

[MS-VGSP]: Visio Graphics Service Protocol Specification

Intellectual Property Rights Notice for Open Specifications Documentation

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

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

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

Preliminary Documentation. This Open Specification provides documentation for past and current releases and/or for the pre-release (beta) version of this technology. This Open Specification is final

documentation for past or current releases as specifically noted in the document, as applicable; it is preliminary documentation for the pre-release (beta) versions. Microsoft will release final documentation in connection with the commercial release of the updated or new version of this technology. As the documentation may change between this preliminary version and the final version of this technology, there are risks in relying on preliminary documentation. To the extent that you incur additional development obligations or any other costs as a result of relying on this preliminary documentation, you do so at your own risk.

Revision Summary

Date	Revision History	Revision Class	Comments
07/13/2009	0.1	Major	Initial Availability
08/28/2009	0.2	Editorial	Revised and edited the technical content
11/06/2009	0.3	Editorial	Revised and edited the technical content
02/19/2010	1.0	Major	Updated and revised the technical content
03/31/2010	1.01	Editorial	Revised and edited the technical content
04/30/2010	1.02	Editorial	Revised and edited the technical content
06/07/2010	1.03	Editorial	Revised and edited the technical content
06/29/2010	1.04	Editorial	Changed language and formatting in the technical content.
07/23/2010	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
09/27/2010	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
03/18/2011	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
06/10/2011	1.5	Minor	Clarified the meaning of the technical content.
01/20/2012	2.0	Major	Significantly changed the technical content.

Table of Contents

1 Introduction	7
1.1 Glossary	7
1.2 References	8
1.2.1 Normative References	8
1.2.2 Informative References	9
1.3 Protocol Overview (Synopsis)	9
1.3.1 Web Drawing	9
1.3.2 Drawing Page	10
1.3.2.1 Foreground Page	10
1.3.3 Shape	10
1.3.4 Shape Data	10
1.3.5 Comment	10
1.4 Relationship to Other Protocols	10
1.5 Prerequisites/Preconditions	11
1.6 Applicability Statement	11
1.7 Versioning and Capability Negotiation	11
1.8 Vendor-Extensible Fields	11
1.9 Standards Assignments	11
2 Messages	12
2.1 Transport	12
2.2 Common Message Syntax	12
2.2.1 Namespaces	12
2.2.2 Messages	13
2.2.3 Elements	13
2.2.4 Complex Types	13
2.2.4.1 AddonDataSource	14
2.2.4.2 ArrayOfAddonDataSource	14
2.2.4.3 BaseRequestContract	15
2.2.4.4 DiagramBase	16
2.2.4.5 VisioGraphicsServiceFault	16
2.2.5 Simple Types	17
2.2.5.1 FaultCode	17
2.2.6 Attributes	19
2.2.7 Groups	19
2.2.8 Attribute Groups	20
2.2.9 Common Data Structures	20
2.2.9.1 ShapeInfo	20
2.2.9.1.1 Page	21
2.2.9.1.2 Pages	22
2.2.9.1.3 PageInfo	23
2.2.9.1.4 ShapeInfo	23
2.2.9.1.5 Hyperlinks	25
2.2.9.1.6 Hyperlink	25
2.2.9.1.7 ShapeDataItems	26
2.2.9.1.8 ShapeData	26
2.2.9.1.9 Shapes	27
2.2.9.1.10 Shape	27
2.2.9.1.11 Path	27
2.2.9.2 Comments	28

2.2.9.2.1	Comments.....	28
2.2.9.2.2	CommentEntry.....	28
2.2.9.3	CommentData.....	29
2.2.9.3.1	Comment.....	29
2.2.9.4	CommentResult.....	30
2.2.9.4.1	Comment.....	30
3	Protocol Details.....	32
3.1	Common Details.....	32
3.1.1	Abstract Data Model.....	34
3.1.2	Timers.....	34
3.1.3	Initialization.....	34
3.1.4	Message Processing Events and Sequencing Rules.....	35
3.1.4.1	GetRasterDiagram.....	35
3.1.4.1.1	Messages.....	36
3.1.4.1.1.1	IVisioGraphicsService_GetRasterDiagram_InputMessage.....	37
3.1.4.1.1.2	IVisioGraphicsService_GetRasterDiagram_OutputMessage.....	37
3.1.4.1.1.3	IVisioGraphicsService_GetRasterDiagram_VisioGraphicsServiceFaultFault_Message.....	37
3.1.4.1.2	Elements.....	37
3.1.4.1.2.1	GetRasterDiagram.....	37
3.1.4.1.2.2	GetRasterDiagramResponse.....	38
3.1.4.1.3	Complex Types.....	38
3.1.4.1.3.1	RasterDiagramRequestContract.....	38
3.1.4.1.3.2	RasterDiagram.....	39
3.1.4.1.4	Simple Types.....	40
3.1.4.1.5	Attributes.....	40
3.1.4.1.6	Groups.....	40
3.1.4.1.7	Attribute Groups.....	40
3.1.4.2	GetVectorDiagram.....	40
3.1.4.2.1	Messages.....	41
3.1.4.2.1.1	IVisioGraphicsService_GetVectorDiagram_InputMessage.....	41
3.1.4.2.1.2	IVisioGraphicsService_GetVectorDiagram_OutputMessage.....	41
3.1.4.2.1.3	IVisioGraphicsService_GetVectorDiagram_VisioGraphicsServiceFaultFault_Message.....	42
3.1.4.2.2	Elements.....	42
3.1.4.2.2.1	GetVectorDiagram.....	42
3.1.4.2.2.2	GetVectorDiagramResponse.....	42
3.1.4.2.3	Complex Types.....	43
3.1.4.2.3.1	VectorDiagramRequestContract.....	43
3.1.4.2.3.2	VectorDiagram.....	43
3.1.4.2.3.3	RenderedPage.....	44
3.1.4.2.3.4	ArrayOfDiagramResource.....	45
3.1.4.2.3.5	DiagramResource.....	45
3.1.4.2.4	Simple Types.....	45
3.1.4.2.5	Attributes.....	45
3.1.4.2.6	Groups.....	46
3.1.4.2.7	Attribute Groups.....	46
3.1.4.3	GetRasterPage.....	46
3.1.4.3.1	Messages.....	47
3.1.4.3.1.1	IVisioGraphicsService_GetRasterPage_InputMessage.....	47

3.1.4.3.1.2	IVisioGraphicsService_GetRasterPage_OutputMessage	47
3.1.4.3.1.3	IVisioGraphicsService_GetRasterPage_VisioGraphicsServiceFaultFault t_FaultMessage	47
3.1.4.3.2	Elements	48
3.1.4.3.2.1	GetRasterPage	48
3.1.4.3.2.2	GetRasterPageResponse	48
3.1.4.3.3	Complex Types	48
3.1.4.3.3.1	RasterPageRequest	49
3.1.4.3.3.2	RasterPageResponse	50
3.1.4.3.3.3	RasterPageTag	51
3.1.4.3.3.4	RasterPageInfo	51
3.1.4.3.4	Simple Types	52
3.1.4.3.4.1	RasterPageResponseCode	52
3.1.4.3.5	Attributes	53
3.1.4.3.6	Groups	53
3.1.4.3.7	Attribute Groups	53
3.1.4.4	GetRasterPageItem	53
3.1.4.4.1	Messages	54
3.1.4.4.1.1	IVisioGraphicsService_GetRasterPageItem_InputMessage	54
3.1.4.4.1.2	IVisioGraphicsService_GetRasterPageItem_OutputMessage	54
3.1.4.4.1.3	IVisioGraphicsService_GetRasterPageItem_VisioGraphicsServiceFault Fault_FaultMessage	54
3.1.4.4.2	Elements	54
3.1.4.4.2.1	GetRasterPageItem	55
3.1.4.4.2.2	GetRasterPageItemResponse	55
3.1.4.4.3	Complex Types	55
3.1.4.4.3.1	RasterPageItemRequest	56
3.1.4.4.3.2	RasterPageItemTag	57
3.1.4.4.3.3	RasterPageItemResponse	57
3.1.4.4.4	Simple Types	58
3.1.4.4.4.1	RasterPageItemResponseCode	58
3.1.4.4.5	Attributes	59
3.1.4.4.6	Groups	59
3.1.4.4.7	Attribute Groups	59
3.1.4.5	SaveComments	59
3.1.4.5.1	Messages	60
3.1.4.5.1.1	IVisioGraphicsService_SaveComments_InputMessage	60
3.1.4.5.1.2	IVisioGraphicsService_SaveComments_OutputMessage	60
3.1.4.5.1.3	IVisioGraphicsService_SaveComments_VisioGraphicsServiceFaultFault t_FaultMessage	60
3.1.4.5.2	Elements	60
3.1.4.5.2.1	SaveComments	61
3.1.4.5.2.2	SaveCommentsResponse	61
3.1.4.5.3	Complex Types	61
3.1.4.5.4	Simple Types	61
3.1.4.5.5	Attributes	61
3.1.4.5.6	Groups	62
3.1.4.5.7	Attribute Groups	62
3.1.5	Timer Events	62
3.1.6	Other Local Events	62

4 Protocol Examples	63
4.1 GetRasterDiagram	63
4.2 GetVectorDiagram	69
4.3 GetVectorDiagram with AddonDataSource	71
5 Security	88
5.1 Security Considerations for Implementers.....	88
5.2 Index of Security Parameters	88
6 Appendix A: Full WSDL	89
7 Appendix B: Full XML Schema	102
7.1 http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer.Schema	102
7.2 http://schemas.microsoft.com/2003/10/Serialization/Arrays.Schema	106
7.3 http://schemas.microsoft.com/2003/10/Serialization/Schema	106
7.4 http://tempuri.org/Schema	107
8 Appendix C: Product Behavior	110
9 Change Tracking	111
10 Index	129

1 Introduction

This document specifies the Visio Graphics Service Protocol. This protocol enables a protocol client to communicate with a protocol server to retrieve information about a [web drawing](#) that is stored in a document repository.

Sections 1.8, 2, and 3 of this specification are normative and contain RFC 2119 language. Sections 1.5 and 1.9 are also normative but cannot contain RFC 2119 language. All other sections and examples in this specification are informative.

1.1 Glossary

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

Coordinated Universal Time (UTC)
domain
GUID
Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)

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

data connection
data provider
data source
drawing
embedded object
endpoint
Extensible Application Markup Language (XAML)
font
fully qualified class name
hyperlink
list
Office data connection (ODC) file
Portable Network Graphics (PNG)
primary key
row
Simple Object Access Protocol (SOAP)
SOAP action
SOAP body
SOAP fault
Uniform Resource Identifier (URI)
Uniform Resource Locator (URL)
view
Web Services Description Language (WSDL)
WSDL message
WSDL operation
XML namespace
XML namespace prefix
XML schema
zoom level

The following terms are specific to this document:

pixel: A discrete unit of display on a computer display device.

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

1.2 References

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

1.2.1 Normative References

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

[MS-DSDIFFGRAM] Microsoft Corporation, "[SharePoint Web Services: DataSet DiffGram Structure Specification](#)".

[MS-SLXV] Microsoft Corporation, "Silverlight XAML Vocabulary Specification 2008", October 2008, [Silverlight XAML Vocabulary Specification 2008](#)

[MS-SPSTWS] Microsoft Corporation, "[SharePoint Security Token Service Web Service Protocol Specification](#)".

[MS-VGSFF] Microsoft Corporation, "[Visio Graphics Service File Format \(.vdw\) Specification](#)".

[MS-VSDX] Microsoft Corporation, "[Visio Graphics Service VSDX File Format](#)".

[RFC1738] Berners-Lee, T., Masinter, L., and McCahill, M., "Uniform Resource Locators (URL)", RFC 1738, December 1994, <http://www.ietf.org/rfc/rfc1738.txt>

[RFC1980] Seidman, J., "A Proposed Extension to HTML : Client-Side Image Maps", RFC 1980, August 1996, <http://www.rfc-editor.org/rfc/rfc1980.txt>

[RFC2083] Boutell, T., "PNG (Portable Network Graphics) Specification Version 1.0", RFC 2083, March 1997, <http://www.ietf.org/rfc/rfc2083.txt>

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

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

[RFC4627] Crockford, D., "The application/json Media Type for Javascript Object Notation (JSON)", RFC 4627, July 2006, <http://www.ietf.org/rfc/rfc4627.txt>

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

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

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

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

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

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

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

1.2.2 Informative References

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

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

[MS-SPTWS] Microsoft Corporation, "[Service Platform Topology Web Service Protocol Specification](#)".

[RFC1952] Deutsch, P., "GZIP file format specification version 4.3", RFC 1952, May 1996, <http://www.rfc-editor.org/rfc/rfc1952.txt>

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

[RFC3629] Yergeau, F., "UTF-8, A Transformation Format of ISO 10646", STD 63, RFC 3629, November 2003, <http://www.ietf.org/rfc/rfc3629.txt>

1.3 Protocol Overview (Synopsis)

The protocol enables a protocol client to request and receive a [drawing page](#) of a [web drawing](#) from the protocol server in either raster or vector format. The supported raster format is the **Portable Network Graphics (PNG)** format, as specified in [\[RFC2083\]](#). The supported vector format is **XAML**, as specified in [\[MS-SLXV\]](#).

Additionally, the protocol enables a protocol client to add, edit, or delete [comments](#) associated with a drawing page or its [shapes](#).

A typical scenario for using this protocol is an application that retrieves and displays web drawings, and collects collaborative content feedback in comments. Typically a web drawing is contained in a file stored in a document repository.

1.3.1 Web Drawing

A **web drawing** is a collection of [drawing pages](#), [shapes](#), resources, [comments](#), and information that can be rendered as a **drawing** in a web browser. Based on the format of the file that contains the web drawing, the protocol makes the distinction between the following types of web drawings.

Type	Description
VGSFF	A web drawing contained in a file that conforms to the format specified in [MS-VGSFF] section

Type	Description
	2.2.1
VSDX	A web drawing contained in a file that conforms to the format specified in [MS-VSDX] section 2.2.1

1.3.2 Drawing Page

A **drawing page** is a collection of [shapes](#) that are viewed together.

Drawing pages of **VGSFF** [web drawings](#) are specified in [\[MS-VGSFF\]](#) section 2.2.2.

Drawing pages of **VSDX** [web drawings](#) are specified in [\[MS-VSDX\]](#) section 2.2.2.

1.3.2.1 Foreground Page

Foreground pages of **VSDX** [web drawings](#) are specified [\[MS-VSDX\]](#) section 2.2.2.4.

1.3.3 Shape

A **shape** is a collection of geometry, formatting, text, images, hyperlinks, and shape data in a [drawing page](#).

Shapes contained in **VGSFF** [web drawings](#) are specified in [\[MS-VGSFF\]](#) section 2.2.3.

Shapes contained in **VSDX** [web drawings](#) are specified in [\[MS-VSDX\]](#) section 2.2.3.

1.3.4 Shape Data

A [shape](#) can have data associated with it that provides information about its meaning. Shape data can come from information stored in the file, refreshed from data providers, or recalculations from an evaluated formula. The shape data is also formatted for display.

Shapes data items contained in **VGSFF** [web drawings](#) are specified in [\[MS-VGSFF\]](#) section 2.2.3.5.

Shapes data items contained in **VSDX** [web drawings](#) are specified in [\[MS-VSDX\]](#) section 2.2.3.6.

1.3.5 Comment

A **comment** is a plain text annotation in a [web drawing](#). Each comment specifies an author, and a relationship with a [drawing page](#) or a specific [shape](#) on the drawing page.

Comments contained in **VSDX** [web drawings](#) are specified in [\[MS-VSDX\]](#) section 2.2.9.

Comments are not contained in **VGSFF** [web drawings](#).

1.4 Relationship to Other Protocols

This protocol uses the **Simple Object Access Protocol (SOAP)** message protocol for formatting request and response messages, as described in [\[SOAP1.1\]](#), [\[SOAP1.2/1\]](#) and [\[SOAP1.2/2\]](#). It transmits those messages by using **HTTP** as described in [\[RFC2616\]](#), or **Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)** as described in [\[RFC2818\]](#).

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

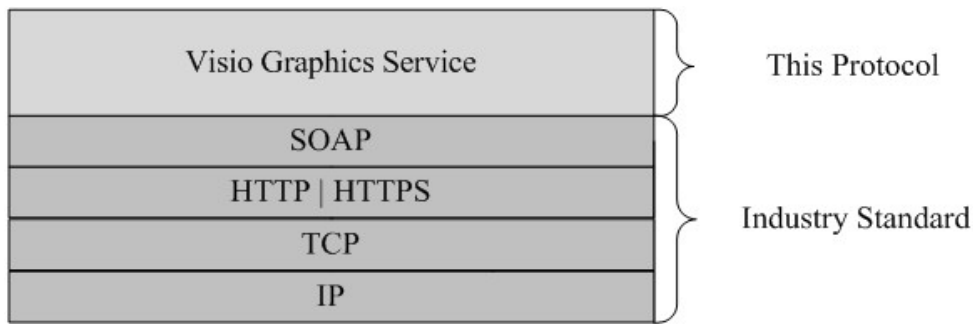


Figure 1: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a protocol server that exposes one or more **endpoint (4) URIs** that are known by protocol clients. The endpoint (4) URI of the protocol server and the transport that is used by the protocol server are either known by the protocol client or obtained by using the discovery mechanism that is described in [\[MS-SPTWS\]](#).

The protocol client obtains the requisite values of the **ApplicationClassId** and **ApplicationVersion** properties and the endpoint (4) URI of the protocol server that provides the discovery mechanism, as described in [\[MS-SPTWS\]](#), by means that are independent of either protocol.

This protocol requires the protocol client to have the appropriate permission to call the methods on the protocol server.

The protocol client implements the token-based security mechanisms that are required by the protocol server and related security protocols, as described in [\[MS-SPSTWS\]](#).

1.6 Applicability Statement

This protocol is designed to support retrieval of [web drawings](#), and collecting collaborative content feedback in [comments](#).

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

- **Supported Transports:** This protocol can be implemented by using transports that support sending Simple Object Access Protocol (SOAP) messages, as specified in section 2.1.
- **Protocol Versions:** This protocol is not versioned.
- **Capability Negotiation:** This protocol does not support version negotiation.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

In the following sections, the schema definition might differ from the processing rules imposed by the protocol. The **WSDL** in this specification matches the WSDL that shipped with the product and provides a base description of the schema. The text that introduces the WSDL might specify differences that reflect actual Microsoft product behavior.

For example, the schema definition might allow for an element to be **empty**, **null**, or **not present**, but the behaviour of the protocol, as specified, restricts the same elements to being **non-empty**, **not null**, and **present**.

2.1 Transport

Protocol servers **MUST** support SOAP over Hypertext Transfer Protocol (HTTP) or Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS).

All protocol messages **MUST** be transported by using HTTP bindings at the transport level.

Protocol messages **MUST** be formatted as specified in either [\[SOAP1.1\]](#) section 4 or [\[SOAP1.2/1\]](#) section 5. Protocol server faults **MUST** be returned by using one of the following:

- HTTP status codes, as specified in [\[RFC2616\]](#) section 10.
- **SOAP faults**, as specified in [\[SOAP1.1\]](#) section 4.4.
- [\[SOAP1.2/1\]](#) section 5.4.

If the HTTPS transport is used, a server certificate **MUST** be deployed.

This protocol **MAY** transmit an additional SOAP header, the **ServiceContext** header, as specified in [\[MS-SPSTWS\]](#) section 2.2.4.1.

This protocol does not define any means for activating a protocol server or protocol client. The protocol server **MUST** be configured, and begin listening in an implementation-specific way. In addition, the protocol client **MUST** know the format and transport that is used by the server. For example, the SOAP format over an HTTP transport.

2.2 Common Message Syntax

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

2.2.1 Namespaces

This specification defines and references various **XML namespaces** using the mechanisms specified in [\[XMLNS\]](#). Although this specification associates a specific **XML namespace prefix** 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
msb	http://schemas.microsoft.com/ws/06/2004/mspolicy/netbinary1	
q1	http://schemas.microsoft.com/2003/10/Serialization/Arrays	

Prefix	Namespace URI	Reference
soap12	http://schemas.xmlsoap.org/wsdl/soap12/	[SOAP1.2/1] [SOAP1.2/2]
sp	http://schemas.xmlsoap.org/ws/2005/07/securitypolicy	
t	http://schemas.xmlsoap.org/ws/2005/02/trust	
tns	http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer	
tns1	http://schemas.microsoft.com/2003/10/Serialization/	
tns2	http://tempuri.org/	
tns3	http://tempuri.org/Imports	
wsaw	http://www.w3.org/2006/05/addressing/wsdl	
wsdl	http://schemas.xmlsoap.org/wsdl/	[WSDL]
wsp	http://schemas.xmlsoap.org/ws/2004/09/policy	
wsu	http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd	
xs	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1] 1 [XMLSCHEMA2] 1

2.2.2 Messages

This specification does not define any common WSDL message definitions.

2.2.3 Elements

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

2.2.4 Complex Types

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

Complex Type	Description
AddonDataSource	The AddonDataSource complex type contains information about a data provider implemented by a protocol client and used in a web drawing .
ArrayOfAddonDataSource	The ArrayOfAddonDataSource complex type contains an array of AddonDataSource elements (section 2.2.4.1).
BaseRequestContract	The BaseRequestContract complex type contains information used to specify the request parameters for a protocol server operation to retrieve a VGSPF web drawing.

Complex Type	Description
DiagramBase	The DiagramBase complex type contains information used to specify response data for a protocol server operation to retrieve a VGSFF web drawing..
VisioGraphicsServiceFault	The VisioGraphicsServiceFault complex type contains information about errors that occur during protocol server operations.

2.2.4.1 AddonDataSource

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **AddonDataSource** complex type contains information about a data provider implemented by a protocol client and used in a [web drawing](#).

```
<xs:complexType name="AddonDataSource" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="ConnectionString" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Data" nillable="true">
      <xs:complexType>
        <xs:sequence>
          <xs:element ref="xs:schema"/>
          <xs:any/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element minOccurs="0" name="Id" type="xs:int"/>
    <xs:element minOccurs="0" name="QueryString" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

ConnectionString: A string that specifies arguments that defines the location of the data provider and how to connect to it. There **MUST** be exactly one instance of this element, and it **MUST NOT** be null. It **MUST** contain the **fully qualified class name** of the data provider.

Data: Specifies the data retrieved from the data provider. It **MUST** be encoded in the DiffGram [\[MS-DSDIFFGRAM\]](#) format, and it **MUST NOT** be null.

xs:schema: Specifies the schema for the **Data** element.

Id: An integer that specifies the identifier of this **AddonDataSource**. It **MUST** be greater than or equal to zero, and it **MUST** be unique across all the **AddonDataSource** elements in the containing **ArrayOfAddonDataSource** array (section [2.2.4.2](#)). There **MUST** be exactly one instance of this element.

QueryString: A string that specifies information used to call and retrieve data from the data provider. There **MUST** be exactly one instance of this element, and it **MUST NOT** be null.

2.2.4.2 ArrayOfAddonDataSource

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **ArrayOfAddonDataSource** complex type contains an array of **AddonDataSource** elements (section [2.2.4.1](#)).

```
<xs:complexType name="ArrayOfAddonDataSource" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="AddonDataSource" nillable="true"
      type="tns:AddonDataSource"/>
  </xs:sequence>
</xs:complexType>
```

AddonDataSource: Specifies an **AddonDataSource** element. This element MUST NOT be null.

2.2.4.3 BaseRequestContract

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **BaseRequestContract** complex type contains information used to specify the request parameters for a protocol server operation to retrieve a **VGSFF web drawing**.

```
<xs:complexType name="BaseRequestContract" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="DataSources" nillable="true"
      type="tns:ArrayOfAddonDataSource"/>
    <xs:element minOccurs="0" name="DisableRefresh" type="xs:boolean"/>
    <xs:element minOccurs="0" name="FileUri" nillable="true" type="xs:anyURI"/>
    <xs:element minOccurs="0" name="PageNumber" type="xs:int"/>
    <xs:element minOccurs="0" name="TimeLastModified" type="xs:dateTime"/>
  </xs:sequence>
</xs:complexType>
```

DataSources: An **ArrayOfAddonDataSource** element (section [2.2.4.2](#)) that specifies the data providers implemented by a protocol client, and used in the web drawing. If there are no data providers, **DataSources** MUST have zero child elements. If **DataSources** has child elements, the **Data** field of each **AddonDataSource** element (section [2.2.4.1](#)) MUST NOT be null.

DisableRefresh: A boolean that specifies whether the requested [drawing page](#) is refreshed by the protocol server. A value of "true" indicates that the returned drawing page is not refreshed; a value of "false" indicates that the returned drawing page is refreshed. There MUST be exactly one instance of this element.

FileUri: An **anyURI** element that specifies the **URL** of the file in a document repository that contains the **VGSFF** web drawing being requested. There MUST be exactly one instance of this element, and it MUST NOT be null.

PageNumber: An integer that specifies the one-based index of a drawing page in the web drawing. It MUST be greater than zero and less than or equal to the total number of drawing pages in the requested web drawing. There MUST be exactly one instance of this element.

TimeLastModified: An optional **dateTime** element that specifies the date and time, in **Coordinated Universal Time (UTC)** format, when the file in a document repository that contains the **VGSFF** web drawing, was last modified. If the attribute is present, there MUST be exactly one instance of this element.

2.2.4.4 DiagramBase

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer

The **DiagramBase** complex type contains information used to specify response data for a protocol server operation to retrieve a **VGSFF** [web drawing](#)..

