocol client.
503	Failure. The server also sets an HTTP header X-Error xsd:unsignedInt ([XMLSCHEMA2] section 3.3.22) in the response. If $100 \leq \text{value} < 200$, the protocol client SHOULD retry this call, else this header MUST be ignored.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

The following examples contain a sample interaction between the protocol client and the protocol server. These examples are illustrative of the preceding specification and they do not cover all possible structure usage scenarios. They are not intended to replace the preceding specification but rather to clarify and enhance it. In the following examples, the client wants to view a presentation located at <http://sptestamd/Documents/Demo.pptx>.

4.1 Presentation Information

The URL for **GetMetadataXml** (section [3.1.4.1.1](#)) is constructed as http://sptestamd/_layouts/MobilePresentation.ashx?presentationId=http://sptestamd/Documents/Demo.pptx.

The protocol client then performs an HTTP GET on this URL.

The protocol server responds with a **CT_MobilePres** ([\[MS-PWVRSC\]](#) section 2.3.12) element which contains information about the presentation.

```
<?xml version="1.0" encoding="utf-8"?>
<p:mobilePres xmlns:p="http://schemas.microsoft.com/server/powerpoint/2009/mobile">
  <p:slideImage
    url="/MobileSlideImage.ashx?presentationId=d%3DH%253Asptestamd%252F%252FDocuments%
    252FDemo%252Epptx%26z%3D79AF7428%2D11BC%2D4D49%2DA0D8%2D4AC7DBA09A951"
    image="imageId" width="width" height="height" format="format" />
  <p:slideInfo
    url="/MobileSlideInfo.ashx?presentationId=d%3DH%253Asptestamd%252F%252FDocuments%2
    52FDemo%252Epptx%26z%3D79AF7428%2D11BC%2D4D49%2DA0D8%2D4AC7DBA09A951" info="infoId"
    />
  <p:pres xmlns:p="http://schemas.microsoft.com/server/powerpoint/2009/mobile"
    ver="0">
    <p:docPr w="640" h="480" />
    <p:sldLst>
      <p:sld title="Selling a Product or Service" id="264" sldImg="img264.png"
        sldInfo="264.sldInfo.xml" />
      <p:sld title="Introduction" id="257" sldImg="img257.png"
        sldInfo="257.sldInfo.xml" />
      <p:sld title="Business opportunities" id="258" sldImg="img258.png"
        sldInfo="258.sldInfo.xml" />
      <p:sld title="Products and services " id="259" sldImg="img259.png"
        sldInfo="259.sldInfo.xml" />
      <p:sld title="Key Goals" id="262" sldImg="img262.png"
        sldInfo="262.sldInfo.xml" />
      <p:sld title="Cast comparison" id="260" sldImg="img260.png"
        sldInfo="260.sldInfo.xml" />
      <p:sld title="Next steps" id="263" sldImg="img263.png"
        sldInfo="263.sldInfo.xml" />
      <p:sld title="Conclusion" id="265" sldImg="img265.png"
        sldInfo="265.sldInfo.xml" />
    </p:sldLst>
    <p:showLst>
      <p:custShow name="Custom Show 1">
        <p:sld>264</p:sld>
        <p:sld>259</p:sld>
        <p:sld>263</p:sld>
      </p:custShow>
    </p:showLst>
    <p:showPr>
      <p:sldRg st="3" end="6" />
    </p:showPr>
  </p:pres>
</p:mobilePres>
```

```
</p:showPr>
</p:pres>
</p:mobilePres>
```

The contained **slideImage** and **slideInfo** elements specify information how to retrieve the presentation slide images and the detailed information about a presentation slide respectively.

The **w** and **h** attributes specify that the width and height of the presentation are 640 pixels and 480 pixels respectively.

4.2 Slide Image

The **CT_SlideImage** [MS-PWVRSC] section 2.3.10 element contained within **CT_MobilePres** ([MS-PWVRSC] section 2.3.12) element specifies how to retrieve the slide images.

The URL to retrieve the image of slide having identifier 264 of width 640 pixels and height 480 pixels is constructed as follows:

```
http://sptestamd/_layouts/MobileSlideImage.ashx?presentationId=d%3DH%253Asptestamd%252F%252FD
ocuments%252FDemo%252Epptx%26z%3DFE7280F5%2D1895%2D4F49%2DA3E8%2D8E4628A833E52&imageId=img264
.png&width=640&height=480
```

The protocol client performs an HTTP GET on this URL to obtain the slide image.