```
<xs:complexType name="DiagramBase" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="CacheAge" type="xs:int"/>
    <xs:element minOccurs="0" name="DataSources" nillable="true"
type="tns:ArrayOfAddonDataSource"/>
  </xs:sequence>
</xs:complexType>
```

CacheAge: An integer that specifies the duration in minutes for which the processed [drawing page](#) is cached in memory by the protocol server. It MUST be greater than or equal to zero. There MUST be exactly one instance of this element.

DataSources: An **ArrayOfAddonDataSource** element (section [2.2.4.2](#)) that specifies the data providers implemented by a protocol client, and used in the web drawing. If there are no data providers, **DataSources** MUST have zero child elements. If **DataSources** has child elements, the **Data** field of each **AddonDataSource** element (section [2.2.4.1](#)) MUST NOT be empty.

2.2.4.5 VisioGraphicsServiceFault

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer

The **VisioGraphicsServiceFault** complex type contains information about errors that occur during protocol server operations.

```
<xs:complexType name="VisioGraphicsServiceFault" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Details" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Error" type="tns:FaultCode"/>
    <xs:element minOccurs="0" name="Flags" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

Details: A string that specifies a detailed description of an error. There MUST be exactly one instance of this element, and it MUST NOT be NULL.

Error: A **FaultCode** simple type (section [2.2.5.1](#)) that specifies the type of error. There MUST be exactly one instance of this element.

Flags: An integer that specifies additional information related to the **FaultCode**. A nonzero value indicates an error occurred during mapping **rows (2)** of data to [shapes](#) when processing the request for a [drawing page](#) of a **VGSFF** [web drawing](#). There MUST be exactly one instance of this element. If **FaultCode** is not equal to **DataBindingConflict**, the value MUST be "0x00000000". If **FaultCode** is equal to **DataBindingConflict**, the value MUST be a bitwise OR combination of one or more of the values in the following table, or the value "0x40000000", which indicates a generic data binding failure.

Value	Description
0x00000001	A column was deleted in the data source (1) .
0x00000002	A column's data type was changed in the data source (1).
0x00000004	A row (2) was deleted in the data source (1).
0x00000008	A duplicate row (2) was found in the data source (1).
0x00000010	A primary key column was deleted in the data source (1).

2.2.5 Simple Types

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

Simple type	Description
FaultCode	The FaultCode simple type contains information about a specific error that occurs during a protocol operation. It MUST be a valid value as defined in the following table.

2.2.5.1 FaultCode

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **FaultCode** simple type contains information about a specific error that occurs during a protocol operation.

It MUST be a valid value as defined in the following table.

```
<xs:simpleType name="FaultCode" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:restriction base="xs:string">
    <xs:enumeration value="DataConnectivityError"/>
    <xs:enumeration value="DataSourceConnectionError"/>
    <xs:enumeration value="DataSourcePermissionError"/>
    <xs:enumeration value="EmptyRasterDiagramRequest"/>
    <xs:enumeration value="EmptyVectorDiagramRequest"/>
    <xs:enumeration value="OfficeDataConnectionFileNotFound"/>
    <xs:enumeration value="OfficeDataConnectionFileRetrievalError"/>
    <xs:enumeration value="OfficeDataConnectionFileAccessDenied"/>
    <xs:enumeration value="IncompatibleDomain"/>
    <xs:enumeration value="WssViewAccessDenied"/>
    <xs:enumeration value="WssListAccessDenied"/>
    <xs:enumeration value="InvalidPageNumber"/>
    <xs:enumeration value="InvalidResolution"/>
    <xs:enumeration value="InvalidFileUrl"/>
    <xs:enumeration value="RasterizationError"/>
    <xs:enumeration value="InvalidFileExtension"/>
    <xs:enumeration value="EmptyServerData"/>
    <xs:enumeration value="OpenServerFileFailed"/>
    <xs:enumeration value="NoReadPermissionToFile"/>
  </xs:restriction>
</xs:simpleType>
```

```

<xs:enumeration value="KeyNotFoundFromCollection"/>
<xs:enumeration value="ExceededMaxDiagramSize"/>
<xs:enumeration value="IncompatibleVersion"/>
<xs:enumeration value="DataProviderNotTrusted"/>
<xs:enumeration value="DataBindingConflict"/>
<xs:enumeration value="UnableToGetExternalDataSourceCredentials"/>
<xs:enumeration value="UnattendedAccountTypeMismatch"/>
<xs:enumeration value="RecalcTimedOut"/>
<xs:enumeration value="LicenseExpired"/>
<xs:enumeration value="VSDXFileLoadGenericError"/>
<xs:enumeration value="VSDXFileLoadPermissionError"/>
<xs:enumeration value="VSDXFileSizeError"/>
<xs:enumeration value="WebPartConfigError"/>
</xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for **FaultCode**.

Value	Meaning
DataConnectivityError	An error occurred while trying to connect to data sources specified in the document.
DataSourceConnectionError	An error occurred while trying to connect to data sources specified in the VSDX Web drawing .
DataSourcePermissionError	The user does not have permission to connect to data sources specified in the VSDX Web drawing.
EmptyRasterDiagramRequest	A required argument to the GetRasterDiagram WSDL operation is empty.
EmptyVectorDiagramRequest	A required argument to the GetVectorDiagram WSDL operation is empty.
OfficeDataConnectionFileNotFound	The Office data connection (ODC) file specified in the document could not be found.
OfficeDataConnectionFileRetrievalError	The Office data connection (ODC) file URL specified in the document is not a well formed URL as specified in [RFC1738] .
OfficeDataConnectionFileAccessDenied	Access to the Office data connection (ODC) file specified in the document was denied.
IncompatibleDomain	The specified document is referring to a list (1) data source that is not in a compatible domain .
WssViewAccessDenied	The user does not have permissions to the view used as external data source by the web drawing.
WssListAccessDenied	The user does not have permissions to the list (1) used as external data source by the web drawing.
InvalidPageNumber	The page index is less than 1.
InvalidResolution	The Resolution specified in RasterDiagramRequestContract is invalid.

Value	Meaning
InvalidFileUrl	Unused and MUST be ignored.
RasterizationError	An error occurred during rasterization of the document.
InvalidFileExtension	The file reference does not refer to a file with a valid extension.
EmptyServerData	The data in the document is empty or invalid.
OpenServerFileFailed	The document could not be opened.
NoReadPermissionToFile	Access to the document was denied.
KeyNotFoundFromCollection	The names of drawing pages cannot be retrieved from the document.
ExceededMaxDiagramSize	The document size exceeded the limit configured by the service administrator.
IncompatibleVersion	The document version is incompatible with the server.
DataProviderNotTrusted	The document requires a connection to data provider(s) that are not allowed by the service administrator.
DataBindingConflict	Unable to update the graphical elements that are mapped to data while processing a request for a drawing page of a VGSFF Web drawing.
UnableToGetExternalDataSourceCredentials	Credentials required to connect to data sources specified in the document could not be retrieved.
UnattendedAccountTypeMismatch	The default credentials used for connecting to data sources are of invalid type.
RecalcTimedOut	The document recalculation duration exceeded the limit configured by the service administrator.
LicenseExpired	Visio Graphics Service license has expired.
VSDXFileLoadGenericError	An error occurred opening the VSDX Web drawing.
VSDXFileLoadPermissionError	The user does not have permission to read the VSDX Web drawing.
VSDXFileSizeError	The VSDX document size exceeded the limit configured by the service administrator.
WebPartConfigError	An invalid request occurred for a VSDX Web drawing.

2.2.6 Attributes

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

2.2.7 Groups

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

2.2.8 Attribute Groups

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

2.2.9 Common Data Structures

The following table summarizes the set of data structures specific to messages that are associated with **VSDX** web drawings.

Data Structure	Description
ShapeInfo	A string that is an XML fragment containing information about shapes , drawing pages , and web drawing .
Comments	A string that is an XML fragment containing comments.
CommentData	A string that is an XML fragment containing the comment action input data.
CommentResult	A string that is an XML fragment containing the result of a comment action.

2.2.9.1 ShapeInfo

The **ShapeInfo** element is a string that specifies information about the [shapes](#), [drawing pages](#), and [web drawing](#) of a **VSDX** web drawing. [<1>](#) The syntax of this data structure is an XML fragment. Elements of this fragment are listed in the following sections.

The set of elements returned is determined by the contents of the web drawing, as described in the following table.

Element	Description
Page	The information about the requested drawing page and the information about additional pages in the web drawing.
Pages	A list of published foreground drawing pages in the web drawing.
PageInfo	The information about the identity of a drawing page in the web drawing.
ShapeInfo	The information about the relevant shape, including shape data , hyperlinks, text and identifying information. A relevant shape is selectable, as specified in [MS-VSDX] section 2.2.3.4, and contains hyperlinks, shape data items, or comments .
Hyperlinks	The list of each visible hyperlink element associated with a shape.
Hyperlink	The hyperlink associated with a shape.
ShapeDataItems	The list of visible shape data items associated with a shape.
ShapeData	The shape data item associated with a shape.
Shapes	The list of geometric outline information for each relevant shape in the web drawing. A relevant shape is selectable, as specified in [MS-VSDX] section 2.2.3.4, and contains hyperlinks, shape data items, or comments.
Shape	The geometric outline information for a shape.
Path	The geometric outline information in the form of a path.

2.2.9.1.1 Page

A **Page** is a complex type that specifies information about the [shapes](#) on a [drawing page](#) in a [web drawing](#). One element is returned for the web drawing.

```
<xs:complexType name="Page" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element name="Pages" minOccurs="1" maxOccurs="1" type="Pages"/>
    <xs:element name="ShapeInfo" minOccurs="0" maxOccurs="unbounded" type="ShapeInfo"/>
    <xs:element name="Shapes" minOccurs="0" maxOccurs="1" type="Shapes"/>
  </xs:sequence>
  <xs:attribute name="ContentBounds" type="xs:string" use="required"/>
  <xs:attribute name="DataBindingConflict" type="xs:boolean" use="required"/>
  <xs:attribute name="DocumentID" type="xs:string" use="required"/>
  <xs:attribute name="DocumentTimeStamp" type="xs:string" use="required"/>
  <xs:attribute name="Name" type="xs:string" use="optional"/>
  <xs:attribute name="ReadOnly" type="xs:boolean" use="required"/>
  <xs:attribute name="RefreshWarning" type="xs:boolean" use="required"/>
  <xs:attribute name="RequiresCheckout" type="xs:boolean" use="optional"/>
  <xs:attribute name="UnsupportedFeatures" type="xs:int" use="optional"/>
  <xs:attribute name="ViewOnly" type="xs:boolean" use="required"/>
</xs:complexType>
```

ContentBounds: A string that specifies the bounding rectangle of the drawing page as specified in the following table.

Name	Description
x	An integer that specifies the upper left x-coordinate of the bounding rectangle.
y	An integer that specifies the upper left y-coordinate of the bounding rectangle.
width	A positive integer that specifies the width of the bounding rectangle.
height	A positive integer that specifies the height of the bounding rectangle.

The coordinates of a bounding rectangle **MUST** be relative to the top, left corner of the window that contains the drawing page. All the values **MUST** be specified in **pixels**.

The string **MUST** be formatted using the JavaScript Object Notation as specified in [\[RFC4627\]](#) as follows:

```
{"x":x-val,"y":y-val,"width":width-val,"height":height-val}
```

where **x-val**, **y-val**, **width-val**, and **height-val** are integers specifying the x, y, width, and height fields, respectively.

DataBindingConflict: An integer that specifies the status of retrieving data from data providers for the drawing page. The value **MUST** be one of the following:

Value	Description
0	The drawing page does not have any data binding conflicts.
1	The drawing page does have data binding conflicts.

DocumentID: A string that specifies a **GUID** that identifies the file in a document repository that contains the web drawing being requested.

DocumentTimeStamp: A string that specifies an opaque identifier for the version of the web drawing. It MUST NOT be an empty string.

Name: A string that specifies the name of the drawing page. It MUST NOT be an empty string. It MUST match the **Name** or **NameU** attribute of the [Page_Type](#) elements in the [Pages XML Part](#), as specified in [\[MS-VSDX\]](#) section 2.2.2.1, of the web drawing. It MUST match a name in the **Name** attribute of one of the **PageInfo** child elements of the **Pages** element.

ReadOnly: A boolean that specifies the read-only state of the file in the document repository. A value of one indicates that the file is read-only; a value of zero indicates the file is not read-only.

RefreshWarning: An optional boolean that specifies whether the drawing page was refreshed by the protocol server. A nonzero value indicates that data connections exist for the drawing page but that the drawing page was not refreshed from the data providers associated with these data connections.

RequiresCheckout: A boolean that specifies if the file requires check-out from the document repository. A value of one indicates that the file requires check-out; a value of zero indicates that the file does not require check-out.

UnsupportedFeatures: An integer that specifies features in the file that are not supported by the server. **UnsupportedFeatures** MUST be zero, or a bitwise OR combination of one or more of the following values.

Value	Description
0x01	The file contains legacy ink.
0x02	The file contains one or more reviewer pages.
0x04	The file contains macros.
0x08	The file contains one or more embedded objects.
0x10	The file contains unsupported cell recalculations.

ViewOnly: A boolean that specifies if the file can be viewed but not opened from the document repository. A value of one indicates that the file is view only and cannot be opened; a value of zero indicates the file is not view only.

2.2.9.1.2 Pages

The **Pages** is a complex type that specifies all the foreground [drawing pages](#) in the [web drawing](#). One element is returned in the **Page** element of the web drawing.

```
<xs:complexType name="Pages" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element name="PageInfo" minOccurs="1" maxOccurs="unbounded" type="PageInfo"/>
  </xs:sequence>
</xs:complexType>
```

2.2.9.1.3 PageInfo

The **Page Info** is a complex type that specifies the identity of a [drawing page](#) in the [web drawing](#). One element is returned for each [foreground](#) page within the **Pages** element.

```
<xs:complexType name="PageInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:attribute name="ID" type="xs:unsignedLong" use="required"/>
  <xs:attribute name="Name" type="xs:string" use="required"/>
</xs:complexType>
```

ID: An unsigned long that specifies the identifier of the [drawing page](#). It MUST match the **ID** attribute of one of the [Page Type](#) elements in the [Pages XML Part](#), as specified in [\[MS-VSDX\]](#) section 2.2.2.1, of the web drawing.

Name: A string that specifies the name of the drawing page. It MUST NOT be an empty string. It MUST match the **Name** or **NameU** attribute of the [Page Type](#) elements in the [Pages XML Part](#), as specified in [\[MS-VSDX\]](#) section 2.2.2.1, of the web drawing, referenced by the **ID** attribute.

2.2.9.1.4 ShapeInfo

The **ShapeInfo** is a complex type that specifies information about a [shape](#). One element is returned for each relevant shape in the **Page** element of the [web drawing](#).

```
<xs:complexType name="ShapeInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element name="ShapeDataItems" minOccurs="0" maxOccurs="1" type="ShapeDataItems"/>
    <xs:element name="Hyperlinks" minOccurs="0" maxOccurs="1" type="Hyperlinks"/>
  </xs:sequence>
  <xs:attribute name="Name" type="xs:string" use="required"/>
  <xs:attribute name="DisplayName" type="xs:string" use="optional"/>
  <xs:attribute name="Guid" type="xs:string" use="optional"/>
  <xs:attribute name="Layout" type="xs:string" use="required"/>
  <xs:attribute name="ShapeText" type="xs:string" use="optional"/>
  <xs:attribute name="TabOrderString" type="xs:string" use="required"/>
</xs:complexType>
```

Name: A string that specifies the identifier of the shape. It MUST be the **ID** attribute of a shape, as specified in the [ShapeSheet_Type](#), as specified in [\[MS-VSDX\]](#) section 2.3.4.2.82, of the web drawing. It MUST NOT be an empty string, and it MUST be unique amongst all the identifiers specified by the **Name** attribute of the **ShapeInfo** elements in the containing **Page** element.

DisplayName: A string that specifies the display name for the shape. It MUST be the **Name** attribute of a shape, as specified in the [ShapeSheet_Type](#), as specified in [\[MS-VSDX\]](#) section 2.3.4.2.82, of the web drawing, of the shape specified by the **Name** attribute of this **ShapeInfo** element.

Guid : An optional string that specifies a GUID for the shape. If this attribute does not exist, the shape does not have a GUID. If this attribute exists, it MUST be the **UniqueID** attribute of a shape, as specified in the [ShapeSheet_Type](#), as specified in [\[MS-VSDX\]](#) section 2.3.4.2.82, of the web drawing, of the shape specified by the **Name** attribute of this **ShapeInfo** element.

Layout: A string that specifies the geometric location of the shape. If the shape is 2D, the string specifies the bounding rectangle of the shape on the [drawing page](#) as specified in the following table.

Name	Description
x	An integer that specifies the upper left x-coordinate of the bounding rectangle.
y	An integer that specifies the upper left y-coordinate of the bounding rectangle.
width	A positive integer that specifies the width of the bounding rectangle.
height	A positive integer that specifies the height of the bounding rectangle.

The string MUST be formatted using the JavaScript Object Notation as specified in [\[RFC4627\]](#) as follows:

```
{"type":"2D","bounds":{"x":x-val,"y":y-val,"width":width-val,"height":height-val}}
```

where **x-val**, **y-val**, **width-val**, and **height-val** are integers specifying the x, y, width and height fields, respectively.

If the shape is 1D, the string specifies the begin and end coordinates of the shape as specified in the following table.

Name	Description
beginX	An integer that specifies the x-coordinate of the beginning of the 1D shape.
beginY	An integer that specifies the y-coordinate of the beginning of the 1D shape.
endX	An integer that specifies the x-coordinate of the end of the 1D shape.
endY	An integer that specifies the y-coordinate of the end of the 1D shape.

The string MUST be formatted using the JavaScript Object Notation as specified in [\[RFC4627\]](#) as follows:

```
{"type":"1D","points":{"beginX":beginx-val,"beginY":beginy-val, "":{"endX":endx-val,"endY":endy-val}}
```

where **beginx-val**, **beginy-val**, **endx-val**, and **endy-val** are integers specifying the beginX, beginY, endX and endY fields, respectively.

The coordinates MUST be relative to the top, left corner of the window that contains the drawing page. All the values MUST be specified in pixels.

ShapeText: An optional string that specifies a text element. If this attribute exists, it is composed of information from the document, shape data items, recalculations on shape data items and formatting information, for a shape.

TabOrderString: A string that specifies the tab order of this shape on the drawing page. The format of the string MUST be "rowIndex_columnIndex", where rowIndex and columnIndex are unsigned integer values separated by an underscore "_".

2.2.9.1.5 Hyperlinks

The **Hyperlinks** is a complex type that specifies each **Hyperlink** element (section [2.2.9.1.6](#)) associated with a **shape**. One element is returned in each **ShapeInfo** element (section [2.2.9.1](#)) that contains a visible hyperlink.

```
<xs:complexType name="Hyperlinks" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element name="Hyperlink" minOccurs="1" maxOccurs="unbounded" type="Hyperlink"/>
  </xs:sequence>
</xs:complexType>
```

2.2.9.1.6 Hyperlink

The **Hyperlink** is a complex type that specifies a hyperlink associated with a **shape**. One element is returned for each visible hyperlink associated with the shape within a **ShapeInfo** element (section [2.2.9.1](#)).

```
<xs:complexType name="Hyperlink" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:attribute name="Name" type="xs:string" use="required"/>
  <xs:attribute name="Value" type="xs:string" use="optional"/>
  <xs:attribute name="Description" type="xs:string" use="optional"/>
  <xs:attribute name="SubAddress" type="xs:string" use="optional"/>
  <xs:attribute name="SubAddressShape" type="xs:string" use="optional"/>
  <xs:attribute name="Zoom" use="optional">
    <xs:simpleType>
      <xs:union>
        <xs:simpleType>
          <xs:restriction base="xs:int">
            <xs:minExclusive value="0"/>
          </xs:restriction>
        </xs:simpleType>
        <xs:simpleType>
          <xs:restriction base="xs:int">
            <xs:enumeration value="-2"/>
            <xs:enumeration value="-1"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:union>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="Default" type="xs:boolean" use="optional"/>
</xs:complexType>
```

Name: A string attribute that specifies an identifier for a hyperlink. It MUST NOT be an empty string, and it MUST be unique amongst the identifiers specified by the **Name** attribute of the hyperlink elements in this XML part.

Value: An optional string attribute that specifies the URI of an external resource referenced by the hyperlink. If this attribute exists, it MUST NOT be an empty string, and the **SubAddress**, **SubAddressShape**, and **Zoom** elements MUST be ignored. If this attribute does not exist, **SubAddress** MUST exist.

Description: An optional string attribute that specifies the description of the hyperlink. If this attribute does not exist, the hyperlink does not have a description.

SubAddress: An optional string attribute that specifies the identifier of the [drawing page](#) referenced by this hyperlink within the [web drawing](#). If this attribute exists, it MUST NOT be an empty string. If this attribute does not exist, the **SubAddressShape** and **Zoom** elements MUST be ignored.

SubAddressShape: An optional string attribute that specifies the identifier of the shape referenced by this hyperlink within the drawing page specified by **SubAddress**. If this attribute exists, it MUST NOT be an empty string. If this attribute does not exist, the hyperlink references the entire drawing page.

Zoom: An optional integer attribute that specifies the **zoom level** for the drawing page specified by **SubAddress**. If this attribute does not exist, the zoom level is 100%. If this attribute does exist, it MUST be a value from the following table.

Value	Description
-2	The drawing page is zoomed to fit the width of the view.
-1	The drawing page is zoomed to fit in the view.
n > 0	The drawing page is zoomed to n%. n MUST be greater than 0.

Default: An optional boolean attribute that specifies the default hyperlink associated with the shape. It MUST exist if and only if this hyperlink is the default. Each shape MUST NOT have more than one default hyperlink. If this attribute exists, the value MUST be one.

2.2.9.1.7 ShapeDataItems

The **ShapeDataItems** is a complex type that specifies a list of [shape](#) data items, as specified in [\[MS-VSDX\]](#) section 2.2.3.6. One element is returned in each **ShapeInfo** element (section [2.2.9.1](#)) that contains a visible [shape data item](#).

```
<xs:complexType name="ShapeDataItems" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element name="ShapeData" minOccurs="1" maxOccurs="unbounded" type="ShapeData"/>
  </xs:sequence>
</xs:complexType>
```

2.2.9.1.8 ShapeData

The **ShapeData** is a complex type that specifies a [shape](#) data item, as specified in [\[MS-VSDX\]](#) section 2.2.3.6. One element is returned for each visible [shape data item](#) associated with the shape with a **ShapeInfo** element (section [2.2.9.1](#)).

```
<xs:complexType name="ShapeData" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:attribute name="Name" type="xs:string" use="required"/>
  <xs:attribute name="FormattedValue" type="xs:string" use="required"/>
  <xs:attribute name="Value" type="xs:string" use="optional"/>
  <xs:attribute name="Format" type="xs:string" use="optional"/>
</xs:complexType>
```

Name: A string that is determined by a combination of shape data row properties that specifies the label of the shape data item.

FormattedValue: A string representation of the value of the data formatted for display. It MUST NOT be an empty string.

Value: An optional string that specifies the value of the shape data item, as specified in [\[MS-VSDX\]](#) section 2.2.3.6.

Format: An optional string that specifies a vFormatString custom structure, as specified in [\[MS-VSDX\]](#) section 2.5.9.6. If this attribute exists, it is used to format the value for display.

2.2.9.1.9 Shapes

The **Shapes** is a complex type that specifies the geometric outline information for multiple [shapes](#). One element is returned for the [web drawing](#).

```
<xs:complexType name="Shapes" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element name="Shape" minOccurs="1" maxOccurs="unbounded" type="Shape"/>
  </xs:sequence>
</xs:complexType>
```

2.2.9.1.10 Shape

The **Shape** is a complex type that specifies the geometric outline information for a [shape](#). One element is returned for each relevant shape.

```
<xs:complexType name="Shape" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element name="Path" minOccurs="1" maxOccurs="1" type="Path"/>
  </xs:sequence>
  <xs:attribute name="Name" type="xs:string" use="required"/>
</xs:complexType>
```

Name: A string that specifies the identifier of the **ShapeInfo** element (section [2.2.9.1](#)) in the **Page** structure (section [2.2.9.1.1](#)) that this element represents. It MUST match exactly the **Name** of a **ShapeInfo** element in the **Pages** element of this **Page** structure.

2.2.9.1.11 Path

The **Path** is a complex type that specifies the geometric outline information for a [shape](#). One element is returned for a **Shape** element.

```
<xs:complexType name="Path" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:attribute name="Shape" type="xs:string" fixed="poly"/>
  <xs:attribute name="Data" use="required">
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:pattern value="\d*(,\d*){3,}"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:attribute>
</xs:complexType>
```

Shape: A string that specifies the type of outline. It MUST be equal to "poly".

Data: A string that specifies the Cartesian coordinates of the outline. It MUST have the following format:

```
x1, y1, x2, y2 . . . xn, yn
```

The fields x_n and y_n MUST be integer values, and they MUST occur in pairs. Each x_n, y_n pair specifies the coordinates of a point that defines one vertex of the outline. The outline is formed by connecting the points in the order they appear in this string attribute. There MUST be at least 4 points. The first point MUST equal the last point. The coordinates MUST be measured in pixels relative to the top, left corner of the [drawing page](#) that contains the shape. All values MUST be separated by commas.

2.2.9.2 Comments

The **Comments** element is a string that specifies information about the [comments](#) in a [drawing page](#).<2> The syntax of this data structure is an XML fragment. Elements of this fragment are listed in the following sections.

2.2.9.2.1 Comments

The **Comments** is a complex type that specifies all [comment](#) information for a [drawing page](#). There MUST be zero or exactly one instance of this element.

```
<xs:complexType name="comments" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="unbounded" name="CommentEntry"/>
  </xs:sequence>
</xs:complexType>
```

The **Comments** element MUST contain one or more **CommentEntry** elements as specified in section [2.2.9.2.2](#).

2.2.9.2.2 CommentEntry

The **CommentEntry** complex type specifies a [comment](#) on a [drawing page](#).

```
<xs:complexType name="CommentEntry" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleContent>
    <xs:extension base="xsd:string">
      <xs:attribute name="CommentID" type="xs:string" use="required">
      <xs:attribute name="Author" type="xs:string" use="required">
      <xs:attribute name="ShapeID" type="xs:string" use="required">
      <xs:attribute name="CreationDate" type="xs:string" use="required">
      <xs:attribute name="ModifiedDate" type="xs:string" use="required">
      <xs:attribute name="Done" type="xs:string" use="required">
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

CommentID: An integer attribute that specifies the identifier of this comment. It MUST be greater than or equal to zero. It MUST be unique across all the **CommentEntry** elements in the containing **Comments** element (section [2.2.9.2](#)). It MUST match the value of a CommentID attribute in a

CommentEntry_Type element, as specified in [\[MS-VSDX\]](#) section 2.3.4.2.9, in the containing drawing page.

Author: A string attribute that specifies the display name of the author of this comment. The value MUST match the value of the **Name** attribute of an AuthorEntry_Type as specified [\[MS-VSDX\]](#) section 2.3.4.2.2.

ShapeID: An integer attribute that specifies the identifier of the [shape](#) this comment is associated with. If no **ShapeID** is specified or if **ShapeID** is zero, the comment refers to the [drawing page](#). If the attribute exists and is not zero, it MUST match the value of the **ID** attribute of a shape, as specified in the ShapeSheet_Type, as specified in [\[MS-VSDX\]](#) section 2.3.4.2.82, of the [web drawing](#).

CreationDate: A dateTime attribute that specifies the date and time, in Coordinated Universal Time (UTC) format, of when the comment was created.

ModifiedDate: A dateTime attribute that specifies the date and time, in Coordinated Universal Time (UTC) format, of when the comment was last modified. MUST be greater than or equal to the value of CreationDate.

Done: A boolean attribute that specifies the current state of the comment. A value of "true" indicates that the comment is completed; a value of "false" indicates that the comment is not completed.

2.2.9.3 CommentData

The **CommentData** attribute is a string that specifies a save comment action and the [comment](#) data for a shape or [drawing page](#) in the **VSDX web drawing**.<3> The syntax of this data structure is an XML fragment. Elements of this fragment are listed in the following sections.

2.2.9.3.1 Comment

The **Comment** is a complex type that specifies a [comment](#) for a shape or [drawing page](#) in a **VSDX web drawing**. There MUST be exactly one instance of this element, and it MUST NOT be null.

The value of the **Comment** element is a string with the comment text. Additional attributes define properties and actions for the comment.

```
<xs:complexType name="Comment" xmlns:xs="http://www.w3.org/2001/XMLSchema">
```

```
  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="Action" type="xs:string" use="required">
      <xs:attribute name="PageID" type="xs:integer" use="required">
      <xs:attribute name="ShapeID" type="xs:integer">
      <xs:attribute name="CommentID" type="xs:integer">
      <xs:attribute name="Author" type="xs:string">
      <xs:attribute name="ModifiedDate" type="xs:dateTime">
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Action: A string attribute that specifies the action to perform. It MUST exist, and it MUST be one of the following values.

Value	Description
"Add"	Write a new comment into the file using the contents and attributes of this element.
"Edit"	Edit an existing comment by modifying the file using the contents and attributes of this element.
"Delete"	Remove a comment from the file.
"Done"	Mark an existing comment as complete in the file.

PageID: An integer attribute that identifies the [drawing page](#) in the **VSDX** web drawing specified by **fileUrl** that the comment is on.

ShapeID: An integer attribute that specifies the identifier of the [shape](#), within the [drawing page](#) specified by **PageID**, this comment is associated with. If no **ShapeID** is specified or if **ShapeID** is zero, the comment refers to the [drawing page](#) specified by **PageID**.

CommentID: An integer attribute that specifies the identifier of the comment. If the **Action** value is "Edit", "Delete" or "Done", this attribute **MUST** exist, and the value **MUST** match the value of the **CommentID** attribute of a **CommentEntry_Type**, as specified in [\[MS-VSDX\]](#) section 2.3.4.2.9, in the **VSDX** web drawing specified by **fileUrl**.

Author: A string attribute that specifies the display name of the author of this comment. If the **Action** value is "Add" or "Edit", this attribute **MUST** exist, and it **MUST NOT** be empty.

ModifiedDate: A **dateTime** attribute that specifies the date and time, in Coordinated Universal Time (UTC) format, of when the comment was last modified. If the **Action** value is "Delete" or "Edit", this attribute **MUST** exist, and it **MUST NOT** be null.

2.2.9.4 CommentResult

The **CommentResult** element is a string that specifies the result of the **SaveComments** operation (section [3.1.4.5](#)). **<4>** The syntax of this data structure is an XML fragment. Elements of this fragment are listed in the following sections.

2.2.9.4.1 Comment

The **Comment** is a complex type that specifies the result of a [comment](#) action in the **VSDX web drawing**. The value of the **Comment** element is a string with comment text. Additional attributes define the properties and actions for the comment response.

```
<xs:complexType name="Comment" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:simpleContent>
    <xs:extension base="xsd:string">
      <xs:attribute name="Action" type="xs:string" use="required">
      <xs:attribute name="ShapeID" type="xs:integer">
      <xs:attribute name="CommentID" type="xs:integer">
      <xs:attribute name="Author" type="xs:string">
      <xs:attribute name="CreationDate" type="xs:dateTime">
      <xs:attribute name="ModifiedDate" type="xs:dateTime">
      <xs:attribute name="Done" type="xs:boolean">
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
```

Action: A string attribute that specifies the action performed. It MUST exist, and it MUST be one of the following values:

Value	Description
"Add"	Add a new comment.
"Edit"	Edit an existing comment.
"Delete"	Delete an existing comment.
"Done"	Mark an existing comment as complete.

ShapeID: An integer attribute that specifies the identifier of a [shape](#) that this comment is associated with. If no **ShapeID** is specified or if **ShapeID** is zero, the comment refers to the [drawing page](#).

CommentID: An integer attribute that specifies the identifier of the comment. If the **Action** value is "Add" or "Edit", this attribute MUST exist, and the value MUST match the value of the **CommentID** attribute of a **CommentEntry_Type**, as specified in [\[MS-VSDX\]](#) section 2.3.4.2.9, in the **VSDX** web drawing specified by **fileUrl**.

Author: A string attribute that specifies the display name of the author of this comment. If the **Action** value is "Add", this attribute MUST exist, and it MUST NOT be empty.

CreationDate: A dateTime attribute that specifies the date and time, in Coordinated Universal Time (UTC) format, of when the comment was created. If the **Action** value is "Add" or "Edit", this attribute MUST exist, and it MUST NOT be null.

ModifiedDate: A dateTime attribute that specifies the date and time, in Coordinated Universal Time (UTC) format, of when the comment was last modified. If the **Action** value is "Add" or "Edit", this attribute MUST exist, and it MUST NOT be null.

Done: A boolean attribute that specifies the current state of the comment. If the **Action** value is "Done", this attribute MUST exist. A value of "true" indicates that the comment is completed; a value of "false" indicates that the comment has not been completed.

3 Protocol Details

In the following sections, the schema definition might differ from the processing rules imposed by the protocol. The WSDL in this specification matches the WSDL that shipped with the product, and provides a base description of the schema. The text that introduces the WSDL might specify differences that reflect actual Microsoft product behavior. For example, the schema definition might allow for an element to be empty, null, or not present but the behavior of the protocol as specified restricts the same elements to being non-empty, not null, and present.

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

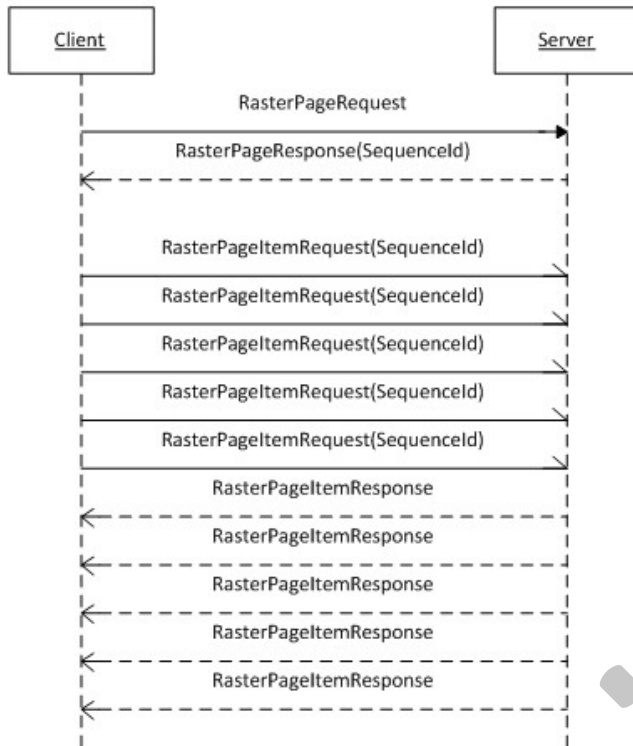
This protocol allows protocol servers to notify protocol clients of application-level faults using SOAP faults. Except where specified, these SOAP faults are not significant for interoperability, and protocol clients can interpret them in an implementation-specific manner.

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

3.1 Common Details

The **GetRasterDiagram** (section [3.1.4.1](#)) and **GetVectorDiagram** (section [3.1.4.2](#)) SDL operations are independent and stateless. The protocol server returns the requested [drawing page](#) of the specified **VGSFF** [web drawing](#) in raster and vector format, respectively. There are no interaction dependencies between instances of client and server.

The **GetRasterPage** (section [3.1.4.3](#)) and **GetRasterPageItem** (section [3.1.4.4](#)) WSDL operations are related and require client state. A drawing page of a **VSDX** web drawing in raster format is retrieved in full as a result of a message sequence, as illustrated in the following high-level sequence diagram:



First, the protocol client sends an **IVisioGraphicsService_GetRasterPage_InputMessage** request (section [3.1.4.3.1.1](#)) **WSDL message** that MUST be unconditional, as specified in section [3.1.1](#). The protocol server initiates the process of retrieving or generating the drawing page. The client waits for the server to respond with the **IVisioGraphicsService_GetRasterPage_OutputMessage** (section [3.1.4.3.1.2](#)) WSDL message that contains drawing page information and includes a message sequence identifier. This message sequence identifier is set in the **SequenceId** element of the **RasterPageTag** complex type (section [3.1.4.3.3.3](#)) and is contained in the **EntityTag** element of the **RasterPageResponse** complex type (section [3.1.4.3.3.2](#)).

Then, the protocol client sends a sequence of **IVisioGraphicsService_GetRasterPageItem_InputMessage** request (section [3.1.4.4.1.1](#)) WSDL messages, one for each item, as specified in section [3.1.1](#), of the drawing page. The client MUST set the value of the **SequenceId** element of the **RasterPageItemRequest** complex type (section [3.1.4.4.3.1](#)), to the message sequence identifier obtained in the preceding **IVisioGraphicsService_GetRasterPage_OutputMessage** response WSDL message. These requests are independent and can be sent asynchronously in any order. The protocol server uses the message sequence identifier to match the requests with the drawing page retrieved or generated as result of the **IVisioGraphicsService_GetRasterPage_InputMessage** request WSDL message that initiated the conversation. The server responds to each of these requests with a corresponding **IVisioGraphicsService_GetRasterPageItem_OutputMessage** response (section [3.1.4.4.1.2](#)) WSDL message that contains information about the requested item, as specified in section [3.1.1](#), of the drawing page.

The **SaveComments** WSDL operation (section [3.1.4.5](#)) is independent and stateless. The protocol server makes the requested [comment](#) updates and saves them to the specified web drawing. This web drawing file MUST conform to the format specified in [\[MS-VSDX\]](#) section 2.2.9.

3.1.1 Abstract Data Model

There is no abstract data model for a **VGSFF** [web drawing](#).

A [drawing page](#) of a **VSDX** web drawing in raster format is composed of 5 **items**: 3 images in Portable Network Graphics (PNG) format, as specified in [\[RFC2083\]](#), the [shape](#) information in XML format, as specified in section [2.2.9.1](#), and [comments](#) in XML format, as specified in section [2.2.9.2.<5>](#) Each image represents the drawing page in different sizes, as specified by the **ItemName** element of the **RasterPageItemRequest** complex type (section [3.1.4.4.3.1](#)).

An **IVisioGraphicsService_GetRasterPage_InputMessage** request (section [3.1.4.3.1.1](#)) or an **IVisioGraphicsService_GetRasterPageItem_InputMessage** request (section [3.1.4.4.1.1](#)) WSDL message is unconditional if the value of the **EntityTag** element of the corresponding complex types **RasterPageRequest** (section [3.1.4.3.3.1](#)) and **RasterPageItemRequest** (section [3.1.4.4.3.1](#)), respectively, is null or empty, as specified by the **RasterPageTag** complex type (section [3.1.4.3.3.3](#)) and the **RasterPageItemTag** complex type (section [3.1.4.4.3.2](#)) respectively. Otherwise, the requesting WSDL message is conditional. Message sequencing rules for conditional messages are specified in [3.1.4](#).

After the initial request for a drawing page of a **VSDX** web drawing is processed and before returning the result to the protocol client, the protocol server caches it for future use. Subsequent requests for the same drawing page are served directly from this cache. The protocol server assigns a unique identifier, **OutputETag** element, to each version of a drawing page in the cache. The protocol server also assigns a unique identifier, **FileETag** element, for the current version of the web drawing that contains this drawing page in the cache. The **OutputETag** and **FileETag** are included in the **EntityTag** element of the **RasterPageResponse** complex type (section [3.1.4.3.3.2](#)). The **EntityTag** received by a client can be used on subsequent conditional requests to determine if the version of the drawing page cached on the client needs to be updated. The protocol server compares the **FileETag** and **OutputETag** from the **EntityTag** of the conditional **RasterPageTag** with the current **FileETag** and **OutputETag**, respectively, of the cached drawing page. If both elements match, the server returns a **RasterPageResponse** having the response code "NotModified"; otherwise, it generates a new drawing page.

3.1.2 Timers

The **Recalculation Timeout** timer measures the time it takes for the protocol server to recalculate the [drawing page](#) of a [web drawing](#). This timeout is configured on the protocol server.

The **Message Sequence Timeout** timer measures the time it takes for the protocol server to receive an **IVisioGraphicsService_GetRasterPageItem_InputMessage** request (section [3.1.4.4.1.1](#)) WSDL message beginning when the **IVisioGraphicsService_GetRasterPage_OutputMessage** response (section [3.1.4.3.1.2](#)) WSDL message is sent, in the sequence of messages to retrieve a drawing page of a **VSDX** web drawing in raster format, as specified in section [3.1.<6>](#) This timeout is set to one minute, and it is not configurable on the protocol server.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

The protocol operations can be executed in any order with the following restrictions:

- To fully retrieve a [drawing page](#) of a **VSDX web drawing** in raster format, the protocol client **MUST** follow the message sequence specified and illustrated in section [3.1](#).
- The protocol client can cache a fully retrieved drawing page of a **VSDX web drawing** in raster format. Furthermore, it can use the information set by the protocol server in the **EntityTag** element of the **RasterPageResponse** complex type (section [3.1.4.3.3.2](#)) and send it back to the server in the **EntityTag** element of a subsequent **RasterPageRequest** complex type (section [3.1.4.3.3.1](#)) in a conditional message. Based on that information, if the server decides that the drawing page has not changed, it responds with a **RasterPageResponse** having the response code "NotModified". Similarly the client can use the information set by the server in the **EntityTag** element of a **RasterPageItemResponse** complex type (section [3.1.4.4.3.3](#)) and send it back to the server in the **EntityTag** element of a subsequent **RasterPageItemRequest** complex type (section [3.1.4.4.3.1](#)) in a conditional message. If the server decides that the requested item, as specified in section [3.1.1](#), of the drawing page has not changed, it responds with a **RasterPageItemResponse** having the response code "NotModified". In these cases, the server expects the client will use the drawing page data that it cached previously. This mechanism reduces network traffic.
- If a **GetRasterPageItem** operation (section [3.1.4.4](#)) is not preceded by a **GetRasterPage** operation (section [3.1.4.3](#)) or the protocol server cannot match the value of the **SequenceId** element of the **RasterPageItemRequest**, as specified in section [3.1](#), it **MUST** return a **RasterPageItemResponse** with the response code "NotModified".

The following table summarizes the list of operations as defined by this specification:

Operation	Description
GetRasterDiagram	The GetRasterDiagram operation retrieves a drawing page from the VGSFF web drawing rendered in a raster format.
GetRasterPage	The GetRasterPage operation retrieves information about a drawing page of a VSDX web drawing rendered in raster format. <7>
GetRasterPageItem	The GetRasterPageItem operation retrieves an item, as specified in section 3.1.1 , of a drawing page of a VSDX web drawing rendered in raster format. <8>
GetVectorDiagram	The GetVectorDiagram operation retrieves a drawing page from the VGSFF web drawing rendered in a vector format.
SaveComments	The SaveComments operation saves a comment to a VSDX web drawing in a document repository, in the form of a CommentEntry_Type element as specified [MS-VSDX] section 2.3.4.2.9. <9>

3.1.4.1 GetRasterDiagram

The **GetRasterDiagram** operation retrieves a [drawing page](#) from the **VGSFF web drawing** rendered in a raster format.

```
<wsdl:operation name="GetRasterDiagram" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterDiagram"
    message="tns2:IVisioGraphicsService_GetRasterDiagram_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
```

```

    <wsdl:output
wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterDiagramResponse"
message="tns2:IVisioGraphicsService_GetRasterDiagram_OutputMessage"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
    <wsdl:fault
wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterDiagramVisioGraphicsServiceFaultFault" name="VisioGraphicsServiceFaultFault"
message="tns2:IVisioGraphicsService_GetRasterDiagram_VisioGraphicsServiceFaultFault_FaultMessage" xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends an **IVisioGraphicsService_GetRasterDiagram_InputMessage** request (section [3.1.4.1.1.1](#)) WSDL message, and the protocol server MUST respond with an **IVisioGraphicsService_GetRasterDiagram_OutputMessage** response (section [3.1.4.1.1.2](#)) WSDL message as follows:

1. If the protocol server determines that an error occurred during the protocol server operation, the protocol server MUST respond with a **VisioGraphicsServiceFault** complex type (section [2.2.4.5](#)) SOAP fault with the **Error** field set to the appropriate **FaultCode** value (section [2.2.5.1](#)).
2. Otherwise, the protocol server MUST return a **RasterDiagram** complex type (section [3.1.4.1.3.2](#)).

If the requested drawing page contains data providers to be implemented by the protocol client, the following sequence of messages MUST be followed:

1. The protocol server MUST respond with a **IVisioGraphicsService_GetRasterDiagram_OutputMessage** response WSDL message. The data providers MUST be specified in the **DataSources** element of the **RasterDiagram** contained within this response. The **Image** element of this **RasterDiagram** MUST be null.
2. The protocol client MUST retrieve the data from the data providers and send another **IVisioGraphicsService_GetRasterDiagram_InputMessage** request WSDL message. This request MUST be identical to the original request except that the **DataSources** element of the **RasterDiagramRequestContract** complex type (section [3.1.4.1.3.1](#)) MUST NOT be null.
3. The protocol server MUST respond with a **IVisioGraphicsService_GetRasterDiagram_OutputMessage** response WSDL message. If no errors occurred, the response MUST contain a **RasterDiagram** and the **Image** element of the **RasterDiagram** MUST NOT be null.

3.1.4.1.1 Messages

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

Message	Description
IVisioGraphicsService_GetRasterDiagram_InputMessage	The request WSDL message for the GetRasterDiagram WSDL operation.
IVisioGraphicsService_GetRasterDiagram_OutputMessage	The response WSDL message for the GetRasterDiagram

Message	Description
	m WSDL operation.
IVisioGraphicsService_GetRasterDiagram_VisioGraphicsServiceFaultFault_FaultMessage	The fault WSDL message for the GetRasterDiagram m WSDL operation.

3.1.4.1.1.1 IVisioGraphicsService_GetRasterDiagram_InputMessage

The request WSDL message for the **GetRasterDiagram** WSDL operation.

The **SOAP action** value is:

```
http://tempuri.org/IVisioGraphicsService/GetRasterDiagram
```

The **SOAP body** contains the **GetRasterDiagram** element.

3.1.4.1.1.2 IVisioGraphicsService_GetRasterDiagram_OutputMessage

The response WSDL message for the **GetRasterDiagram** WSDL operation.

The SOAP body contains the **GetRasterDiagramResponse** element.

3.1.4.1.1.3

IVisioGraphicsService_GetRasterDiagram_VisioGraphicsServiceFaultFault_FaultMessage

The fault WSDL message for the **GetRasterDiagram** WSDL operation.

The SOAP body contains the **VisioGraphicsServiceFault** element.

3.1.4.1.2 Elements

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

Element	Description
GetRasterDiagram	The input data for the GetRasterDiagram WSDL operation.
GetRasterDiagramResponse	The result data for the GetRasterDiagram WSDL operation.

3.1.4.1.2.1 GetRasterDiagram

The input data for the **GetRasterDiagram** WSDL operation.

```
<xs:element name="GetRasterDiagram" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element
        xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer"
        minOccurs="0" name="rasterDiagramRequestContract" nillable="true"
        type="tns:RasterDiagramRequestContract"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```

    </xs:sequence>
  </xs:complexType>
</xs:element>

```

rasterDiagramRequestContract: The request parameters for a **GetRasterDiagram** WSDL operation (section [3.1.4.1](#)). There MUST be exactly one instance of this element.

3.1.4.1.2.2 GetRasterDiagramResponse

The result data for the **GetRasterDiagram** WSDL operation.

```

<xs:element name="GetRasterDiagramResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element
        xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer"
        minOccurs="0" name="GetRasterDiagramResult" nillable="true" type="tns:RasterDiagram"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

GetRasterDiagramResult: The results for a **GetRasterDiagram** WSDL operation (section [3.1.4.1](#)). There MUST be exactly one instance of this element.

3.1.4.1.3 Complex Types

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

Complex type	Description
RasterDiagramRequestContract	The input data for the GetRasterDiagram WSDL operation.
RasterDiagram	The response data from the GetRasterDiagram WSDL operation.

3.1.4.1.3.1 RasterDiagramRequestContract

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer

The **RasterDiagramRequestContract** complex type contains information used to specify the request parameters for a **GetRasterDiagram** WSDL operation (section [3.1.4.1](#)). This complex type inherits from the **BaseRequestContract** complex type, as specified in section [2.2.4.3](#), which specifies additional parameters.

```

<xs:complexType name="RasterDiagramRequestContract"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:BaseRequestContract">
      <xs:sequence>
        <xs:element minOccurs="0" name="Resolution" type="xs:int"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>

```

</xs:complexType>

Resolution: An integer that is unused, and MUST be equal to 1 or 2.

3.1.4.1.3.2 RasterDiagram

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer

The **RasterDiagram** complex type contains information used to specify the response for a **GetRasterDiagram** WSDL operation (section 3.1.4.1). This complex type inherits from the **DiagramBase** complex type, as specified in section 2.2.4.4, which specifies additional parameters.

```
<xs:complexType name="RasterDiagram" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:DiagramBase">
      <xs:sequence>
        <xs:element minOccurs="0" name="AreaMaps" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="Image" nillable="true" type="xs:base64Binary"/>
        <xs:element minOccurs="0" name="ShapeInfo" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

AreaMaps: A string that specifies the geometric outline information of the shapes, as specified in section 1.3.3, in the raster image. The format of this data is specified in [RFC1980]. There MUST be exactly one instance of this element.