4.3 Slide Information

The **CT_SlideInfo** ([MS-PWVRSC] section 2.3.11) element contained within a **CT_MobilePres** ([MS-PWVRSC] section 2.3.12) element specifies how to retrieve the slide information.

The URL to retrieve information about slide 264 is constructed as follows:

```
http://sptestamd/_layouts/MobileSlideInfo.ashx?presentationId=d%3DH%253Asptestamd%252F%252FD
ocuments%252FDemo%252Epptx%26z%3DFE7280F5%2D1895%2D4F49%2DA3E8%2D8E4628A833E52&infoId=264.sldI
nfo.xml
```

The protocol client performs an HTTP GET on this URL.

The protocol server responds with a **CT_Slide** ([MS-PWVRSC] section 2.3.18) element which contains information about the slide.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<p:sld xmlns:p="http://schemas.microsoft.com/server/powerpoint/2009/mobile"
ver="0">
  <p:sldTxt>
    <p:ph type="t">
      <p:p>
        <p:t>Selling a Product or Service</p:t>
      </p:p>
    </p:ph>
  </p:sldTxt>
  <p:notes>
    <p:p buChar="•">
      <p:t>Total time: 30 minutes</p:t>
    </p:p>
  </p:notes>
</p:sld>
```

```
<p:t>Concentrate on Business opportunities</p:t>
</p:p>
<p:p level="2" buChar="•">
  <p:t>Spend 5 minutes on intro</p:t>
</p:p>
<p:p>
  <p:t/>
</p:p>
<p:p>
  <p:t/>
</p:p>
<p:p>
  <p:t>Text for test</p:t>
</p:p>
<p:p>
  <p:t/>
</p:p>
<p:p>
  <p:t/>
</p:p>
</p:notes>
</p:sld>
```

The contained **notes** element specifies the text content of the notes of a presentation slide.

5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

None.

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 FAST Search Server 2010
- Microsoft PowerPoint Mobile 2010
- Microsoft PowerPoint Online

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.

8 Change Tracking

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

9 Index

A

Abstract data model
[server](#) 10
[Applicability](#) 8
[Attribute groups](#) 9
[Attributes](#) 9

C

[Capability negotiation](#) 8
[Change tracking](#) 21
Client
[overview](#) 10
[Complex types](#) 9

D

Data model - abstract
[server](#) 10

E

Events
[local - server](#) 14
[timer - server](#) 14
Examples
[overview](#) 15
[presentation information](#) 15
[slide image](#) 16
[slide information](#) 16

F

[Fields - vendor-extensible](#) 8
[Full WSDL](#) 19

G

[Glossary](#) 6
[Groups](#) 9

I

[Implementer - security considerations](#) 18
[Index of security parameters](#) 18
[Informative references](#) 7
Initialization
[server](#) 11
[Introduction](#) 6

L

Local events
[server](#) 14

M

Message processing
[server](#) 11
Messages

[attribute groups](#) 9
[attributes](#) 9
[complex types](#) 9
[elements](#) 9
[enumerated](#) 9
[groups](#) 9
[namespaces](#) 9
[simple types](#) 9
[syntax](#) 9
[transport](#) 9

N

[Namespaces](#) 9
[Normative references](#) 7

O

Operations
[MobilePresentation](#) 11
[MobileSlideImage](#) 14
[MobileSlideInfo](#) 13
[Overview \(synopsis\)](#) 7

P

[Parameters - security index](#) 18
[Preconditions](#) 8
[Prerequisites](#) 8
[Presentation information example](#) 15
[Product behavior](#) 20
Protocol Details
[overview](#) 10

R

[References](#) 6
[informative](#) 7
[normative](#) 7
[Relationship to other protocols](#) 7

S

Security
[implementer considerations](#) 18
[parameter index](#) 18
Sequencing rules
[server](#) 11
Server
[abstract data model](#) 10
[details](#) 10
[initialization](#) 11
[local events](#) 14
[message processing](#) 11
[MobilePresentation operation](#) 11
[MobileSlideImage operation](#) 14
[MobileSlideInfo operation](#) 13
[overview](#) 10
[sequencing rules](#) 11
[timer events](#) 14
[timers](#) 10
[Simple types](#) 9

[Slide image example](#) 16
[Slide information example](#) 16
[Standards assignments](#) 8
Syntax
[messages - overview](#) 9

T

Timer events
[server](#) 14
Timers
[server](#) 10
[Tracking changes](#) 21
[Transport](#) 9
Types
[complex](#) 9
[simple](#) 9

V

[Vendor-extensible fields](#) 8
[Versioning](#) 8

W

[WSDL](#) 19