Image: A base64Binary that specifies the drawing page, as specified in section 1.3.2, as a raster image in PNG format, as specified in [RFC2083]. There MUST be exactly one instance of this element.

ShapeInfo: A string that specifies information about the shapes, as specified in section 1.3.3, in the drawing page, as specified in section 1.3.2. There MUST be exactly one instance of this element.

The syntax of this element is specified in [MS-VGSFF] section 2.4.8. The syntax of this element is also specified by additional attributes of the **CT_Page** element, as specified in [MS-VGSFF] section 2.4.8.2.8, that are listed in the following table.

Attribute of CT_Page element	Description
Conflict	An optional integer that specifies the status of retrieving data from data providers for the drawing page. If this attribute is present, the value MUST be equal to zero.
DocumentID	A string that specifies a GUID that identifies the file in a document repository that contains the VGSFF web drawing , being requested.
DocumentTimeStamp	A dateTime value that specifies the date and time, in Coordinated Universal Time (UTC) format, when the file in a document repository that contains the VGSFF web drawing , was last modified in the document repository.
RefreshWarning	An optional integer that specifies whether the drawing page was refreshed by the protocol server. If this attribute is present, a value of one indicates that

Attribute of CT_Page element	Description
	data connections (2) exist for the drawing page, but that the drawing page was not refreshed from the data providers associated with these data connections (2).

3.1.4.1.4 Simple Types

None.

3.1.4.1.5 Attributes

None.

3.1.4.1.6 Groups

None.

3.1.4.1.7 Attribute Groups

None.

3.1.4.2 GetVectorDiagram

The **GetVectorDiagram** operation retrieves a [drawing page](#) from the **VGSFF** [web drawing](#) rendered in a vector format.

```
<wsdl:operation name="GetVectorDiagram" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetVectorDiagram"
    message="tns2:IVisioGraphicsService_GetVectorDiagram_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetVectorDiagramResponse"
    message="tns2:IVisioGraphicsService_GetVectorDiagram_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:fault
    wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetVectorDiagramVisioGraphicsServiceFaultFault"
    name="VisioGraphicsServiceFaultFault"
    message="tns2:IVisioGraphicsService_GetVectorDiagram_VisioGraphicsServiceFaultFault_FaultMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends a **IVisioGraphicsService_GetVectorDiagram_InputMessage** request (section [3.1.4.2.1.1](#)) WSDL message, and the protocol server MUST respond with a **IVisioGraphicsService_GetVectorDiagram_OutputMessage** response (section [3.1.4.2.1.2](#)) WSDL message as follows:

1. If the protocol server determines that an error occurred during the protocol server operation, the protocol server MUST respond with a **VisioGraphicsServiceFault** complex type (section [2.2.4.5](#)) SOAP fault with the **Error** field set to the appropriate **FaultCode** value (section [2.2.5.1](#)).
2. Otherwise, the protocol server MUST return a **VectorDiagram** complex type (section [3.1.4.2.3.2](#)).

If the requested drawing page contains data providers to be implemented by the protocol client, the following sequence of messages MUST be followed:

1. The protocol server MUST respond with a **IVisioGraphicsService_GetVectorDiagram_OutputMessage** response WSDL message. The data providers MUST be specified in the **DataSources** element of the **VectorDiagram** contained within this response. The **OutputPage** element of this **VectorDiagram** MUST be null.
2. The protocol client MUST retrieve the data from the data providers and send another **IVisioGraphicsService_GetVectorDiagram_InputMessage** request WSDL message. This request MUST be identical to the original request except that the **DataSources** element of the **VectorDiagramRequestContract** complex type (section [3.1.4.2.3.1](#)) MUST NOT be null.
3. The protocol server MUST respond with a **IVisioGraphicsService_GetVectorDiagram_OutputMessage** response WSDL message. If no errors occurred, the response MUST contain a **VectorDiagram** and the **OutputPage** element of the **VectorDiagram** MUST NOT be null.

3.1.4.2.1 Messages

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

Message	Description
IVisioGraphicsService_GetVectorDiagram_InputMessage	The request WSDL message for the GetVectorDiagram WSDL operation.
IVisioGraphicsService_GetVectorDiagram_OutputMessage	The response WSDL message for the GetVectorDiagram WSDL operation.
IVisioGraphicsService_GetVectorDiagram_VisioGraphicsServiceFaultFault_FaultMessage	The fault WSDL message for the GetVectorDiagram WSDL operation.

3.1.4.2.1.1 IVisioGraphicsService_GetVectorDiagram_InputMessage

The request WSDL message for the **GetVectorDiagram** WSDL operation.

The SOAP action value is:

```
http://tempuri.org/IVisioGraphicsService/GetVectorDiagram
```

The SOAP body contains the **GetVectorDiagram** element.

3.1.4.2.1.2 IVisioGraphicsService_GetVectorDiagram_OutputMessage

The response WSDL message for the **GetVectorDiagram** WSDL operation.

The SOAP body contains the **GetVectorDiagramResponse** element.

3.1.4.2.1.3

IVisioGraphicsService_GetVectorDiagram_VisioGraphicsServiceFaultFault_FaultMessage

The fault WSDL message for the **GetVectorDiagram** WSDL operation.

The SOAP body contains the **VisioGraphicsServiceFault** element.

3.1.4.2.2 Elements

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

Element	Description
GetVectorDiagram	The input data for the GetVectorDiagram WSDL operation.
GetVectorDiagramResponse	The result data for the GetVectorDiagram WSDL operation.

3.1.4.2.2.1 GetVectorDiagram

The input data for the **GetVectorDiagram** WSDL operation.

```
<xs:element name="GetVectorDiagram" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element
        xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer" minOccurs="0" name="vectorDiagramRequest" nillable="true"
        type="tns:VectorDiagramRequestContract"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

vectorDiagramRequest: The request parameters for a **GetVectorDiagram** (section [3.1.4.2](#)) WSDL operation. There **MUST** be exactly one instance of this element, and it **MUST NOT** be null.

3.1.4.2.2.2 GetVectorDiagramResponse

The result data for the **GetVectorDiagram** WSDL operation.

```
<xs:element name="GetVectorDiagramResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element
        xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer" minOccurs="0" name="GetVectorDiagramResult" nillable="true" type="tns:VectorDiagram"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetVectorDiagramResult: The results for a **GetVectorDiagram** (section [3.1.4.2](#)) WSDL operation. There **MUST** be exactly one instance of this element, and it **MUST NOT** be null.

3.1.4.2.3 Complex Types

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

Complex type	Description
VectorDiagramRequestContract	The input data for the GetVectorDiagram (section 3.1.4.2) WSDL operation.
VectorDiagram	The response data from the GetVectorDiagram WSDL operation.
RenderedPage	The drawing page information contained in the VectorDiagram complex type (section 3.1.4.2.3.2).
ArrayOfDiagramResource	An array of resources information contained in the RenderedPage complex type (section 3.1.4.2.3.3).
DiagramResource	A font or an image required by a drawing page.

3.1.4.2.3.1 VectorDiagramRequestContract

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **VectorDiagramRequestContract** complex type contains information used to specify the request parameters for a **GetVectorDiagram** (section [3.1.4.2](#)) WSDL operation. This complex type inherits from the **BaseRequestContract** complex type (section [2.2.4.3](#)), which specifies additional parameters.

```
<xs:complexType name="VectorDiagramRequestContract"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:BaseRequestContract">
      <xs:sequence/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
```

3.1.4.2.3.2 VectorDiagram

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **VectorDiagram** complex type contains information used to specify the response for a **GetVectorDiagram** (section [3.1.4.2](#)) WSDL operation. This complex type inherits from **DiagramBase** complex type (section [2.2.4.4](#)), which specifies additional parameters.

```
<xs:complexType name="VectorDiagram" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:DiagramBase">
      <xs:sequence>
        <xs:element minOccurs="0" name="OutputPage" nillable="true" type="tns:RenderedPage"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
```

```
</xs:complexType>
```

OutputPage: A **RenderedPage** complex type (section [3.1.4.2.3.3](#)), that specifies the [drawing page](#) in vector format.

3.1.4.2.3.3 RenderedPage

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **RenderedPage** complex type contains information that specifies a drawing page, as specified in section [1.3.2](#), in a vector format.

```
<xs:complexType name="RenderedPage" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="Resources" nillable="true"
      type="tns:ArrayOfDiagramResource"/>
    <xs:element minOccurs="0" name="ShapeInfo" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Xaml" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

Resources: An **ArrayOfDiagramResource**, as specified in section [3.1.4.2.3.4](#), that specifies all the resources used by the **Xaml** field. There MUST be exactly one instance of this element, and it MUST NOT be null.

ShapeInfo: A string that specifies information about the shapes, as specified in section [1.3.3](#), in the drawing page, as specified in section [1.3.2](#). There MUST be exactly one instance of this element, and it MUST NOT be null.

The syntax of this element is specified in [\[MS-VGSFF\]](#) section 2.4.8. The syntax of this element is also specified by additional attributes of the **CT_Page** element, as specified in [\[MS-VGSFF\]](#) section 2.4.8.2.8, that are listed in the following table.

Attribute of CT_Page element	Description
Conflict	An integer that specifies the status of retrieving data from data providers for the drawing page. The value MUST be equal to zero.
DocumentID	A string that specifies a GUID that identifies the file in a document repository that contains the VGSFF web drawing being requested.
DocumentTimeStamp	A dateTime value that specifies the date and time, in Coordinated Universal Time (UTC) format, when the file in a document repository that contains the VGSFF web drawing was last modified in the document repository.
RefreshWarning	An optional integer that specifies whether the drawing page was refreshed by the protocol server. If this attribute is present, a value of one indicates that data connections (2) exist for the drawing page but that the drawing page was not refreshed from the data providers associated with these data connections (2).

Xaml: A string that specifies the vector drawing page, as specified in section [1.3.2](#), in XAML format, as specified in [\[MS-SLXV\]](#). There MUST be exactly one instance of this element, and it MUST NOT be null.

3.1.4.2.3.4 ArrayOfDiagramResource

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **ArrayOfDiagramResource** complex type contains an array of **DiagramResource** complex types (section [3.1.4.2.3.5](#)).

```
<xs:complexType name="ArrayOfDiagramResource" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="DiagramResource" nillable="true"
      type="tns:DiagramResource"/>
  </xs:sequence>
</xs:complexType>
```

DiagramResource: A **DiagramResource** element that specifies the **fonts** and images required by a [drawing page](#). It MUST NOT be null.

3.1.4.2.3.5 DiagramResource

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **DiagramResource** complex type contains information that specifies either a font or an image required by a [drawing page](#).

```
<xs:complexType name="DiagramResource" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="ByteData" nillable="true" type="xs:base64Binary"/>
    <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

ByteData: A base64Binary that specifies the resource in the format of a stream of bytes. There MUST be exactly one instance of this element.

Name: A string that specifies the name for the resource as referenced by the **Xaml** field of the **RenderedPage** complex type (section [3.1.4.2.3.3](#)). It MUST refer to either a font as specified in [\[MS-VGSFF\]](#) section 2.1.6.2 or an image as specified in [\[MS-VGSFF\]](#) section 2.1.6.3. There MUST be exactly one instance of this element, and it MUST NOT be null.

3.1.4.2.4 Simple Types

None.

3.1.4.2.5 Attributes

None.

3.1.4.2.6 Groups

None.

3.1.4.2.7 Attribute Groups

None.

3.1.4.3 GetRasterPage

The **GetRasterPage** operation retrieves information about a [drawing page](#) of a **VSDX web drawing** rendered in raster format. <10>

```
<wsdl:operation name="GetRasterPage" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPage"
  message="tns2:IVisioGraphicsService_GetRasterPage_InputMessage"
  xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPageResponse"
  message="tns2:IVisioGraphicsService_GetRasterPage_OutputMessage"
  xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:fault
  wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPageVisioGraphicsServiceFaultF
  ault" name="VisioGraphicsServiceFaultFault"
  message="tns2:IVisioGraphicsService_GetRasterPage_VisioGraphicsServiceFaultFault_FaultMessage"
  xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IVisioGraphicsService_GetRasterPage_InputMessage** request (section [3.1.4.3.1.1](#)) WSDL message, and the protocol server MUST respond with an **IVisioGraphicsService_GetRasterPage_OutputMessage** response (section [3.1.4.3.1.2](#)) WSDL message as follows:

1. If the protocol server determines that an error occurred during the protocol server operation, the protocol server MUST respond with a **VisioGraphicsServiceFault** complex type (section [2.2.4.5](#)) SOAP fault having the **Error** field set to the corresponding **FaultCode** value (section [2.2.5.1](#)).
2. If the protocol server determines that the drawing page requested by the protocol client has not changed, based on the information set in the **EntityTag** element of the **RasterPageRequest** complex type (section [3.1.4.3.3.1](#)), as described in the Messages and Event Sequence Rules in section [3.1.4](#), the protocol server MUST return a **RasterPageResponse** (section [3.1.4.3.3.2](#)) with the response code "NotModified".
3. If the requested drawing page contains data providers to be implemented by the protocol client, the protocol server MUST return a **RasterPageResponse** with the response code "DataSources" and the following sequence of messages MUST be followed:
 1. The protocol server MUST respond with a **IVisioGraphicsService_GetRasterPage_OutputMessage** response (section [3.1.4.3.1.2](#)) WSDL message. The data providers MUST be specified in the **DataSources** element of the **RasterPageResponse** complex type (section [3.1.4.3.3.2](#)) contained within this response.
 2. The protocol client MUST retrieve the data from the data providers and MUST send another **IVisioGraphicsService_GetRasterPage_InputMessage** request (section [3.1.4.3.1.1](#)) WSDL message. This request MUST be identical to the original request except that the **DataSources** element of the **RasterPageRequest** complex type (section [3.1.4.3.3.1](#)) MUST NOT be null.

3. The protocol server MUST respond with a **IVisioGraphicsService_GetRasterPage_OutputMessage** response WSDL message. If no errors occurred, the response MUST contain a **RasterPageResponse** with the response code "OK".
4. Otherwise, the protocol server MUST return a **RasterPageResponse** with the response code **OK**.

3.1.4.3.1 Messages

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

Message	Description
IVisioGraphicsService_GetRasterPage_InputMessage	The request WSDL message for the GetRasterPage WSDL operation.
IVisioGraphicsService_GetRasterPage_OutputMessage	The response WSDL message for the GetRasterPage WSDL operation.
IVisioGraphicsService_GetRasterPage_VisioGraphicsServiceFaultFault_FaultMessage	The fault WSDL message for the GetRasterPage WSDL operation.

3.1.4.3.1.1 IVisioGraphicsService_GetRasterPage_InputMessage

The request WSDL message for the **GetRasterPage** WSDL operation.

The SOAP action value is:

```
http://tempuri.org/IVisioGraphicsService/GetRasterPage
```

The SOAP body contains the **GetRasterPage** element.

3.1.4.3.1.2 IVisioGraphicsService_GetRasterPage_OutputMessage

The response WSDL message for the **GetRasterPage** WSDL operation.

The SOAP body contains the **GetRasterPageResponse** element.

3.1.4.3.1.3 IVisioGraphicsService_GetRasterPage_VisioGraphicsServiceFaultFault_FaultMessage

The fault WSDL message for the **GetRasterPage** WSDL operation.

The SOAP body contains the **VisioGraphicsServiceFault** element.

3.1.4.3.2 Elements

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

Element	Description
GetRasterPage	The input data for the GetRasterPage WSDL operation.
GetRasterPageResponse	The result data for the GetRasterPage WSDL operation.

3.1.4.3.2.1 GetRasterPage

The input data for the **GetRasterPage** WSDL operation.

```
<xs:element name="GetRasterPage" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element
        xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer" minOccurs="0" name="rasterPageRequestContract" nillable="true"
        type="tns:RasterPageRequest"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

rasterPageRequestContract: A **RasterPageRequest** complex type (section [3.1.4.3.3.1](#)) that specifies the input data. There MUST be exactly one instance of this element, and it MUST NOT be null.

3.1.4.3.2.2 GetRasterPageResponse

The result data for the **GetRasterPage** WSDL operation.

```
<xs:element name="GetRasterPageResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element
        xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer" minOccurs="0" name="GetRasterPageResult" nillable="true" type="tns:RasterPageResponse"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetRasterPageResult: A **RasterPageResponse** complex type (section [3.1.4.3.3.2](#)), that specifies the result data. There MUST be exactly one instance of this element, and it MUST NOT be null.

3.1.4.3.3 Complex Types

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

Complex type	Description
RasterPageRequest	The set of elements that specifies the input data for a GetRasterPage (section 3.1.4.3) WSDL operation.
RasterPageTag	The set of elements that specifies opaque information assigned by the protocol server to a drawing page of a VSDX web drawing when it is requested for the first time. It is used on subsequent requests to determine whether the same drawing page has changed since the initial request.
RasterPageResponse	A set of elements that specifies the result data for a GetRasterPage WSDL operation.
RasterPageInfo	The information about a drawing page of a VSDX web drawing.

3.1.4.3.3.1 RasterPageRequest

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer

The **RasterPageRequest** complex type contains the set of elements that specifies the input data for a **GetRasterPage** (section [3.1.4.3.2.1](#)) WSDL operation.

```
<xs:complexType name="RasterPageRequest" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="DataModeName" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="DataSources" nillable="true"
type="tns:ArrayOfAddonDataSource"/>
    <xs:element minOccurs="0" name="EntityTag" nillable="true" type="tns:RasterPageTag"/>
    <xs:element minOccurs="0" name="FileUrl" nillable="true" type="xs:anyURI"/>
    <xs:element minOccurs="0" name="PageId" type="xs:long"/>
  </xs:sequence>
</xs:complexType>
```

DataModeName: A string that specifies whether the protocol server refreshes the requested [drawing page](#). There MUST be exactly one instance of this element, and it MUST be one of the following values.

Value	Description
Static	The drawing page MUST NOT be updated, and contains only data embedded in the corresponding web drawing .
Dynamic	The drawing page MUST be updated with data retrieved from data providers, as specified in [MS-VSDX] section 2.2.10, and from the system, as specified in [MS-VSDX] section 2.2.11.
DynamicSafe	The protocol server will decide whether the refresh queries, specified by the Command attribute described in [MS-VSDX] section 2.3.4.2.31, are safe to execute in order to retrieve data from data providers, as specified in [MS-VSDX] section 2.2.10. If all the queries are considered safe, the drawing page MUST be refreshed as specified by Dynamic . Otherwise, the drawing page MUST NOT be refreshed, as specified by Static , and the Page element of the shape information XML, specified in section 3.1.1 , MUST have the RefreshWarning attribute set to one, as specified in section 2.2.9.1.1 . This allows the protocol client to implement a mechanism that requests permission from the user before executing queries that are not considered safe.

DataSources: An **ArrayOfAddonDataSource** complex type (section [2.2.4.2](#)) that specifies the data providers implemented by a protocol client and used in the requested drawing page. There MUST be exactly one instance of this element. It MUST have zero child elements if there are no data providers. If **DataSources** has child elements, the **Data** field of each **AddonDataSource** complex type (section [2.2.4.1](#)) MUST NOT be null.

EntityTag: A **RasterPageTag** complex type (section [3.1.4.3.3.3](#)) that specifies opaque information about the requested drawing page, and used by the protocol server to determine if the drawing page has changed since it was initially requested. There MUST be exactly one instance of this element. A null or empty value indicates an unconditional message as specified in section [3.1.1](#).

FileUrl: An **anyURI** that specifies the URL of the file in a document repository that contains the web drawing being requested. There MUST be exactly one instance of this element, and it MUST NOT be null.

PageId: A long that specifies the requested drawing page. If greater than or equal to zero it specifies the identifier of the drawing page. If less than zero it specifies the bitwise complement of the zero-based index of the drawing page. There MUST be exactly one instance of this element.

3.1.4.3.3.2 RasterPageResponse

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **RasterPageResponse** complex type contains the set of elements that specifies the result data for a **GetRasterPage** (section [3.1.4.3](#)) WSDL operation.

```
<xs:complexType name="RasterPageResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="DataSources" nillable="true"
      type="tns:ArrayOfAddonDataSource"/>
    <xs:element minOccurs="0" name="EntityTag" nillable="true" type="tns:RasterPageTag"/>
    <xs:element minOccurs="0" name="PageInfo" nillable="true" type="tns:RasterPageInfo"/>
    <xs:element minOccurs="0" name="ResponseCode" type="tns:RasterPageResponseCode"/>
  </xs:sequence>
</xs:complexType>
```

DataSources: An **ArrayOfAddonDataSource** complex type (section [2.2.4.2](#)) that specifies the data providers implemented by a protocol client and used in the requested [drawing page](#). There MUST be exactly one instance of this element. It MUST have zero child elements, if there are no data providers. If **DataSources** has child elements, the **Data** field of each contained **AddonDataSource** complex type (section [2.2.4.1](#)) MUST NOT be empty. If the value of **ResponseCode** is "DataSources", it MUST NOT be null or empty. Otherwise, the **DataSources** element MUST be null.

EntityTag: A **RasterPageTag** complex type (section [3.1.4.3.3.3](#)), that specifies opaque information about the requested drawing page. There MUST be exactly one instance of this element. If the value of **ResponseCode** is "OK" or "NotModified", it MUST NOT be null. If the value of **ResponseCode** is "NotModified", it MUST be equal to the value of the **EntityTag** element of the corresponding **RasterPageRequest** complex type (section [3.1.4.3.3.1](#)). Otherwise, the **EntityTag** element MUST be null.

PageInfo: A **RasterPageInfo** complex type (section [3.1.4.3.3.4](#)) that specifies information about the requested drawing page. There MUST be exactly one instance of this element. If the value of **ResponseCode** is "OK", it MUST NOT be null. Otherwise, the **PageInfo** element MUST be null.

ResponseCode: A **RasterPageResponseCode** value (section [3.1.4.3.4.1](#)) that specifies the type of the response. There MUST be exactly one instance of this element.

3.1.4.3.3.3 RasterPageTag

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **RasterPageTag** complex type contains the set of elements that specifies opaque information assigned by the protocol server to a [drawing page](#) of a **VSDX** [web drawing](#) when it is requested for the first time, and used on subsequent requests to determine whether the same drawing page has changed since the initial request. A **RasterPageTag** is empty if all its elements have default values.

```
<xs:complexType name="RasterPageTag" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="DataModeName" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="FileETag" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="OutputETag" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="SequenceId" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

DataModeName: A string, that specifies whether the drawing page was refreshed by the protocol server. There MUST be exactly one instance of this element, and it MUST be one of the following values.

Value	Description
Static	The drawing page was not updated, and contains only data embedded in the corresponding web drawing. This is the default value.
Dynamic	The drawing page was updated with data retrieved from data providers, as specified in [MS-VSDX] section 2.2.10, or from the system, as specified in [MS-VSDX] section 2.2.11.

FileETag: A string that specifies an opaque identifier for the version of the web drawing. There MUST be exactly one instance of this element. It MUST NOT be null if **OutputETag** or **SequenceId** is not null. The default value is null.

OutputETag: A string that specifies an opaque identifier for the entity that stores the drawing page in the protocol server's output cache, as described in the Messages and Event Sequence Rules in section [3.1.4](#),. There MUST be exactly one instance of this element. It MUST NOT be null if **FileETag** or **SequenceId** is not null. The default value is null.

SequenceId: A string that specifies an identifier for the sequence of subsequent messages required to retrieve the items of the drawing page, as specified in section [3.1](#). There MUST be exactly one instance of this element. It MUST NOT be null if **FileETag** or **OutputETag** is not null. The default value is null.

3.1.4.3.3.4 RasterPageInfo

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **RasterPageInfo** complex type contains information about a [drawing page](#) of a **VSDX** [web drawing](#) rendered in raster format.

```

<xs:complexType name="RasterPageInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="DefaultImage" type="xs:int"/>
    <xs:element xmlns:q1="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
      minOccurs="0" name="ZoomValues" nillable="true" type="q1:ArrayOfint"/>
  </xs:sequence>
</xs:complexType>

```

DefaultImage: An integer that specifies the zero based index of the item of a drawing page, as specified in section 3.1.1, that corresponds to the default image. There MUST be exactly one instance of this element. It MUST be greater than or equal to zero, and MUST be less than the length of the **ZoomValues** array.

ZoomValues: An **ArrayOfInt** that contains the zoom values for the items of a drawing page, as specified in section 3.1.1, which correspond to images. There MUST be exactly one instance of this element. The length of the array MUST be equal to 3. Each zoom value in the array MUST be an **integer** greater than or equal to 25, and less than or equal to 400.

3.1.4.3.4 Simple Types

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

Simple type	Description
RasterPageResponseCode	The possible types of response codes returned as an element of a GetRasterPage (section 3.1.4.3) WSDL operation.

3.1.4.3.4.1 RasterPageResponseCode

Namespace:

http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer

The **RasterPageResponseCode** simple type specifies the possible types of response codes returned as an element of a **GetRasterPage** (section 3.1.4.3) WSDL operation.

```

<xs:simpleType name="RasterPageResponseCode" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:restriction base="xs:string">
    <xs:enumeration value="NotModified"/>
    <xs:enumeration value="OK"/>
    <xs:enumeration value="DataSources"/>
  </xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for **RasterPageResponseCode**.

Value	Meaning
NotModified	Specifies that the requested drawing page has not changed since it was initially requested by the protocol client.
OK	Specifies that the response includes full information about the requested drawing page.
DataSources	Specifies that the response includes information about the data providers implemented by

Value	Meaning
	the protocol client and used by the requested drawing page.

3.1.4.3.5 Attributes

None.

3.1.4.3.6 Groups

None.

3.1.4.3.7 Attribute Groups

None.

3.1.4.4 GetRasterPageItem

The **GetRasterPageItem** operation retrieves an item, as specified in section 3.1.1, of a [drawing page](#) of a **VSDX web drawing** rendered in raster format. <11>

```
<wsdl:operation name="GetRasterPageItem" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPageItem"
    message="tns2:IVisioGraphicsService_GetRasterPageItem_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output
    wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPageItemResponse"
    message="tns2:IVisioGraphicsService_GetRasterPageItem_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:fault
    wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPageItemVisioGraphicsServiceFaultFault"
    name="VisioGraphicsServiceFaultFault"
    message="tns2:IVisioGraphicsService_GetRasterPageItem_VisioGraphicsServiceFaultFault_FaultMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>
```

The protocol client sends an **IVisioGraphicsService_GetRasterPageItem_InputMessage** request (section 3.1.4.4.1.1) WSDL message, and the protocol server MUST respond with an **IVisioGraphicsService_GetRasterPageItem_OutputMessage** response (section 3.1.4.4.1.2) WSDL message as follows:

1. If the protocol server determines that an error occurred during the protocol server operation, then the protocol server MUST respond with a **VisioGraphicsServiceFault** complex type (section 2.2.4.5) SOAP fault with the **Error** field set to the appropriate **FaultCode** value (section 2.2.5.1).
2. If the protocol server determines that the item specified by the protocol client has not changed based on the information set in the **EntityTag** element of the **RasterPageItemRequest** complex type (section 3.1.4.4.3.1), as specified in section 3.1.4, the protocol server MUST return a **RasterPageItemResponse** complex type (section 3.1.4.4.3.3) with the response code "NotModified".
3. Otherwise, the protocol server MUST return a **RasterPageItemResponse** with the response code "OK".

3.1.4.4.1 Messages

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

Message	Description
IVisioGraphicsService_GetRasterPageItem_InputMessage	The request WSDL message for the GetRasterPageItem WSDL operation.
IVisioGraphicsService_GetRasterPageItem_OutputMessage	The response WSDL message for the GetRasterPageItem WSDL operation.
IVisioGraphicsService_GetRasterPageItem_VisioGraphicsServiceFaultFault_FaultMessage	The fault WSDL message for the GetRasterPageItem WSDL operation.

3.1.4.4.1.1 IVisioGraphicsService_GetRasterPageItem_InputMessage

The request WSDL message for the **GetRasterPageItem** WSDL operation.

The SOAP action value is:

`http://tempuri.org/IVisioGraphicsService/GetRasterPageItem`

The SOAP body contains the **GetRasterPageItem** element.

3.1.4.4.1.2 IVisioGraphicsService_GetRasterPageItem_OutputMessage

The response WSDL message for the **GetRasterPageItem** WSDL operation.

The SOAP body contains the **GetRasterPageItemResponse** element.

3.1.4.4.1.3 IVisioGraphicsService_GetRasterPageItem_VisioGraphicsServiceFaultFault_FaultMessage

The fault WSDL message for the **GetRasterPageItem** WSDL operation.

The SOAP body contains the **VisioGraphicsServiceFault** element.

3.1.4.4.2 Elements

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

Element	Description
GetRasterPageItem	The input data for the GetRasterPageItem WSDL operation.

Element	Description
GetRasterPageItemResponse	The result data for the GetRasterPageItem WSDL operation.

3.1.4.4.2.1 GetRasterPageItem

The input data for the **GetRasterPageItem** WSDL operation.

```
<xs:element name="GetRasterPageItem" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element
        xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer"
        minOccurs="0" name="rasterPageItemRequestContract" nillable="true"
        type="tns:RasterPageItemRequest"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

rasterPageItemRequestContract: A **RasterPageItemRequest** complex type (section [3.1.4.4.3.1](#)), that specifies the input data. There MUST be exactly one instance of this element, and it MUST NOT be null.

3.1.4.4.2.2 GetRasterPageItemResponse

The result data for the **GetRasterPageItem** WSDL operation.

```
<xs:element name="GetRasterPageItemResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element
        xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer"
        minOccurs="0" name="GetRasterPageItemResult" nillable="true"
        type="tns:RasterPageItemResponse"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetRasterPageItemResult: A **RasterPageItemResponse** complex type (section [3.1.4.4.3.3](#)), that specifies the result data. There MUST be exactly one instance of this element, and it MUST NOT be null.

3.1.4.4.3 Complex Types

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

Complex type	Description
RasterPageItemRequest	The set of elements that specifies the input data for the GetRasterPageItem (section 3.1.4.4) WSDL operation.
RasterPageItemTag	The set of elements that specifies opaque information assigned by the protocol server to an item of a drawing page , as specified in section 3.1.1 , of a VSDX web drawing , when it is requested for the first time, and used on subsequent

Complex type	Description
	requests to determine whether the same item has changed since the initial request.
RasterPageItemResponse	The set of elements that specifies the result data for the GetRasterPageItem WSDL operation.

3.1.4.4.3.1 RasterPageItemRequest

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **RasterPageItemRequest** complex type contains a set of elements that specifies the input data for the **GetRasterPageItem** (section [3.1.4.4](#)) WSDL operation.

```
<xs:complexType name="RasterPageItemRequest" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="EntityTag" nillable="true" type="tns:RasterPageItemTag"/>
    <xs:element minOccurs="0" name="FileUrl" nillable="true" type="xs:anyURI"/>
    <xs:element minOccurs="0" name="ItemName" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="SequenceId" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

EntityTag: A **RasterPageItemTag** complex type (section [3.1.4.4.3.2](#)) that specifies opaque information about the requested item of the [drawing page](#), as specified in section [3.1.1](#), and used by the protocol server to determine if it has changed since it was initially requested. There MUST be exactly one instance of this element. A null or empty value indicates an unconditional message as specified in section [3.1.1](#).

FileUrl: An **anyURI** that specifies the URL of the file in a document repository that contains the [web drawing](#) being requested. There MUST be exactly one instance of this element, and it MUST NOT be null.

ItemName: A string that specifies the name of the requested item in a drawing page, as specified in section [3.1.1](#). There MUST be exactly one instance of this element, and it MUST be one of the following values.

Value	Description
Image0	Specifies the requested item is a small image, in Portable Network Graphics (PNG) format as specified in [RFC2083]
Image1	Specifies the requested item is a medium image, in Portable Network Graphics (PNG) format as specified in [RFC2083]
Image2	Specifies the requested item is a large image, in Portable Network Graphics (PNG) format as specified in [RFC2083]
ShapeInfo	Specifies the requested item is shape information in XML format, as specified in section 2.2.9.1
Comments	Specifies the requested item is comments in XML format, as specified in section 2.2.9.2

SequenceId: A string that specifies the message sequence identifier obtained in the **IVisioGraphicsService_GetRasterPage_InputMessage** request (section [3.1.4.3.1.1](#)) WSDL message that initiated the retrieval of a drawing page, as specified in section [3.1](#).

3.1.4.4.3.2 RasterPageItemTag

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **RasterPageItemTag** complex type contains the set of elements that specifies opaque information assigned by the protocol server to an item of a [drawing page](#), as specified in section [3.1.1](#), of a **VSDX web drawing**, when it is requested for the first time. The **RasterPageItemTag** is used on subsequent requests to determine whether the same item has changed since the initial request. A **RasterPageItemTag** is empty, if all its elements have default values.

```
<xs:complexType name="RasterPageItemTag" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="DataModeName" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="FileETag" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

DataModeName: A string that specifies whether the drawing page was refreshed by the protocol server. There MUST be exactly one instance of this element, and it MUST be one of the following values.

Value	Description
Static	The drawing page was not updated and contains only data embedded in the corresponding web drawing. This is the default value.
Dynamic	The drawing page was updated with data retrieved from data providers, as specified in [MS-VSDX] section 2.2.10, or from the system, as specified in [MS-VSDX] section 2.2.11.

FileETag: A string that specifies an opaque identifier for the version of the web drawing. There MUST be exactly one instance of this element. The default value is null.

3.1.4.4.3.3 RasterPageItemResponse

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **RasterPageItemResponse** contains the set of elements that specifies the response data for the **GetRasterPageItem** (section [3.1.4.4](#)) WSDL operation.

```
<xs:complexType name="RasterPageItemResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" name="EntityTag" nillable="true" type="tns:RasterPageItemTag"/>
    <xs:element minOccurs="0" name="ItemData" nillable="true" type="xs:base64Binary"/>
    <xs:element minOccurs="0" name="ResponseCode" type="tns:RasterPageItemResponseCode"/>
  </xs:sequence>
</xs:complexType>
```

EntityTag: A **RasterPageItemTag** complex type (section [3.1.4.4.3.2](#)) that specifies opaque information about the requested [drawing page](#) item, as specified in section [3.1.1](#). There MUST be exactly one instance of this element, and it MUST NOT be null. If the value of **ResponseCode** is "NotModified", it MUST be equal to the value of the **EntityTag** element of the corresponding **RasterPageItemRequest** complex type (section [3.1.4.4.3.1](#)).

ItemData: A base64Binary that specifies the drawing page item data. There MUST be exactly one instance of this element. The value of **ItemData** depends on the value of the **ItemName** element of the corresponding **RasterPageItemRequest**, according to the following table.

Value of ItemName in RasterPageItemRequest	Value of ItemData
Image0	A small image of the rendered drawing page, in Portable Network Graphics (PNG) format as specified in [RFC2083] .
Image1	A medium image of the rendered drawing page, in Portable Network Graphics (PNG) format as specified in [RFC2083] .
Image2	A large image of the rendered drawing page, in Portable Network Graphics (PNG) format, as specified in [RFC2083] .
ShapeInfo	The shape information of the drawing page, in XML format, as specified in section 2.2.9.1 , and compressed using the GZIP format, as specified in [RFC1952] .
Comments	The comments of the drawing page, in XML format, as specified in section 2.2.9.2 , and encoded using UTF-8, as specified in [RFC3629] .

ResponseCode: A **RasterPageItemResponseCode** value (section [3.1.4.4.4.1](#)), that specifies the type of the response. There MUST be exactly one instance of this element.

3.1.4.4.4 Simple Types

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

Simple type	Description
RasterPageItemResponseCode	The possible types of response codes returned as an element of a GetRasterPageItem (section 3.1.4.4) WSDL operation.

3.1.4.4.4.1 RasterPageItemResponseCode

Namespace:

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer>

The **RasterPageItemResponseCode** simple type specifies the possible types of response codes returned as an element of a **GetRasterPageItem** (section [3.1.4.4](#)) WSDL operation.

```
<xs:simpleType name="RasterPageItemResponseCode" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:restriction base="xs:string">
    <xs:enumeration value="NotModified"/>
  </xs:restriction>
</xs:simpleType>
```

```

    <xs:enumeration value="OK"/>
  </xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for **RasterPageItemResponseCode**.

Value	Meaning
NotModified	Specifies that the requested drawing page item, as specified in section 3.1.1 , has not changed since it was initially requested by the protocol client.
OK	Specifies that the response includes drawing page item data.

3.1.4.4.5 Attributes

None.

3.1.4.4.6 Groups

None.

3.1.4.4.7 Attribute Groups

None.

3.1.4.5 SaveComments

The **SaveComments** operation saves a [comment](#) to a **VSDX web drawing** in a document repository, in the form of a `CommentEntry_Type` element as specified [\[MS-VSDX\]](#) section 2.3.4.2.9. [<12>](#)

```

<wsdl:operation name="SaveComments" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input wsaw:Action="http://tempuri.org/IVisioGraphicsService/SaveComments"
    message="tns2:IVisioGraphicsService_SaveComments_InputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:output wsaw:Action="http://tempuri.org/IVisioGraphicsService/SaveCommentsResponse"
    message="tns2:IVisioGraphicsService_SaveComments_OutputMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
  <wsdl:fault
    wsaw:Action="http://tempuri.org/IVisioGraphicsService/SaveCommentsVisioGraphicsServiceFaultFa
    ult" name="VisioGraphicsServiceFaultFault"
    message="tns2:IVisioGraphicsService_SaveComments_VisioGraphicsServiceFaultFault_FaultMessage"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"/>
</wsdl:operation>

```

The protocol client sends a **IVisioGraphicsService_SaveComments_InputMessage** request (section [3.1.4.5.1.1](#)) WSDL message, and the protocol server MUST respond with a **IVisioGraphicsService_SaveComments_OutputMessage** response (section [3.1.4.5.1.2](#)) WSDL message as follows:

1. If the protocol server determines that an error occurred during the protocol server operation, then the protocol server MUST respond with a **VisioGraphicsServiceFault** complex type (section [2.2.4.5](#)) SOAP fault with the **Error** field set to the appropriate **FaultCode** value (section [2.2.5.1](#)).

2. Otherwise, the protocol server MUST return a **SaveCommentsResponse** element (section [3.1.4.5.2.2](#)).

3.1.4.5.1 Messages

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

Message	Description
IVisioGraphicsService_SaveComments_InputMessage	The request WSDL message for the SaveComments WSDL operation.
IVisioGraphicsService_SaveComments_OutputMessage	The response WSDL message for the SaveComments WSDL operation.
IVisioGraphicsService_SaveComments_VisioGraphicsServiceFaultFault_FaultMessage	The fault WSDL message for the SaveComments WSDL operation.

3.1.4.5.1.1 IVisioGraphicsService_SaveComments_InputMessage

The request WSDL message for the **SaveComments** WSDL operation.

The SOAP action value is:

```
http://tempuri.org/IVisioGraphicsService/SaveComments
```

The SOAP body contains the **SaveComments** element.

3.1.4.5.1.2 IVisioGraphicsService_SaveComments_OutputMessage

The response WSDL message for the **SaveComments** WSDL operation.

The SOAP body contains the **SaveCommentsResponse** element.

3.1.4.5.1.3

IVisioGraphicsService_SaveComments_VisioGraphicsServiceFaultFault_FaultMessage

The fault WSDL message for the **SaveComments** WSDL operation.

The SOAP body contains the **VisioGraphicsServiceFault** element.

3.1.4.5.2 Elements

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

Element	Description
SaveComments	The input data for the SaveComments WSDL operation.
SaveCommentsResponse	The result data for the SaveComments WSDL operation.

3.1.4.5.2.1 SaveComments

The input data for the **SaveComments** WSDL operation.

```
<xs:element name="SaveComments" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="fileUrl" nillable="true" type="xs:anyURI"/>
      <xs:element minOccurs="0" name="commentData" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

fileUrl: An **anyURI** that specifies the URL of the **VSDX web drawing** file in a document repository. There **MUST** be exactly one instance of this element, and it **MUST NOT** be null.

commentData: A string attribute that specifies data about a **comment**, in XML format as specified in section [2.2.9.3](#), for a **shape** in the **VSDX web drawing**. There **MUST** be exactly one instance of this element, and it **MUST NOT** be null.

3.1.4.5.2.2 SaveCommentsResponse

The result data for the **SaveComments** WSDL operation.

```
<xs:element name="SaveCommentsResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" name="SaveCommentsResult" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

SaveCommentsResult: A string that specifies the result of the **SaveComments** operation (section [3.1.4.5](#)), in XML format as specified in section [2.2.9.4](#). There **MUST** be exactly one instance of this element, and it **MUST NOT** be null.

3.1.4.5.3 Complex Types

None.

3.1.4.5.4 Simple Types

None.

3.1.4.5.5 Attributes

None.

3.1.4.5.6 Groups

None.

3.1.4.5.7 Attribute Groups

None.

3.1.5 Timer Events

The **Recalculation Timeout** timer event cancels the request for the drawing page, as specified in section [1.3.2](#), with a **VisioGraphicsServiceFault**, as specified in section [2.2.4.5](#). The **Error** field of **VisioGraphicsServiceFault** MUST be set to the "RecalcTimedOut" **FaultCode** value, as specified in section [2.2.5.1](#).

The **Message Sequence Timeout** timer event expires the entity cached on the protocol server that is used to match the sequence identifier of the message sequence initiated by a **IVisioGraphicsService_GetRasterPageItem_InputMessage** request (section [3.1.4.4.1.1](#)) WSDL message to retrieve a drawing page of a **VSDX** [web drawing](#) in raster format, as specified in section [3.1.<13>](#)

3.1.6 Other Local Events

None.

4 Protocol Examples

The following sections contain examples of this protocol's usage.

4.1 GetRasterDiagram

In this scenario, the protocol client calls the **GetRasterDiagram** method (section [3.1.4.1](#)) on a protocol server named "ExampleServer". The protocol server responds with a **GetRasterDiagramResponse** element (section [3.1.4.1.2.2](#)). In this case, the [drawing page](#) does not contain data providers to be implemented by the protocol client.

To request the raster formatted drawing page, the protocol client constructs the following WSDL message:

```
<s:Envelope>
  <s:Header>
    <a:Action
s:mustUnderstand="1">http://tempuri.org/IVisioGraphicsService/GetRasterDiagram</a:Action>
    <a:MessageID>urn:uuid:891be72c-aaaf-473a-a4f8-82c3da217414</a:MessageID>
    <a:ReplyTo>
      <a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
    </a:ReplyTo>
    <a:To s:mustUnderstand="1">https://ExampleServer:
32844/69e08a7b0e904d1a95de6dbf31c51529/VisioGraphicsService.svc/secure</a:To>
    <o:Security>.....</o:Security>
  </s:Header>
  <s:Body>
    <GetRasterDiagram>
      <rasterDiagramRequestContract>
        <b:DataSources i:nil="true"/>
        <b:DisableRefresh>true</b:DisableRefresh>
        <b:FileUri>http://ExampleServer/BasicDiagram.vdw</b:FileUri>
        <b:PageNumber>1</b:PageNumber>
        <b:TimeLastModified>2009-05-20T17:42:20</b:TimeLastModified>
        <b:Resolution>1</b:Resolution>
      </rasterDiagramRequestContract>
    </GetRasterDiagram>
  </s:Body>
</s:Envelope>
```

The protocol server responds with the following:

```
<s:Envelope xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action
s:mustUnderstand="1">http://tempuri.org/IVisioGraphicsService/GetRasterDiagramResponse</a:Act
ion>
  </s:Header>
  <s:Body>
    <GetRasterDiagramResponse xmlns="http://tempuri.org/">
      <GetRasterDiagramResult
xmlns:d4p1="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsSer
ver" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <d4p1:CacheAge>60</d4p1:CacheAge>
        <d4p1:DataSources i:nil="true"></d4p1:DataSources>
        <d4p1:AreaMaps>
```


IAAAQIEGgUEmEzsrQgQIECAAAECBAGqYAQEmMxPNQECBAGQIECAAAECjQICTCO2VgQIECBAGaAABAGQI ZAICTOanmgABAG
QIECBAGACBRgEBphFbKwIECBAGQIAAAQIEMgEBJvNTTYAAAQIECBAGQIBA04AA04itFQECBAGQIECAAAECmYAAk/mpJkC
AAECBAGQIECgUUAacTWigABAGQIECBAGACBTECAyfxUEyBAGaAABAGQIECDQKCDANGJrRYAAAQIECBAGQIBAjiDAZH6q
CRAGQIAAAQIECBBoFBBgGrG1IkCAAECBAGQIEAgExBgMj/VBAGQIECAAAECBAG0Cggwj dhaESBAGaAABAGQIECCQCgwm
Z9qAgQIECBAGaAABAGqABQSYRmytCBAGQIAAAQIECBDBASyze81AQIECBAGQIAAAQKNAgJMI7ZWBAGQIECAAAECBAh
kAgJM5qeaAAECBAGQIECAAI FGAQGmEVsrAgQIECBAGaAABAGqYAQEm81NNgAABAGQIECBAGECjgADTiK0VAQIECBAGQIAA
AQKZgACT+akmQIAAAQIECBAGQKBRQIBpxNaKAAECBAGQIECAAI FMQIDJ/FQTI ECAAAECBAGQINAo IMA0YmtFgAABAGQIECBA
gEAmIMBkfqoJECBAGaAABAGQIEGgUEGAasbUiQIAAAQIECBAGQcATEGAyP9UECBAGQIAAAQIECDQKCDN2FoRIECAAAECB
AgQIJAJCDCzn2oCBAGQIECAAAECBBoFBjHGbK0IECBAGaAABAGQIEMgEBJjMTzUBAGQIECBAGaAAB00CAkwjt1YECBAGQI
AAAQIECGQCAkzmp5oAAQIECBAGQIAAgUYBAaYRWysCBAGQIECAAAECBDIBASbzU02AAECBAGQIECAQKOAANOIrRUBAGQ
IECBAGaAABApMAAJp5qSZAgAABAGQIECBAoFFAgGnE1oAAQIECBAGQIAAgUxAgMn8VBMgQIAAAQIECBAG0CggwDRia0WA
AAECBAGQIECAQCYgWR+qgkQIECAAAECBAGqABQQYBqxtSJAgaAABAGQIECBAIBMQYDI/1QQIECBAGaAABAGQINAoIMI3YW
hEgQIAAAQIECBAGkAkIMJmfagIECBAGQIAAAQIEGgUEmEzsrQgQIECAAAECBAGqYAQEmMxPNQECBAGQIECAAAECjQICTC
O2VgQIECBAGaAABAGQI ZAICTOanmgABAGQIECBAGACBRgEBphFbKwIECBAGQIAAAQIEMgEBJvNTTYAAAQIECBAGQIBA04A
A04itFQECBAGQIECAAAECmYAAk/mpJkCAAECBAGQIECgUUAacTWigABAGQIECBAGACBTECAyfxUEyBAGaAABAGQIECDQ
KCDANGJrRYAAAQIECBAGQIBAjiDAZH6qCRAGQIAAAQIECBBoFBBgGrG1IkCAAECBAGQIEAgExBgMj/VBAGQIECAAAECB
Ag0Cggwj dhaESBAGaAABAGQIECCQCgwmZ9qAgQIECBAGaAABAGqABQSYRmytCBAGQIAAAQIECBDBASyze81AQIECBAGQI
AAAQKNAgJMI7ZWBAGQIECAAAECBAhKAgJM5qeaAAECBAGQIECAAI FGAQGmEVsrAgQIECBAGaAABAGqYAQEm81NNgAABAG
QIECBAGECjgADTiK0VAQIECBAGQIAAAQKZgACT+akmQIAAAQIECBAGQKBRQIBpxNaKAAECBAGQIECAAI FMQIDJ/FQTI ECA
AAECBAGQINAo IMA0YmtFgAABAGQIECBAGeAmIMBkfqoJECBAGaAABAGQIEGgUEGAasbUiQIAAAQIECBAGQcATEGAyP9UEC
BAGQIAAAQIECDQKCDN2FoRIECAAAECBAGQIJAJCDCzn2oCBAGQIECAAAECBBoFBjHGbK0IECBAGaAABAGQIEMgEBJjMTz
UBAGQIECBAGaAAB00CAkwjt1YECBAGQIAAAQIECGQCAkzmp5oAAQIECBAGQIAAgUYBAaYRWysCBAGQIECAAAECBDIBASb
zU02AAECBAGQIECAQKOAANOIrRUBAGQIECBAGaAABApMAAJp5qSZAgAABAGQIECBAoFFAgGnE1oAAQIECBAGQIAAgUxAg
Mn8VBMgQIAAAQIECBAG0CggwDRia0WAAECBAGQIECAQCYgWR+qgkQIECAAAECBAGqABQQYBqxtSJAgaAABAGQIECBAI
BMQYDI/1QQIECBAGaAABAGQINAoIMI3YWhEgQIAAAQIECBAGkAkIMJmfagIECBAGQIAAAQIEGgUEmEzsrQgQIECAAAECBA
gYAQEmMxPNQECBAGQIECAAAECjQICTCO2VgQIECBAGaAABAGQI ZAICTOanmgABAGQIECBAGACBRgEBphFbKwIECBAGQIA
AAQIEMgEBJvNTTYAAAQIECBAGQIBA04AA04itFQECBAGQIECAAAECmYAAk/mpJkCAAECBAGQIECgUUAacTWigABAGQIE
CBAGACBTECAyfxUEyBAGaAABAGQIECDQKCDANGJrRYAAAQIECBAGQIBAjiDAZH6qCRAGQIAAAQIECBBoFBBgGrG1IkCAA
ECBAGQIEAgExBgMj/VBAGQIECAAAECBAG0Cggwj dhaESBAGaAABAGQIECCQCgwmZ9qAgQIECBAGaAABAGqABQSYRmytCB
AgQIAAAQIECBDBASyze81AQIECBAGQIAAAQKNAgJMI7ZWBAGQIECAAAECBAhKAgJM5qeaAAECBAGQIECAAI FGAQGmEVsr
AgQIECBAGaAABAGqYAQEm81NNgAABAGQIECBAGECjgADTiK0VAQIECBAGQIAAAQKZgACT+akmQIAAAQIECBAGQKBRQIBp
xNaKAAECBAGQIECAAI FMQIDJ/FQTI ECAAAECBAGQINAo IMA0YmtFgAABAGQIECBAGeAmIMBkfqoJECBAGaAABAGQIEGgUE
GAasbUiQIAAAQIECBAGQcATEGAyP9UECBAGQIAAAQIECDQKCDN2FoRIECAAAECBAGQIJAJCDCzn2oCBAGQIECAAAECB
BoFBjHGbK0IECBAGaAABAGQIEMgEBJjMTzUBAGQIECBAGaAAB00CAkwjt1YECBAGQIAAAQIECGQCAkzmp5oAAQIECBAGQIA
AgUYBAaYRWysCBAGQIECAAAECBDIBASbzU02AAECBAGQIECAQKOAANOIrRUBAGQIECBAGaAABApMAAJp5qSZAgAABAGQIE
CBAoFFAgGnE1oAAQIECBAGQIAAgUxAgMn8VBMgQIAAAQIECBAG0CggwDRia0WAAECBAGQIECAQCYgWR+qgkQIECAAA
ECBAGqABQQYBqxtSJAgaAABAGQIECBAIBMQYDI/1QQIECBAGaAABAGQINAoIMI3YWhEgQIAAAQIECBAGkAkIMJmfagIECB
AgQIAAAQIEGgUEmEzsrQgQIECAAAECBAGqYAQEmMxPNQECBAGQIECAAAECjQICTCO2VgQIECBAGaAABAGQI ZAICTOanmgA
BAGQIECBAGACBRgEBphFbKwIECBAGQIAAAQIEMgEBJvNTTYAAAQIECBAGQIBA04AA04itFQECBAGQIECAAAECmYAAk/mp
JkCAAECBAGQIECgUUAacTWigABAGQIECBAGACBTECAyfxUEyBAGaAABAGQIECDQKCDANGJrRYAAAQIECBAGQIBAjiDAZ
H6qCRAGQIAAAQIECBBoFBBgGrG1IkCAAECBAGQIEAgExBgMj/VBAGQIECAAAECBAG0Cggwj dhaESBAGaAABAGQIECCQC
gwmZ9qAgQIECBAGaAABAGqABQSYRmytCBAGQIAAAQIECBDBASyze81AQIECBAGQIAAAQKNAgJMI7ZWBAGQIECAAAECBA
hKAgJM5qeaAAECBAGQIECAAI FGAQGmEVsrAgQIECBAGaAABAGqYAQEm81NNgAABAGQIECBAGECjgADTiK0VAQIECBAGQIA
AQKZgACT+akmQIAAAQIECBAGQKBRQIBpxNaKAAECBAGQIECAAI FMQIDJ/FQTI ECAAAECBAGQINAo IMA0YmtFgAABAGQIE
CBAGeAmIMBkfqoJECBAGaAABAGQIEGgUEGAasbUiQIAAAQIECBAGQcATEGAyP9UECBAGQIAAAQIECDQKCDN2FoRIECAAA
ECBAGQIJAJCDCzn2oCBAGQIECAAAECBBoFBjHGbK0IECBAGaAABAGQIEMgEBJjMTzUBAGQIECBAGaAAB00CAkwjt1YECBA
gQIAAAQIECGQCAkzmp5oAAQIECBAGQIAAgUYBAaYRWysCBAGQIECAAAECBDIBASbzU02AAECBAGQIECAQKOAANOIrRUB
AgQIECBAGaAABApMAAJp5qSZAgAABAGQIECBAoFFAgGnE1oAAQIECBAGQIAAgUxAgMn8VBMgQIAAAQIECBAG0CggwDRia
0WAAECBAGQIECAQCYgWR+qgkQIECAAAECBAGqABQQYBqxtSJAgaAABAGQIECBAIBMQYDI/1QQIECBAGaAABAGQINAoIMI
3YWhEgQIAAAQIECBAGkAkIMJmfagIECBAGQIAAAQIEGgUEmEzsrQgQIECAAAECBAGqYAQEmMxPNQECBAGQIECAAAECjQICT
CO2VgQIECBAGaAABAGQI ZAICTOanmgABAGQIECBAGACBRgEBphFbKwIECBAGQIAAAQIEMgEBJvNTTYAAAQIECBAGQIBA
o4AA04itFQECBAGQIECAAAECmYAAk/mpJkCAAECBAGQIECgUUAacTWigABAGQIECBAGACBTECAyfxUEyBAGaAABAGQIE
CDQKCDANGJrRYAAAQIECBAGQIBAjiDAZH6qCRAGQIAAAQIECBBoFBBgGrG1IkCAAECBAGQIEAgExBgMj/VBAGQIECAAA
ECBAG0Cggwj dhaESBAGaAABAGQIECCQCgwmZ9qAgQIECBAGaAABAGqABQSYRmytCBAGQIAAAQIECBDBASyze81AQIECBA
gQIAAAQKNAgJMI7ZWBAGQIECAAAECBAhKAgJM5qeaAAECBAGQIECAAI FGAQGmEVsrAgQIECBAGaAABAGqYAQEm81NNgAAB
AgQIECBAGECjgADTiK0VAQIECBAGQIAAAQKZgACT+akmQIAAAQIECBAGQKBRQIBpxNaKAAECBAGQIECAAI FMQIDJ/FQTI
ECAAAECBAGQINAo IMA0YmtFgAABAGQIECBAGeAmIMBkfqoJECBAGaAABAGQIEGgUEGAasbUiQIAAAQIECBAGQcATEGAyP
9UECBAGQIAAAQIECDQKCDN2FoRIECAAAECBAGQIJAJCDCzn2oCBAGQIECAAAECBBoFBjHGbK0IECBAGaAABAGQIEMgEBJ
jMTzUBAGQIECBAGaAAB00CAkwjt1YECBAGQIAAAQIECGQCAkzmp5oAAQIECBAGQIAAgUYBAaYRWysCBAGQIECAAAECBDIB
ASbzU02AAECBAGQIECAQKOAANOIrRUBAGQIECBAGaAABApMAAJp5qSZAgAABAGQIECBAoFFAgGnE1oAAQIECBAGQIAAg
UxAgMn8VBMgQIAAAQIECBAG0CggwDRia0WAAECBAGQIECAQCYgWR+qgkQIECAAAECBAGqABQQYBqxtSJAgaAABAGQIEC
BAIBMQYDI/1QQIECBAGaAABAGQINAoIMI3YWhEgQIAAAQIECBAGkAkIMJmfagIECBAGQIAAAQIEGgUEmEzsrQgQIECAAA
ECBAGqYAQEmMxPNQECBAGQIECAAAECjQICTCO2VgQIECBAGaAABAGQI ZAICTOanmgABAGQIECBAGACBRgEBphFbKwIECB
AGQIAAAQIEMgEBJvNTTYAAAQIECBAGQIBA04AA04itFQECBAGQIECAAAECmYAAk/mpJkCAAECBAGQIECgUUAacTWigABA
gQIECBAGACBTECAyfxUEyBAGaAABAGQIECDQKCDANGJrRYAAAQIECBAGQIBAjiDAZH6qCRAGQIAAAQIECBBoFBBgGrG1Ik
CAAECBAGQIEAgExBgMj/VBAGQIECAAAECBAG0Cggwj dhaESBAGaAABAGQIECCQCgwmZ9qAgQIECBAGaAABAGqABQSYRmy


```

        <ShapeInfo Name="_1" DisplayName="Sheet.1"
Layout="\{&quot;type&quot;:&quot;2D&quot;,&quot;bounds&quot;:\{&quot;x&quot;:191,&quot;y&quot;
;:191,&quot;width&quot;:287,&quot;height&quot;:287\}\}" />
    </Page>
    </d4pl:ShapeInfo>
</GetRasterDiagramResult>
</GetRasterDiagramResponse>
</s:Body>
</s:Envelope>

```

4.2 GetVectorDiagram

In this scenario, the protocol client calls the **GetVectorDiagram** method (section 3.1.4.2) on a protocol server named "ExampleServer". The protocol server responds with a **GetVectorDiagramResponse** element (section 3.1.4.2.2). In this case the [drawing page](#) does not contain data providers to be implemented by the protocol client.

To request the vector formatted drawing page, the protocol client constructs the following WSDL message:

```

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action s:mustUnderstand="1"
xmlns:a="http://www.w3.org/2005/08/addressing">http://tempuri.org/IVisioGraphicsService/GetVe
ctorDiagram</a:Action>
    <a:MessageID xmlns:a="http://www.w3.org/2005/08/addressing">urn:uuid:d6266798-5744-42ed-
9753-cc263fda28ae</a:MessageID>
    <a:ReplyTo xmlns:a="http://www.w3.org/2005/08/addressing">
    <a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
    </a:ReplyTo>
    <a:To s:mustUnderstand="1"
xmlns:a="http://www.w3.org/2005/08/addressing">https://ExampleServer:32844/69e08a7b0e904d1a95
de6dbf31c51529/VisioGraphicsService.svc/secure</a:To>
    <o:Security xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">.....</o:Security>
  </s:Header>
  <s:Body>
    <GetVectorDiagram xmlns="http://tempuri.org/">
      <vectorDiagramRequestContract
xmlns:b="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer
" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <b:DataSources i:nil="true"></b:DataSources>
        <b:DisableRefresh>true</b:DisableRefresh>
        <b:FileUri>http://ExampleServer/BasicDiagram.vdw</b:FileUri>
        <b:PageNumber>1</b:PageNumber>
        <b:TimeLastModified>2009-05-20T17:42:20</b:TimeLastModified>
      </vectorDiagramRequestContract>
    </GetVectorDiagram>
  </s:Body>
</s:Envelope>

```

The protocol server responds with the following:

```

<s:Envelope xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>

```

```

    <a:Action
s:mustUnderstand="1">http://tempuri.org/IVisioGraphicsService/GetVectorDiagramResponse</a:Act
ion>
  </s:Header>
  <s:Body>
    <GetVectorDiagramResponse xmlns="http://tempuri.org/">
      <GetVectorDiagramResult
xmlns:d4p1="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsSer
ver" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <d4p1:CacheAge>60</d4p1:CacheAge>
        <d4p1:DataSources i:nil="true"></d4p1:DataSources>
        <d4p1:OutputPage>
          <d4p1:Resources></d4p1:Resources>
          <d4p1:ShapeInfo>
            <Page Name="Page-1" Zoom="-1.000000" OffsetX="407.142857" OffsetY="528.857143"
DefaultUnitsText="50" DefaultUnitsAngle="81" DefaultUnitsDuration="44" DefaultUnitsPage="73"
DocumentID="1360559653">
              <Pages>
                <Page Name="Page-1" />
              </Pages>
              <ShapeInfo Name="_1" DisplayName="Sheet.1"
Layout="{&quot;type&quot;:&quot;2D&quot;,&quot;bounds&quot;:{&quot;x&quot;:191,&quot;y&quot;
:191,&quot;width&quot;:287,&quot;height&quot;:287}}"/>
            </d4p1:ShapeInfo>
            <d4p1:Xaml>
              <Canvas Height="1056" Name="Workspace" Width="816"
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation">
                <!-- Page -->
                <Canvas Width="816" Height="1056" Name="D">
                  <Canvas.Background>
                    <SolidColorBrush Color="#FFFFFFFF"/>
                  </Canvas.Background>
                  <Canvas
Tag="{&#34;type&#34;:&#34;2D&#34;,&#34;bounds&#34;:{&#34;x&#34;:192,&#34;y&#34;:192,&#34;wi
dth&#34;:285,&#34;height&#34;:285}}"/>
                    Name="_1" RenderTransform="1, 0, 0, 1, 192, -579">
                      <Rectangle Canvas.Left="-0.5" Canvas.Top="770.5" Height="286"
Name="_1_G0_P0" Width="286" StrokeThickness="1"
StrokeDashCap="Round" StrokeEndLineCap="Round"
StrokeLineJoin="Round" StrokeStartLineCap="Round">
                        <Rectangle.Stroke>
                          <SolidColorBrush Color="#FF000000"/>
                        </Rectangle.Stroke>
                        <Rectangle.Fill>
                          <SolidColorBrush Color="#FFFFFFFF"/>
                        </Rectangle.Fill>
                      </Rectangle>
                    </Canvas>
                  </Canvas>
                </Canvas>
              </d4p1:Xaml>
            </d4p1:OutputPage>
          </GetVectorDiagramResult>
        </GetVectorDiagramResponse>
      </s:Body>
    </s:Envelope>
  </s:Header>
</s:Body>
</s:Envelope>

```

4.3 GetVectorDiagram with AddonDataSource

In this scenario, the protocol client calls the **GetVectorDiagram** method (section [3.1.4.2](#)) on a protocol server named "ExampleServer". In this case the protocol client is requesting a [drawing page](#) in a [web drawing](#) that is linked to data with a data provider to be implemented by the protocol client.

Initially the protocol client constructs the following WSDL message:

```
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action s:mustUnderstand="1"
xmlns:a="http://www.w3.org/2005/08/addressing">http://tempuri.org/IVisioGraphicsService/GetVectorDiagram</a:Action>
    <a:MessageID xmlns:a="http://www.w3.org/2005/08/addressing">urn:uuid:cd490d30-778d-447c-91f3-c0b0ea0e5de4</a:MessageID>
    <a:ReplyTo xmlns:a="http://www.w3.org/2005/08/addressing">
      <a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
    </a:ReplyTo>
    <a:To s:mustUnderstand="1"
xmlns:a="http://www.w3.org/2005/08/addressing">https://ExampleServer:32844/69e08a7b0e904d1a95de6dbf31c51529/VisioGraphicsService.svc/secure</a:To>
    <o:Security xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-secext-1.0.xsd">.....</o:Security>
  </s:Header>
  <s:Body>
    <GetVectorDiagram xmlns="http://tempuri.org/">
      <vectorDiagramRequestContract
xmlns:b="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <b:DataSources i:nil="true"></b:DataSources>
        <b:DisableRefresh>>false</b:DisableRefresh>
        <b:FileUri>http://ExampleServer/Data_Connected.vdw</b:FileUri>
        <b:PageNumber>1</b:PageNumber>
        <b:TimeLastModified>2009-05-22T16:24:27</b:TimeLastModified>
      </vectorDiagramRequestContract>
    </GetVectorDiagram>
  </s:Body>
</s:Envelope>
```

The protocol server responds with the following **GetVectorDiagramResponse** element (section [3.1.4.2.2.2](#)):

```
<s:Envelope xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action
s:mustUnderstand="1">http://tempuri.org/IVisioGraphicsService/GetVectorDiagramResponse</a:Action>
  </s:Header>
  <s:Body>
    <GetVectorDiagramResponse xmlns="http://tempuri.org/">
      <GetVectorDiagramResult
xmlns:d4p1="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <d4p1:CacheAge>0</d4p1:CacheAge>
        <d4p1:DataSources>
          <d4p1:AddonDataSource>
```

```

        <d4p1:ConnectionString>datamodule=Microsoft.Office.Visio.Server.EcsDataHandler,
Microsoft.Office.Visio.Server, Version=14.0.0.0, Culture=neutral,
PublicKeyToken=71e9bce11e9429c;provider=Microsoft.ACE.OLEDB.12.0;user id=Admin;data
source=http://ExampleServer/sampledatab.xlsx;mode=Read;extended
properties='HDR=YES;imex=1;maxscanrows=0;excel 12.0;';jet oledb:system database='';jet
oledb:registry path='';jet oledb:engine type=37;jet oledb:database locking mode=0;jet
oledb:global partial bulk ops=2;jet oledb:global bulk transactions=1;jet oledb:new database
password='';jet oledb:create system database=False;jet oledb:encrypt database=False;jet
oledb:don't copy locale on compact=False;jet oledb:compact without replica repair=False;jet
oledb:sfp=False;jet oledb:support complex data=False</d4p1:ConnectionString>
    <d4p1:Data>
        <xs:schema id="NewDataSet" xmlns="" xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
            <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
                <xs:complexType>
                    <xs:choice minOccurs="0" maxOccurs="unbounded">
                        <xs:element name="Table1" msdata:Locale="">
                            <xs:complexType>
                                <xs:sequence>
                                    <xs:element name="ID" type="xs:string"
minOccurs="0"></xs:element>
                                    <xs:element name="FN" type="xs:string"
minOccurs="0"></xs:element>
                                    <xs:element name="LN" type="xs:string"
minOccurs="0"></xs:element>
                                </xs:sequence>
                            </xs:complexType>
                        </xs:element>
                    </xs:choice>
                </xs:complexType>
            </xs:element>
            <diffgr:diffgram xmlns:msdata="urn:schemas-microsoft-com:xml-msdata"
xmlns:diffgr="urn:schemas-microsoft-com:xml-diffgram-v1"></diffgr:diffgram>
        </d4p1:Data>
        <d4p1:Id>1</d4p1:Id>
        <d4p1:QueryString>SheetName=Sheet1; RangeName=A1:C5;</d4p1:QueryString>
    </d4p1:AddonDataSource>
</d4p1:DataSources>
<d4p1:OutputPage i:nil="true"></d4p1:OutputPage>
</GetVectorDiagramResult>
</GetVectorDiagramResponse>
</s:Body>
</s:Envelope>

```

To request the vector formatted drawing page with an **AddonDataSource** complex type (section [2.2.4.1](#)), the protocol client now constructs the following WSDL message:

```

<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action s:mustUnderstand="1"
xmlns:a="http://www.w3.org/2005/08/addressing">http://tempuri.org/IVisioGraphicsService/GetVe
ctorDiagram</a:Action>
    <a:MessageID xmlns:a="http://www.w3.org/2005/08/addressing">urn:uuid:c22e255b-6923-496d-
a3f5-a2f3c45f4bbd</a:MessageID>
    <a:ReplyTo xmlns:a="http://www.w3.org/2005/08/addressing">
      <a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
    </a:ReplyTo>
  </s:Header>
  <s:Body>
    <GetVectorDiagramResult>
      <GetVectorDiagramResponse>
        <AddonDataSource>
          <DataSources>
            <Data>
              <Id>1</Id>
              <QueryString>SheetName=Sheet1; RangeName=A1:C5;</QueryString>
            </Data>
          </DataSources>
          <OutputPage i:nil="true"></OutputPage>
        </GetVectorDiagramResponse>
      </GetVectorDiagramResult>
    </s:Body>
  </s:Envelope>

```



```

    <a:To s:mustUnderstand="1"
xmlns:a="http://www.w3.org/2005/08/addressing">https://ExampleServer:32844/69e08a7b0e904d1a95
de6dbf31c51529/VisioGraphicsService.svc/secure</a:To>
    <o:Security xmlns:o="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
secext-1.0.xsd">.....</o:Security>
</s:Header>
<s:Body>
    <GetVectorDiagram xmlns="http://tempuri.org/">
    <vectorDiagramRequest
xmlns:b="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer
" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
    <b:DataSources>
    <b:AddonDataSource>
    <b:ConnectionString>datamodule=Microsoft.Office.Visio.Server.EcsDataHandler,
Microsoft.Office.Visio.Server, Version=14.0.0.0, Culture=neutral,
PublicKeyToken=71e9bce111e9429c;provider=Microsoft.ACE.OLEDB.12.0;user id=Admin;data
source=http://ExampleServer/sampledata.xlsx;mode=Read;extended
properties='HDR=YES;imex=1;maxscanrows=0;excel 12.0;';jet oledb:system database='';jet
oledb:registry path='';jet oledb:engine type=37;jet oledb:database locking mode=0;jet
oledb:global partial bulk ops=2;jet oledb:global bulk transactions=1;jet oledb:new database
password='';jet oledb:create system database=False;jet oledb:encrypt database=False;jet
oledb:don't copy locale on compact=False;jet oledb:compact without replica repair=False;jet
oledb:sfp=False;jet oledb:support complex data=False</b:ConnectionString>
    <b>Data>
    <xs:schema id="NewDataSet" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns=""
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
    <xs:element name="NewDataSet" msdata:IsDataSet="true" msdata:Locale="">
    <xs:complexType>
    <xs:choice minOccurs="0" maxOccurs="unbounded">
    <xs:element name="Table1">
    <xs:complexType>
    <xs:sequence>
    <xs:element name="ID" type="xs:double"
minOccurs="0"></xs:element>
    <xs:element name="FN" type="xs:string"
minOccurs="0"></xs:element>
    <xs:element name="LN" type="xs:string"
minOccurs="0"></xs:element>
    </xs:sequence>
    </xs:complexType>
    </xs:element>
    </xs:choice>
    </xs:complexType>
    </xs:element>
    </xs:schema>
    <diffgr:diffgram xmlns:diffgr="urn:schemas-microsoft-com:xml-diffgram-v1"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
    <NewDataSet xmlns="">
    <Table1 diffgr:id="Table11" msdata:rowOrder="0"
diffgr:hasChanges="inserted">
    <ID>1</ID>
    <FN>Happy</FN>
    <LN>kitty</LN>
    </Table1>
    <Table1 diffgr:id="Table12" msdata:rowOrder="1"
diffgr:hasChanges="inserted">
    <ID>2</ID>
    <FN>Silly</FN>
    <LN>dog</LN>
    </Table1>

```

```

        <Table1 diffgr:id="Table13" msdata:rowOrder="2"
diffgr:hasChanges="inserted">
        <ID>3</ID>
        <FN>Order</FN>
        <LN>Id</LN>
        </Table1>
        <Table1 diffgr:id="Table14" msdata:rowOrder="3"
diffgr:hasChanges="inserted">
        <ID>4</ID>
        <FN>green</FN>
        <LN>turtle</LN>
        </Table1>
        </NewDataSet>
    </diffgr:diffgram>
</b>Data>
<b:Id>1</b:Id>
<b:QueryString>SheetName=Sheet1; RangeName=A1:C5;</b:QueryString>
</b:AddonDataSource>
</b>DataSources>
<b:DisableRefresh>>false</b:DisableRefresh>
<b:FileUri>http://ExampleServer/Data_Connected.vdw</b:FileUri>
<b:PageNumber>1</b:PageNumber>
<b:TimeLastModified>2009-05-22T16:24:27</b:TimeLastModified>
</vectorDiagramRequest>
</GetVectorDiagram>
</s:Body>
</s:Envelope>

```

Finally, the protocol server responds with the following **GetVectorDiagramResponse** element (section [3.1.4.2.2.2](#)):

```

<s:Envelope xmlns:a="http://www.w3.org/2005/08/addressing"
xmlns:s="http://www.w3.org/2003/05/soap-envelope">
  <s:Header>
    <a:Action
s:mustUnderstand="1">http://tempuri.org/IVisioGraphicsService/GetVectorDiagramResponse</a:Action>
  </s:Header>
  <s:Body>
    <GetVectorDiagramResponse xmlns="http://tempuri.org/">
      <GetVectorDiagramResult
xmlns:d4pl="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <d4pl:CacheAge>5</d4pl:CacheAge>
        <d4pl>DataSources i:nil="true"></d4pl>DataSources>

        <d4pl:OutputPage>
          <d4pl:Resources></d4pl:Resources>
          <d4pl:ShapeInfo>
            <Page Name="Page-1" Zoom="1.192093" OffsetX="346.714286" OffsetY="769.714286"
DefaultUnitsText="50" DefaultUnitsAngle="81" DefaultUnitsDuration="44" DefaultUnitsPage="73"
Conflict="0" DocumentID="-1092095043">
              <Pages>
                <Page Name="Page-1" />
              </Pages>
              <ShapeInfo Name="_1" DisplayName="Tagged process"
Layout="{&quot;type&quot;:&quot;2D&quot;,&quot;bounds&quot;:{&quot;x&quot;:132,&quot;y&quot;:276,&quot;width&quot;:222,&quot;height&quot;:75}}">
                <ShapeDataItems>

```

```

        <ShapeData Name="ID" FormattedValue="1" Value="1" Format="{0:0.####} {1}"
Type="2" LangID="1033" UnitLabel="0" Unit="32" BindingID="1" />
        <ShapeData Name="FN" FormattedValue="Happy" Value="Happy" Format="{0}"
Type="0" LangID="1033" UnitLabel="0" Unit="105" BindingID="1" />
        <ShapeData Name="LN" FormattedValue="kitty" Value="kitty" Format="{0}"
Type="0" LangID="1033" UnitLabel="0" Unit="105" BindingID="1" />
    </ShapeDataItems>
</ShapeInfo>
    <ShapeInfo Name="_9" DisplayName="Tagged process.9"
Layout="\{\&quot;type&quot;;:&quot;2D&quot;;,&quot;bounds&quot;:\{\&quot;x&quot;;:96,&quot;y&quot;;
:456,&quot;width&quot;;:222,&quot;height&quot;;:75\}\}">
        <ShapeDataItems>
            <ShapeData Name="ID" FormattedValue="2" Value="2" Format="{0:0.####} {1}"
Type="2" LangID="1033" UnitLabel="0" Unit="32" BindingID="2" />
            <ShapeData Name="FN" FormattedValue="Silly" Value="Silly" Format="{0}"
Type="0" LangID="1033" UnitLabel="0" Unit="105" BindingID="2" />
            <ShapeData Name="LN" FormattedValue="dog" Value="dog" Format="{0}" Type="0"
LangID="1033" UnitLabel="0" Unit="105" BindingID="2" />
        </ShapeDataItems>
    </ShapeInfo>
    <ShapeInfo Name="_17" DisplayName="Tagged process.17"
Layout="\{\&quot;type&quot;;:&quot;2D&quot;;,&quot;bounds&quot;:\{\&quot;x&quot;;:120,&quot;y&quot;;
:648,&quot;width&quot;;:222,&quot;height&quot;;:75\}\}">
        <ShapeDataItems>
            <ShapeData Name="ID" FormattedValue="3" Value="3" Format="{0:0.####} {1}"
Type="2" LangID="1033" UnitLabel="0" Unit="32" BindingID="3" />
            <ShapeData Name="FN" FormattedValue="Order " Value="Order" Format="{0}"
Type="0" LangID="1033" UnitLabel="0" Unit="105" BindingID="3" />
            <ShapeData Name="LN" FormattedValue="Id" Value="Id" Format="{0}" Type="0"
LangID="1033" UnitLabel="0" Unit="105" BindingID="3" />
        </ShapeDataItems>
    </ShapeInfo>
    <ShapeInfo Name="_25" DisplayName="Tagged process.25"
Layout="\{\&quot;type&quot;;:&quot;2D&quot;;,&quot;bounds&quot;:\{\&quot;x&quot;;:480,&quot;y&quot;;
:492,&quot;width&quot;;:222,&quot;height&quot;;:75\}\}">
        <ShapeDataItems>
            <ShapeData Name="ID" FormattedValue="4" Value="4" Format="{0:0.####} {1}"
Type="2" LangID="1033" UnitLabel="0" Unit="32" BindingID="4" />
            <ShapeData Name="FN" FormattedValue="green" Value="green" Format="{0}"
Type="0" LangID="1033" UnitLabel="0" Unit="105" BindingID="4" />
            <ShapeData Name="LN" FormattedValue="turtle" Value="turtle" Format="{0}"
Type="0" LangID="1033" UnitLabel="0" Unit="105" BindingID="4" />
        </ShapeDataItems>
    </ShapeInfo>
</Page>
</d4pl:ShapeInfo>
<d4pl:Xaml>
    <Canvas Height="1056" Name="Workspace" Width="816"
xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation">
        <!-- Page -->
        <Canvas Width="816" Height="1056" Name="D">
            <Canvas.Background>
                <SolidColorBrush Color="#FFFFFFFF" />
            </Canvas.Background>
            <Canvas
Tag="\{\&quot;type&quot;;:&quot;2D&quot;;,&quot;bounds&quot;:\{\&quot;x&quot;;:132,&quot;y&quot;;:2
76,&quot;width&quot;;:96,&quot;height&quot;;:72\}\}" Name="_1" RenderTransform="1, 0, 0, 1,
132, -708">
                <Canvas Name="_2">
                    <Canvas Tag="Shdw" RenderTransform="1, 0, 0, 1, 1.92, 1.92">

```

```

        <Path Name="S_2_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M72 1056
L96 1056 L96 1032 L96 984 L0 984 L0 1056 L72 1056 Z">
        <Path.Stroke>
        <SolidColorBrush Color="#7FDEDEDE" />
        </Path.Stroke>
        <Path.Fill>
        <SolidColorBrush Color="#7FDEDEDE" />
        </Path.Fill>
    </Path>
</Canvas>
    <Path Name="_2_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M72 1056
L96 1056 L96 1032 L96 984 L0 984 L0 1056 L72 1056 Z">
    <Path.Stroke>
    <SolidColorBrush Color="#00404040" />
    </Path.Stroke>
    <Path.Fill>
    <LinearGradientBrush StartPoint="0.5,0" EndPoint="0.5,1">
    <LinearGradientBrush.GradientStops>
    <GradientStop Color="#00FF00FF" />
    <GradientStop Color="#00FF00FF" Offset="1" />
    </LinearGradientBrush.GradientStops>
    </LinearGradientBrush>
    </Path.Fill>
    </Path>
</Canvas>
<Canvas Name="_3" RenderTransform="1, 0, 0, 1, 72, 0">
    <Path Name="_3_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M0 1056
L24 1056 L24 1032 L0 1056 Z">
    <Path.Stroke>
    <SolidColorBrush Color="#FF404040" />
    </Path.Stroke>
    <Path.Fill>
    <SolidColorBrush Color="#FF000000" />
    </Path.Fill>
    </Path>
</Canvas>
<Canvas Name="_5">
    <Canvas.RenderTransform>
    <MatrixTransform Matrix="1,0,0,1,102,-16.65076" />
    </Canvas.RenderTransform>
    <Path Name="_5_G0_P0" Data="M1.92 1054.08 L118.08 1054.08 L118.08
1038.57080078125 L1.92 1038.57080078125 L1.92 1054.08 Z"></Path>
    <Path Name="_5_G1_P1" Data="M1.92 1054.08 L118.08 1054.08"></Path>
    <Path Name="_5_G0_P0_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08 L118.08 1038.57080078125 L1.92 1038.57080078125 L1.92 1054.08 Z">
    <Path.Stroke>
    <SolidColorBrush Opacity="0" Color="#FF404040" />
    </Path.Stroke>
    </Path>
    <Path Name="_5_G1_P1_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08">
    <Path.Stroke>
    <SolidColorBrush Color="#FF404040" />
    </Path.Stroke>
    </Path>

```

```

        <TextBlock
Tag="Left;Top;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888889"
Height="0.201553942620754" Visibility="Collapsed" Width="0.75625" RenderTransform="96, 0, 0,
96, 1.92, 1038.608582" TextWrapping="Wrap">
    <Run Name="txt_5_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">LN </Run>
</TextBlock>
</Canvas>
<Canvas>
    <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="3.2534" OriginY="1049.5" UnicodeString="LN">
        <Glyphs.Fill>
            <SolidColorBrush Color="#FF000000" />
        </Glyphs.Fill>
    </Glyphs>
</Canvas>
<Canvas Name="_6">
    <TextBlock
Tag="Right;Center;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888888
9" Height="0.201553942620754" Visibility="Collapsed" Width="1.0386175337222"
RenderTransform="96, 0, 0, 96, 18.372717, 1038.608582" TextWrapping="Wrap">
    <Run Name="txt_6_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">not kitty </Run>
</TextBlock>
</Canvas>
<Canvas>
    <Canvas.RenderTransform>
        <MatrixTransform Matrix="96,0,0,96,18.3744,1038.571" />
    </Canvas.RenderTransform>
    <Glyphs UnicodeString="Kitty" OriginX="0.825711078964174"
OriginY="0.111110001802444" FontUri="CommonFonts/4C1AABE5-BA71-4F2C-8263-66731F995C4B.odttf"
FontRenderingEmSize="0.111110001802444" Fill="#FF000000" Visibility="Visible" />
</Canvas>
</Canvas>
</Canvas>
<Canvas Name="_7">
    <Canvas.RenderTransform>
        <MatrixTransform Matrix="1,0,0,1,102,-36" />
    </Canvas.RenderTransform>
    <Path Name="_7_G0_P0" Data="M1.92 1054.08 L118.08 1054.08 L118.08
1038.57080078125 L1.92 1038.57080078125 L1.92 1054.08 Z"></Path>
    <Path Name="_7_G1_P1" Data="M1.92 1054.08 L118.08 1054.08"></Path>
    <Path Name="_7_G0_P0_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08 L118.08 1038.57080078125 L1.92 1038.57080078125 L1.92 1054.08 Z">
        <Path.Stroke>
            <SolidColorBrush Opacity="0" Color="#FF404040" />
        </Path.Stroke>
    </Path>
    <Path Name="_7_G1_P1_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08">
        <Path.Stroke>
            <SolidColorBrush Color="#FF404040" />
        </Path.Stroke>
    </Path>
</Canvas>
    <TextBlock
Tag="Left;Top;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888889"
Height="0.201553942620754" Visibility="Collapsed" Width="0.75625" RenderTransform="96, 0, 0,
96, 1.92, 1038.608582" TextWrapping="Wrap">
    <Run Name="txt_7_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">FN </Run>
</TextBlock>

```

```

<Canvas>
  <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="3.2534" OriginY="1049.5" UnicodeString="FN">
    <Glyphs.Fill>
      <SolidColorBrush Color="#FF000000" />
    </Glyphs.Fill>
  </Glyphs>
</Canvas>
<Canvas Name="_8">
  <TextBlock
Tag="Right;Center;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888888
9" Height="0.201553942620754" Visibility="Collapsed" Width="1.0342764082339"
RenderTransform="96, 0, 0, 96, 18.789465, 1038.608582" TextWrapping="Wrap">
  <Run Name="txt_8_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">not Happy </Run>
</TextBlock>
<Canvas>
  <Canvas.RenderTransform>
    <MatrixTransform Matrix="96,0,0,96,18.7872,1038.571" />
  </Canvas.RenderTransform>
  <Glyphs UnicodeString="Happy" OriginX="0.787411080680788"
OriginY="0.111110001802444" FontUri="CommonFonts/4C1AABE5-BA71-4F2C-8263-66731F995C4B.odttf"
FontRenderingEmSize="0.111110001802444" Fill="#FF000000" Visibility="Visible" />
</Canvas>
</Canvas>
<Canvas Name="_4">
  <Path Name="_4_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M0 1056
L72 1056 L96 1032 L96 984 L0 984 L0 1056 Z">
    <Path.Stroke>
      <SolidColorBrush Color="#FF404040" />
    </Path.Stroke>
    <Path.Fill>
      <LinearGradientBrush StartPoint="0.5,0" EndPoint="0.5,1">
        <LinearGradientBrush.GradientStops>
          <GradientStop Color="#FFFAFAFA" />
          <GradientStop Color="#FFFFFFF" Offset="1" />
        </LinearGradientBrush.GradientStops>
      </LinearGradientBrush>
    </Path.Fill>
  </Path>
</Canvas>
</Canvas>
<Canvas
Tag="\{&quot;type&quot;:&quot;2D&quot;,&quot;bounds&quot;:\{&quot;x&quot;:96,&quot;y&quot;:45
6,&quot;width&quot;:96,&quot;height&quot;:72\}\}" Name="_9" RenderTransform="1, 0, 0, 1, 96,
-528">
  <Canvas Name="_10">
    <Canvas Tag="Shdw" RenderTransform="1, 0, 0, 1, 1.92, 1.92">
      <Path Name="S_10_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M72 1056
L96 1056 L96 1032 L96 984 L0 984 L0 1056 L72 1056 Z">
        <Path.Stroke>
          <SolidColorBrush Color="#7FDEDEDE" />
        </Path.Stroke>
        <Path.Fill>
          <SolidColorBrush Color="#7FDEDEDE" />
        </Path.Fill>
      </Path>
    </Canvas>
  </Canvas>

```

```

        </Canvas>
        <Path Name="_10_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M72 1056
L96 1056 L96 1032 L96 984 L0 984 L0 1056 L72 1056 Z">
        <Path.Stroke>
        <SolidColorBrush Color="#00404040" />
        </Path.Stroke>
        <Path.Fill>
        <LinearGradientBrush StartPoint="0.5,0" EndPoint="0.5,1">
        <LinearGradientBrush.GradientStops>
        <GradientStop Color="#00FF00FF" />
        <GradientStop Color="#00FF00FF" Offset="1" />
        </LinearGradientBrush.GradientStops>
        </LinearGradientBrush>
        </Path.Fill>
        </Path>
    </Canvas>
    <Canvas Name="_11" RenderTransform="1, 0, 0, 1, 72, 0">
        <Path Name="_11_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M0 1056
L24 1056 L24 1032 L0 1056 Z">
        <Path.Stroke>
        <SolidColorBrush Color="#FF404040" />
        </Path.Stroke>
        <Path.Fill>
        <SolidColorBrush Color="#FF000000" />
        </Path.Fill>
        </Path>
    </Canvas>
    <Canvas Name="_13">
        <Canvas.RenderTransform>
        <MatrixTransform Matrix="1, 0, 0, 1, 102, -16.688582" />
        </Canvas.RenderTransform>
        <Path Name="_13_G0_P0" Data="M1.92 1054.08 L118.08 1054.08 L118.08
1038.61 L1.92 1038.61 L1.92 1054.08 Z" />
        <Path Name="_13_G1_P1" Data="M1.92 1054.08 L118.08 1054.08" />
        <Path Name="_13_G0_P0_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08 L118.08 1038.61 L1.92 1038.61 L1.92 1054.08">
        <Path.Stroke>
        <SolidColorBrush Opacity="0" Color="#FF404040" />
        </Path.Stroke>
        </Path>
        <Path Name="_13_G1_P1_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08">
        <Path.Stroke>
        <SolidColorBrush Color="#FF404040" />
        </Path.Stroke>
        </Path>
        <TextBlock
Tag="Left;Top;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888889"
Height="0.16116" Visibility="Collapsed" Width="0.75625" RenderTransform="96, 0, 0, 96, 1.92,
1038.608582" TextWrapping="Wrap">
        <Run Name="txt_13_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">LN </Run>
        </TextBlock>
    </Canvas>
        <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="3.2534" OriginY="1049.5" UnicodeString="LN">

```

```

        <Glyphs.Fill>
            <SolidColorBrush Color="#FF000000" />
        </Glyphs.Fill>
    </Glyphs>
</Canvas>
<Canvas Name="_14">
    <TextBlock
Tag="Right;Center;;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888888
9" Height="0.16116" Visibility="Collapsed" Width="1.0386" RenderTransform="96, 0, 0, 96,
18.372717, 1038.608582" TextWrapping="Wrap">
        <Run Name="txt_14_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">dog </Run>
    </TextBlock>
</Canvas>
    <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="100.5" OriginY="1049.5" UnicodeString="dog">
        <Glyphs.Fill>
            <SolidColorBrush Color="#FF000000" />
        </Glyphs.Fill>
    </Glyphs>
</Canvas>
</Canvas>
<Canvas Name="_15">
    <Canvas.RenderTransform>
        <MatrixTransform Matrix="1, 0, 0, 1, 102, -36" />
    </Canvas.RenderTransform>
    <Path Name="_15_G0_P0" Data="M1.92 1054.08 L118.08 1054.08 L118.08
1038.61 L1.92 1038.61 L1.92 1054.08 Z" />
    <Path Name="_15_G1_P1" Data="M1.92 1054.08 L118.08 1054.08" />
    <Path Name="_15_G0_P0_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08 L118.08 1038.61 L1.92 1038.61 L1.92 1054.08">
        <Path.Stroke>
            <SolidColorBrush Opacity="0" Color="#FF404040" />
        </Path.Stroke>
    </Path>
    <Path Name="_15_G1_P1_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08">
        <Path.Stroke>
            <SolidColorBrush Color="#FF404040" />
        </Path.Stroke>
    </Path>
    <TextBlock
Tag="Left;Top;;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888889"
Height="0.16116" Visibility="Collapsed" Width="0.75625" RenderTransform="96, 0, 0, 96, 1.92,
1038.608582" TextWrapping="Wrap">
        <Run Name="txt_15_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">FN </Run>
    </TextBlock>
</Canvas>
    <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="3.2534" OriginY="1049.5" UnicodeString="FN">
        <Glyphs.Fill>
            <SolidColorBrush Color="#FF000000" />
        </Glyphs.Fill>
    </Glyphs>
</Canvas>
<Canvas Name="_16">

```



```

        <TextBlock
Tag="Right;Center;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888888
9" Height="0.16116" Visibility="Collapsed" Width="1.0343" RenderTransform="96, 0, 0, 96,
18.789465, 1038.608582" TextWrapping="Wrap">
    <Run Name="txt_16_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">Silly </Run>
    </TextBlock>
    <Canvas>
    <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="84.559" OriginY="1049.5" UnicodeString="Silly">
    <Glyphs.Fill>
    <SolidColorBrush Color="#FF000000" />
    </Glyphs.Fill>
    </Glyphs>
    </Canvas>
    </Canvas>
    </Canvas>
    <Canvas Name="_12">
    <Path Name="_12_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M0 1056
L72 1056 L96 1032 L96 984 L0 984 L0 1056 Z">
    <Path.Stroke>
    <SolidColorBrush Color="#FF404040" />
    </Path.Stroke>
    <Path.Fill>
    <LinearGradientBrush StartPoint="0.5,0" EndPoint="0.5,1">
    <LinearGradientBrush.GradientStops>
    <GradientStop Color="#FFFAFAFA" />
    <GradientStop Color="#FFFFFF" Offset="1" />
    </LinearGradientBrush.GradientStops>
    </LinearGradientBrush>
    </Path.Fill>
    </Path>
    </Canvas>
    </Canvas>
    </Canvas>
    <Canvas
Tag="\{&quot;type&quot;:&quot;2D&quot;,&quot;bounds&quot;:\{&quot;x&quot;:120,&quot;y&quot;:6
48,&quot;width&quot;:96,&quot;height&quot;:72\}\}" Name="_17" RenderTransform="1, 0, 1,
120, -336">
    <Canvas Name="_18">
    <Canvas Tag="Shdw" RenderTransform="1, 0, 0, 1, 1.92, 1.92">
    <Path Name="S_18_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M72 1056
L96 1056 L96 1032 L96 984 L0 984 L0 1056 L72 1056 Z">
    <Path.Stroke>
    <SolidColorBrush Color="#7FDEDEDE" />
    </Path.Stroke>
    <Path.Fill>
    <SolidColorBrush Color="#7FDEDEDE" />
    </Path.Fill>
    </Path>
    </Canvas>
    <Path Name="_18_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M72 1056
L96 1056 L96 1032 L96 984 L0 984 L0 1056 L72 1056 Z">
    <Path.Stroke>
    <SolidColorBrush Color="#00404040" />
    </Path.Stroke>
    <Path.Fill>
    <LinearGradientBrush StartPoint="0.5,0" EndPoint="0.5,1">

```

```

        <LinearGradientBrush.GradientStops>
            <GradientStop Color="#00FF00FF" />
            <GradientStop Color="#00FF00FF" Offset="1" />
        </LinearGradientBrush.GradientStops>
    </LinearGradientBrush>
</Path.Fill>
</Path>
</Canvas>
<Canvas Name="_19" RenderTransform="1, 0, 0, 1, 72, 0">
    <Path Name="_19_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M0 1056
L24 1056 L24 1032 L0 1056 Z">
        <Path.Stroke>
            <SolidColorBrush Color="#FF404040" />
        </Path.Stroke>
        <Path.Fill>
            <SolidColorBrush Color="#FF000000" />
        </Path.Fill>
    </Path>
</Canvas>
<Canvas Name="_21">
    <Canvas.RenderTransform>
        <MatrixTransform Matrix="1, 0, 0, 1, 102, -16.688582" />
    </Canvas.RenderTransform>
    <Path Name="_21_G0_P0" Data="M1.92 1054.08 L118.08 1054.08 L118.08
1038.61 L1.92 1038.61 L1.92 1054.08 Z" />
    <Path Name="_21_G1_P1" Data="M1.92 1054.08 L118.08 1054.08" />
    <Path Name="_21_G0_P0_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08 L118.08 1038.61 L1.92 1038.61 L1.92 1054.08">
        <Path.Stroke>
            <SolidColorBrush Opacity="0" Color="#FF404040" />
        </Path.Stroke>
    </Path>
    <Path Name="_21_G1_P1_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08">
        <Path.Stroke>
            <SolidColorBrush Color="#FF404040" />
        </Path.Stroke>
    </Path>
    <TextBlock
Tag="Left;Top;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888889"
Height="0.16116" Visibility="Collapsed" Width="0.75625" RenderTransform="96, 0, 0, 96, 1.92,
1038.608582" TextWrapping="Wrap">
        <Run Name="txt_21_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">LN </Run>
    </TextBlock>
</Canvas>
    <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="3.2534" OriginY="1049.5" UnicodeString="LN">
        <Glyphs.Fill>
            <SolidColorBrush Color="#FF000000" />
        </Glyphs.Fill>
    </Glyphs>
</Canvas>
<Canvas Name="_22">
    <TextBlock
Tag="Right;Center;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888888

```

```

9" Height="0.16116" Visibility="Collapsed" Width="1.0386" RenderTransform="96, 0, 0, 96,
18.372717, 1038.608582" TextWrapping="Wrap">
  <Run Name="txt_22_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">Id </Run>
</TextBlock>
</Canvas>
  <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="103.67" OriginY="1049.5" UnicodeString="Id">
  <Glyphs.Fill>
    <SolidColorBrush Color="#FF000000" />
  </Glyphs.Fill>
</Glyphs>
</Canvas>
</Canvas>
</Canvas>
<Canvas Name="_23">
  <Canvas.RenderTransform>
    <MatrixTransform Matrix="1, 0, 0, 1, 102, -36" />
  </Canvas.RenderTransform>
  <Path Name="_23_G0_P0" Data="M1.92 1054.08 L118.08 1054.08 L118.08
1038.61 L1.92 1038.61 L1.92 1054.08 Z" />
  <Path Name="_23_G1_P1" Data="M1.92 1054.08 L118.08 1054.08" />
  <Path Name="_23_G0_P0_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08 L118.08 1038.61 L1.92 1038.61 L1.92 1054.08">
  <Path.Stroke>
    <SolidColorBrush Opacity="0" Color="#FF404040" />
  </Path.Stroke>
</Path>
  <Path Name="_23_G1_P1_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08">
  <Path.Stroke>
    <SolidColorBrush Color="#FF404040" />
  </Path.Stroke>
</Path>
</TextBlock>
Tag="Left;Top;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888889"
Height="0.16116" Visibility="Collapsed" Width="0.75625" RenderTransform="96, 0, 0, 96, 1.92,
1038.608582" TextWrapping="Wrap">
  <Run Name="txt_23_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">FN </Run>
</TextBlock>
</Canvas>
  <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="3.2534" OriginY="1049.5" UnicodeString="FN">
  <Glyphs.Fill>
    <SolidColorBrush Color="#FF000000" />
  </Glyphs.Fill>
</Glyphs>
</Canvas>
<Canvas Name="_24">
  <TextBlock>
    Tag="Right;Center;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888888
9" Height="0.16116" Visibility="Collapsed" Width="1.0343" RenderTransform="96, 0, 0, 96,
18.789465, 1038.608582" TextWrapping="Wrap">
    <Run Name="txt_24_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">Order </Run>
  </TextBlock>
</Canvas>

```

```

        <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
        BA71-4F2C-8263-66731F995C4B.odttf" OriginX="93.242" OriginY="1049.5" UnicodeString="Order">
            <Glyphs.Fill>
                <SolidColorBrush Color="#FF000000" />
            </Glyphs.Fill>
        </Glyphs>
    </Canvas>
</Canvas>
</Canvas>
<Canvas Name="_20">
    <Path Name="_20_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
    StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M0 1056
    L72 1056 L96 1032 L96 984 L0 984 L0 1056 Z">
        <Path.Stroke>
            <SolidColorBrush Color="#FF404040" />
        </Path.Stroke>
        <Path.Fill>
            <LinearGradientBrush StartPoint="0.5,0" EndPoint="0.5,1">
                <LinearGradientBrush.GradientStops>
                    <GradientStop Color="#FFFAFAFA" />
                    <GradientStop Color="#FFFFFF" Offset="1" />
                </LinearGradientBrush.GradientStops>
            </LinearGradientBrush>
        </Path.Fill>
    </Path>
</Canvas>
</Canvas>
<Canvas
Tag="{&quot;type&quot;:&quot;2D&quot;,&quot;bounds&quot;:{&quot;x&quot;:480,&quot;y&quot;:4
92,&quot;width&quot;:96,&quot;height&quot;:72\}}" Name="_25" RenderTransform="1, 0, 0, 1,
480, -492">
    <Canvas Name="_26">
        <Canvas Tag="Shdw" RenderTransform="1, 0, 0, 1, 1.92, 1.92">
            <Path Name="S_26_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
            StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M72 1056
            L96 1056 L96 1032 L96 984 L0 984 L0 1056 L72 1056 Z">
                <Path.Stroke>
                    <SolidColorBrush Color="#7FDEDEDE" />
                </Path.Stroke>
                <Path.Fill>
                    <SolidColorBrush Color="#7FDEDEDE" />
                </Path.Fill>
            </Path>
        </Canvas>
        <Path Name="_26_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
        StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M72 1056
        L96 1056 L96 1032 L96 984 L0 984 L0 1056 L72 1056 Z">
            <Path.Stroke>
                <SolidColorBrush Color="#00404040" />
            </Path.Stroke>
            <Path.Fill>
                <LinearGradientBrush StartPoint="0.5,0" EndPoint="0.5,1">
                    <LinearGradientBrush.GradientStops>
                        <GradientStop Color="#00FF00FF" />
                        <GradientStop Color="#00FF00FF" Offset="1" />
                    </LinearGradientBrush.GradientStops>
                </LinearGradientBrush>
            </Path.Fill>
        </Path>
    </Canvas>

```

```

<Canvas Name="_27" RenderTransform="1, 0, 0, 1, 72, 0">
  <Path Name="_27_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M0 1056
L24 1056 L24 1032 L0 1056 Z">
    <Path.Stroke>
      <SolidColorBrush Color="#FF404040" />
    </Path.Stroke>
    <Path.Fill>
      <SolidColorBrush Color="#FF000000" />
    </Path.Fill>
  </Path>
</Canvas>
<Canvas Name="_29">
  <Canvas.RenderTransform>
    <MatrixTransform Matrix="1, 0, 0, 1, 102, -16.688582" />
  </Canvas.RenderTransform>
  <Path Name="_29_G0_P0" Data="M1.92 1054.08 L118.08 1054.08 L118.08
1038.61 L1.92 1038.61 L1.92 1054.08 Z" />
  <Path Name="_29_G1_P1" Data="M1.92 1054.08 L118.08 1054.08" />
  <Path Name="_29_G0_P0_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08 L118.08 1038.61 L1.92 1038.61 L1.92 1054.08">
    <Path.Stroke>
      <SolidColorBrush Opacity="0" Color="#FF404040" />
    </Path.Stroke>
  </Path>
  <Path Name="_29_G1_P1_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08">
    <Path.Stroke>
      <SolidColorBrush Color="#FF404040" />
    </Path.Stroke>
  </Path>
  <TextBlock
Tag="Left;Top;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888889"
Height="0.16116" Visibility="Collapsed" Width="0.75625" RenderTransform="96, 0, 0, 96, 1.92,
1038.608582" TextWrapping="Wrap">
    <Run Name="txt_29_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">LN </Run>
  </TextBlock>
</Canvas>
  <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="3.2534" OriginY="1049.5" UnicodeString="LN">
    <Glyphs.Fill>
      <SolidColorBrush Color="#FF000000" />
    </Glyphs.Fill>
  </Glyphs>
</Canvas>
<Canvas Name="_30">
  <TextBlock
Tag="Right;Center;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888888
9" Height="0.16116" Visibility="Collapsed" Width="1.0386" RenderTransform="96, 0, 0, 96,
18.372717, 1038.608582" TextWrapping="Wrap">
    <Run Name="txt_30_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">turtle </Run>
  </TextBlock>
</Canvas>
  <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="92.523" OriginY="1049.5" UnicodeString="turtle">
    <Glyphs.Fill>

```

```

        <SolidColorBrush Color="#FF000000" />
    </Glyphs.Fill>
</Glyphs>
</Canvas>
</Canvas>
</Canvas>
<Canvas Name="_31">
    <Canvas.RenderTransform>
        <MatrixTransform Matrix="1, 0, 0, 1, 102, -36" />
    </Canvas.RenderTransform>
    <Path Name="_31_G0_P0" Data="M1.92 1054.08 L118.08 1054.08 L118.08
1038.61 L1.92 1038.61 L1.92 1054.08 Z" />
    <Path Name="_31_G1_P1" Data="M1.92 1054.08 L118.08 1054.08" />
    <Path Name="_31_G0_P0_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08 L118.08 1038.61 L1.92 1038.61 L1.92 1054.08">
        <Path.Stroke>
            <SolidColorBrush Opacity="0" Color="#FF404040" />
        </Path.Stroke>
    </Path>
    <Path Name="_31_G1_P1_line" StrokeThickness="0.5" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="M1.92
1054.08 L118.08 1054.08">
        <Path.Stroke>
            <SolidColorBrush Color="#FF404040" />
        </Path.Stroke>
    </Path>
    <TextBlock
Tag="Left;Top;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888889"
Height="0.16116" Visibility="Collapsed" Width="0.75625" RenderTransform="96, 0, 0, 96, 1.92,
1038.608582" TextWrapping="Wrap">
        <Run Name="txt_31_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">FN </Run>
    </TextBlock>
    <Canvas>
        <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="3.2534" OriginY="1049.5" UnicodeString="FN">
            <Glyphs.Fill>
                <SolidColorBrush Color="#FF000000" />
            </Glyphs.Fill>
        </Glyphs>
    </Canvas>
    <Canvas Name="_32">
        <TextBlock
Tag="Right;Center;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888888
9" Height="0.16116" Visibility="Collapsed" Width="1.0343" RenderTransform="96, 0, 0, 96,
18.789465, 1038.608582" TextWrapping="Wrap">
            <Run Name="txt_32_fld0" FontFamily="Calibri" FontSize="0.11111"
Foreground="#FF000000">green </Run>
        </TextBlock>
        <Canvas>
            <Glyphs FontRenderingEmSize="10.667" FontUri="CommonFonts/4C1AABE5-
BA71-4F2C-8263-66731F995C4B.odttf" OriginX="91.789" OriginY="1049.5" UnicodeString="green">
                <Glyphs.Fill>
                    <SolidColorBrush Color="#FF000000" />
                </Glyphs.Fill>
            </Glyphs>
        </Canvas>
    </Canvas>
</Canvas>

```

```
<Canvas Name="_28">
  <Path Name="_28_G0" StrokeThickness="0.33333" StrokeDashCap="Round"
StrokeEndLineCap="Round" StrokeLineJoin="Round" StrokeStartLineCap="Round" Data="F0 M0 1056
L72 1056 L96 1032 L96 984 L0 984 L0 1056 Z">
  <Path.Stroke>
    <SolidColorBrush Color="#FF404040" />
  </Path.Stroke>
  <Path.Fill>
    <LinearGradientBrush StartPoint="0.5,0" EndPoint="0.5,1">
      <LinearGradientBrush.GradientStops>
        <GradientStop Color="#FFFAFAFA" />
        <GradientStop Color="#FFFFFF" Offset="1" />
      </LinearGradientBrush.GradientStops>
    </LinearGradientBrush>
  </Path.Fill>
</Path>
</Canvas>
</Canvas>
</Canvas>
</Canvas>
</d4p1:Xaml>
</d4p1:OutputPage>

</GetVectorDiagramResult>
</GetVectorDiagramResponse>
</s:Body>
</s:Envelope>
```

5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.

Preliminary

6 Appendix A: Full WSDL

For ease of implementation, the full WSDL is provided in this appendix.

```
<?xml version="1.0"?>
<wsdl:definitions xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl" xmlns:tns2="http://tempuri.org/"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:wsp="http://schemas.xmlsoap.org/ws/2004/09/policy" xmlns:wsu="http://docs.oasis-
open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd" name="VisioGraphicsService"
targetNamespace="http://tempuri.org/" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsp:Policy wsu:Id="CustomBinding_IVisioGraphicsService_policy">
    <wsp:ExactlyOne>
      <wsp:All>
        <sp:SymmetricBinding xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
          <wsp:Policy>
            <sp:ProtectionToken>
              <wsp:Policy>
                <sp:IssuedToken
sp:IncludeToken="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy/IncludeToken/AlwaysToRe
cipient">
                  <sp:RequestSecurityTokenTemplate>
                    <t:KeyType
xmlns:t="http://schemas.xmlsoap.org/ws/2005/02/trust">http://schemas.xmlsoap.org/ws/2005/02/t
rust/SymmetricKey</t:KeyType>
                    </sp:RequestSecurityTokenTemplate>
                    <wsp:Policy>
                      <sp:RequireDerivedKeys/>
                      <sp:RequireInternalReference/>
                    </wsp:Policy>
                  </sp:IssuedToken>
                </wsp:Policy>
              </sp:ProtectionToken>
            <sp:AlgorithmSuite>
              <wsp:Policy>
                <sp:Basic256/>
              </wsp:Policy>
            </sp:AlgorithmSuite>
            <sp:Layout>
              <wsp:Policy>
                <sp:Strict/>
              </wsp:Policy>
            </sp:Layout>
            <sp:IncludeTimestamp/>
            <sp:EncryptSignature/>
            <sp:OnlySignEntireHeadersAndBody/>
          </wsp:Policy>
        </sp:SymmetricBinding>
      <sp:Wss11 xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <wsp:Policy/>
      </sp:Wss11>
      <sp:Trust10 xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <wsp:Policy>
          <sp:MustSupportIssuedTokens/>
          <sp:RequireClientEntropy/>
          <sp:RequireServerEntropy/>
        </wsp:Policy>
      </sp:Trust10>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
```

```

    <msb:BinaryEncoding
xmlns:msb="http://schemas.microsoft.com/ws/06/2004/mspolicy/netbinary1"/>
    <wsaw:UsingAddressing/>
  </wsp:All>
</wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy wsu:Id="CustomBinding_IVisioGraphicsService_GetRasterDiagram_Input_policy">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>
      <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
      </sp:EncryptedParts>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy wsu:Id="CustomBinding_IVisioGraphicsService_GetRasterDiagram_output_policy">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>
      <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
      </sp:EncryptedParts>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy
wsu:Id="CustomBinding_IVisioGraphicsService_GetRasterDiagram_VisioGraphicsServiceFaultFault_F
ault">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>

```

```

    <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
      <sp:Body/>
    </sp:EncryptedParts>
  </wsp:All>
</wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy wsu:Id="CustomBinding_IVisioGraphicsService_GetVectorDiagram_Input_policy">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>
      <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
      </sp:EncryptedParts>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy wsu:Id="CustomBinding_IVisioGraphicsService_GetVectorDiagram_output_policy">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>
      <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
      </sp:EncryptedParts>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy
wsu:Id="CustomBinding_IVisioGraphicsService_GetVectorDiagram_VisioGraphicsServiceFaultFault_F
ault">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>

```

```

    <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
      <sp:Body/>
    </sp:EncryptedParts>
  </wsp:All>
</wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy wsu:Id="CustomBinding_IVisioGraphicsService_GetRasterPage_Input_policy">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>
      <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
      </sp:EncryptedParts>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy wsu:Id="CustomBinding_IVisioGraphicsService_GetRasterPage_output_policy">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>
      <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
      </sp:EncryptedParts>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy
wsu:Id="CustomBinding_IVisioGraphicsService_GetRasterPage_VisioGraphicsServiceFaultFault_Fault">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>

```

```

    <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
      <sp:Body/>
    </sp:EncryptedParts>
  </wsp:All>
</wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy wsu:Id="CustomBinding_IVisioGraphicsService_GetRasterPageItem_Input_policy">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>
      <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
      </sp:EncryptedParts>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy wsu:Id="CustomBinding_IVisioGraphicsService_GetRasterPageItem_output_policy">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>
      <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
      </sp:EncryptedParts>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy
wsu:Id="CustomBinding_IVisioGraphicsService_GetRasterPageItem_VisioGraphicsServiceFaultFault_Fault">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>

```

```

    <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
      <sp:Body/>
    </sp:EncryptedParts>
  </wsp:All>
</wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy wsu:Id="CustomBinding_IVisioGraphicsService_SaveComments_Input_policy">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>
      <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
      </sp:EncryptedParts>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy wsu:Id="CustomBinding_IVisioGraphicsService_SaveComments_output_policy">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>
      <sp:EncryptedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
      </sp:EncryptedParts>
    </wsp:All>
  </wsp:ExactlyOne>
</wsp:Policy>
<wsp:Policy
wsu:Id="CustomBinding_IVisioGraphicsService_SaveComments_VisioGraphicsServiceFaultFault_Fault
">
  <wsp:ExactlyOne>
    <wsp:All>
      <sp:SignedParts xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy">
        <sp:Body/>
        <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="MessageID" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
        <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
      </sp:SignedParts>

```



```

    </wsp:All>
    </wsp:ExactlyOne>
  </wsp:Policy>
  <wsdl:types>
    <xs:schema xmlns:tns3="http://tempuri.org/Imports"
targetNamespace="http://tempuri.org/Imports">
      <xs:import
schemaLocation="http://www.example.com/bfa9dba966bd4821ac2fca057487aaa8/VisioGraphicsService.
svc?xsd=xsd0" namespace="http://tempuri.org/">
        <xs:import
schemaLocation="http://www.example.com/bfa9dba966bd4821ac2fca057487aaa8/VisioGraphicsService.
svc?xsd=xsd2"
namespace="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er"/>
          <xs:import
schemaLocation="http://www.example.com/bfa9dba966bd4821ac2fca057487aaa8/VisioGraphicsService.
svc?xsd=xsd1" namespace="http://schemas.microsoft.com/2003/10/Serialization/">
            <xs:import
schemaLocation="http://www.example.com/bfa9dba966bd4821ac2fca057487aaa8/VisioGraphicsService.
svc?xsd=xsd3"/>
              <xs:import
schemaLocation="http://www.example.com/bfa9dba966bd4821ac2fca057487aaa8/VisioGraphicsService.
svc?xsd=xsd4" namespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays"/>
                </xs:schema>
          </wsdl:types>
          <wsdl:message name="IVisioGraphicsService_GetRasterDiagram_InputMessage">
            <wsdl:part name="parameters" element="tns2:GetRasterDiagram"/>
          </wsdl:message>
          <wsdl:message name="IVisioGraphicsService_GetRasterDiagram_OutputMessage">
            <wsdl:part name="parameters" element="tns2:GetRasterDiagramResponse"/>
          </wsdl:message>
          <wsdl:message
name="IVisioGraphicsService_GetRasterDiagram_VisioGraphicsServiceFaultFault_FaultMessage">
            <wsdl:part
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" name="detail" element="tns:VisioGraphicsServiceFault"/>
          </wsdl:message>
          <wsdl:message name="IVisioGraphicsService_GetVectorDiagram_InputMessage">
            <wsdl:part name="parameters" element="tns2:GetVectorDiagram"/>
          </wsdl:message>
          <wsdl:message name="IVisioGraphicsService_GetVectorDiagram_OutputMessage">
            <wsdl:part name="parameters" element="tns2:GetVectorDiagramResponse"/>
          </wsdl:message>
          <wsdl:message
name="IVisioGraphicsService_GetVectorDiagram_VisioGraphicsServiceFaultFault_FaultMessage">
            <wsdl:part
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" name="detail" element="tns:VisioGraphicsServiceFault"/>
          </wsdl:message>
          <wsdl:message name="IVisioGraphicsService_GetRasterPage_InputMessage">
            <wsdl:part name="parameters" element="tns2:GetRasterPage"/>
          </wsdl:message>
          <wsdl:message name="IVisioGraphicsService_GetRasterPage_OutputMessage">
            <wsdl:part name="parameters" element="tns2:GetRasterPageResponse"/>
          </wsdl:message>
          <wsdl:message
name="IVisioGraphicsService_GetRasterPage_VisioGraphicsServiceFaultFault_FaultMessage">
            <wsdl:part
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" name="detail" element="tns:VisioGraphicsServiceFault"/>
          </wsdl:message>

```



```

<wsdl:message name="IVisioGraphicsService_GetRasterPageItem_InputMessage">
  <wsdl:part name="parameters" element="tns2:GetRasterPageItem"/>
</wsdl:message>
<wsdl:message name="IVisioGraphicsService_GetRasterPageItem_OutputMessage">
  <wsdl:part name="parameters" element="tns2:GetRasterPageItemResponse"/>
</wsdl:message>
<wsdl:message
name="IVisioGraphicsService_GetRasterPageItem_VisioGraphicsServiceFaultFault_FaultMessage">
  <wsdl:part
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" name="detail" element="tns:VisioGraphicsServiceFault"/>
</wsdl:message>
<wsdl:message name="IVisioGraphicsService_SaveComments_InputMessage">
  <wsdl:part name="parameters" element="tns2:SaveComments"/>
</wsdl:message>
<wsdl:message name="IVisioGraphicsService_SaveComments_OutputMessage">
  <wsdl:part name="parameters" element="tns2:SaveCommentsResponse"/>
</wsdl:message>
<wsdl:message
name="IVisioGraphicsService_SaveComments_VisioGraphicsServiceFaultFault_FaultMessage">
  <wsdl:part
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" name="detail" element="tns:VisioGraphicsServiceFault"/>
</wsdl:message>
<wsdl:portType name="IVisioGraphicsService">
  <wsdl:operation name="GetRasterDiagram">
    <wsdl:input wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterDiagram"
message="tns2:IVisioGraphicsService_GetRasterDiagram_InputMessage"/>
    <wsdl:output
wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterDiagramResponse"
message="tns2:IVisioGraphicsService_GetRasterDiagram_OutputMessage"/>
    <wsdl:fault
wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterDiagramVisioGraphicsServiceFau
ltFault" name="VisioGraphicsServiceFaultFault"
message="tns2:IVisioGraphicsService_GetRasterDiagram_VisioGraphicsServiceFaultFault_FaultMess
age"/>
  </wsdl:operation>
  <wsdl:operation name="GetVectorDiagram">
    <wsdl:input wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetVectorDiagram"
message="tns2:IVisioGraphicsService_GetVectorDiagram_InputMessage"/>
    <wsdl:output
wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetVectorDiagramResponse"
message="tns2:IVisioGraphicsService_GetVectorDiagram_OutputMessage"/>
    <wsdl:fault
wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetVectorDiagramVisioGraphicsServiceFau
ltFault" name="VisioGraphicsServiceFaultFault"
message="tns2:IVisioGraphicsService_GetVectorDiagram_VisioGraphicsServiceFaultFault_FaultMess
age"/>
  </wsdl:operation>
  <wsdl:operation name="GetRasterPage">
    <wsdl:input wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPage"
message="tns2:IVisioGraphicsService_GetRasterPage_InputMessage"/>
    <wsdl:output
wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPageResponse"
message="tns2:IVisioGraphicsService_GetRasterPage_OutputMessage"/>
    <wsdl:fault
wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPageVisioGraphicsServiceFau
ltFault" name="VisioGraphicsServiceFaultFault"
message="tns2:IVisioGraphicsService_GetRasterPage_VisioGraphicsServiceFaultFault_FaultMess
age"/>
  </wsdl:operation>

```

```

    <wsdl:operation name="GetRasterPageItem">
      <wsdl:input wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPageItem"
message="tns2:IVisioGraphicsService_GetRasterPageItem_InputMessage"/>
      <wsdl:output
wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPageItemResponse"
message="tns2:IVisioGraphicsService_GetRasterPageItem_OutputMessage"/>
      <wsdl:fault
wsaw:Action="http://tempuri.org/IVisioGraphicsService/GetRasterPageItemVisioGraphicsServiceFaultFault"
name="VisioGraphicsServiceFaultFault"
message="tns2:IVisioGraphicsService_GetRasterPageItem_VisioGraphicsServiceFaultFault_FaultMessage"/>
    </wsdl:operation>
    <wsdl:operation name="SaveComments">
      <wsdl:input wsaw:Action="http://tempuri.org/IVisioGraphicsService/SaveComments"
message="tns2:IVisioGraphicsService_SaveComments_InputMessage"/>
      <wsdl:output
wsaw:Action="http://tempuri.org/IVisioGraphicsService/SaveCommentsResponse"
message="tns2:IVisioGraphicsService_SaveComments_OutputMessage"/>
      <wsdl:fault
wsaw:Action="http://tempuri.org/IVisioGraphicsService/SaveCommentsVisioGraphicsServiceFaultFault"
name="VisioGraphicsServiceFaultFault"
message="tns2:IVisioGraphicsService_SaveComments_VisioGraphicsServiceFaultFault_FaultMessage"
/>
    </wsdl:operation>
  </wsdl:portType>
  <wsdl:binding name="CustomBinding_IVisioGraphicsService" type="tns2:IVisioGraphicsService">
    <wsp:PolicyReference URI="#CustomBinding_IVisioGraphicsService_policy"/>
    <soap12:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="GetRasterDiagram">
      <soap12:operation
soapAction="http://tempuri.org/IVisioGraphicsService/GetRasterDiagram" style="document"/>
      <wsdl:input>
        <wsp:PolicyReference
URI="#CustomBinding_IVisioGraphicsService_GetRasterDiagram_Input_policy"/>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <wsp:PolicyReference
URI="#CustomBinding_IVisioGraphicsService_GetRasterDiagram_output_policy"/>
        <soap12:body use="literal"/>
      </wsdl:output>
      <wsdl:fault name="VisioGraphicsServiceFaultFault">
        <wsp:PolicyReference
URI="#CustomBinding_IVisioGraphicsService_GetRasterDiagram_VisioGraphicsServiceFaultFault_Fault"
lt"/>
        <soap12:fault use="literal" name="VisioGraphicsServiceFaultFault" namespace=""/>
      </wsdl:fault>
    </wsdl:operation>
    <wsdl:operation name="GetVectorDiagram">
      <soap12:operation
soapAction="http://tempuri.org/IVisioGraphicsService/GetVectorDiagram" style="document"/>
      <wsdl:input>
        <wsp:PolicyReference
URI="#CustomBinding_IVisioGraphicsService_GetVectorDiagram_Input_policy"/>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <wsp:PolicyReference
URI="#CustomBinding_IVisioGraphicsService_GetVectorDiagram_output_policy"/>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>

```

```

        <wsdl:fault name="VisioGraphicsServiceFaultFault">
          <wsp:PolicyReference
            URI="#CustomBinding_IVisioGraphicsService_GetVectorDiagram_VisioGraphicsServiceFaultFault_Fault"/>
          <soap12:fault use="literal" name="VisioGraphicsServiceFaultFault" namespace=""/>
        </wsdl:fault>
      </wsdl:operation>
      <wsdl:operation name="GetRasterPage">
        <soap12:operation soapAction="http://tempuri.org/IVisioGraphicsService/GetRasterPage"
          style="document"/>
        <wsdl:input>
          <wsp:PolicyReference
            URI="#CustomBinding_IVisioGraphicsService_GetRasterPage_Input_policy"/>
          <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <wsp:PolicyReference
            URI="#CustomBinding_IVisioGraphicsService_GetRasterPage_output_policy"/>
          <soap12:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="VisioGraphicsServiceFaultFault">
          <wsp:PolicyReference
            URI="#CustomBinding_IVisioGraphicsService_GetRasterPage_VisioGraphicsServiceFaultFault_Fault"/>
        </wsdl:fault>
        <soap12:fault use="literal" name="VisioGraphicsServiceFaultFault" namespace=""/>
      </wsdl:operation>
      <wsdl:operation name="GetRasterPageItem">
        <soap12:operation
          soapAction="http://tempuri.org/IVisioGraphicsService/GetRasterPageItem" style="document"/>
        <wsdl:input>
          <wsp:PolicyReference
            URI="#CustomBinding_IVisioGraphicsService_GetRasterPageItem_Input_policy"/>
          <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <wsp:PolicyReference
            URI="#CustomBinding_IVisioGraphicsService_GetRasterPageItem_output_policy"/>
          <soap12:body use="literal"/>
        </wsdl:output>
        <wsdl:fault name="VisioGraphicsServiceFaultFault">
          <wsp:PolicyReference
            URI="#CustomBinding_IVisioGraphicsService_GetRasterPageItem_VisioGraphicsServiceFaultFault_Fault"/>
        </wsdl:fault>
        <soap12:fault use="literal" name="VisioGraphicsServiceFaultFault" namespace=""/>
      </wsdl:operation>
      <wsdl:operation name="SaveComments">
        <soap12:operation soapAction="http://tempuri.org/IVisioGraphicsService/SaveComments"
          style="document"/>
        <wsdl:input>
          <wsp:PolicyReference
            URI="#CustomBinding_IVisioGraphicsService_SaveComments_Input_policy"/>
          <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
          <wsp:PolicyReference
            URI="#CustomBinding_IVisioGraphicsService_SaveComments_output_policy"/>
          <soap12:body use="literal"/>
        </wsdl:output>

```

```

    <wsdl:fault name="VisioGraphicsServiceFaultFault">
      <wsp:PolicyReference
URI="#CustomBinding_IVisioGraphicsService_SaveComments_VisioGraphicsServiceFaultFault_Fault"/>
    >
      <soap12:fault use="literal" name="VisioGraphicsServiceFaultFault" namespace=""/>
    </wsdl:fault>
  </wsdl:operation>
</wsdl:binding>
<wsdl:binding name="Secure" type="tns2:IVisioGraphicsService">
  <wsp:PolicyReference URI="#Secure_policy"/>
  <soap12:binding transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="GetRasterDiagram">
    <soap12:operation
soapAction="http://tempuri.org/IVisioGraphicsService/GetRasterDiagram" style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
    <wsdl:fault name="VisioGraphicsServiceFaultFault">
      <soap12:fault use="literal" name="VisioGraphicsServiceFaultFault" namespace=""/>
    </wsdl:fault>
  </wsdl:operation>
  <wsdl:operation name="GetVectorDiagram">
    <soap12:operation
soapAction="http://tempuri.org/IVisioGraphicsService/GetVectorDiagram" style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
    <wsdl:fault name="VisioGraphicsServiceFaultFault">
      <soap12:fault use="literal" name="VisioGraphicsServiceFaultFault" namespace=""/>
    </wsdl:fault>
  </wsdl:operation>
  <wsdl:operation name="GetRasterPage">
    <soap12:operation soapAction="http://tempuri.org/IVisioGraphicsService/GetRasterPage"
style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
    <wsdl:fault name="VisioGraphicsServiceFaultFault">
      <soap12:fault use="literal" name="VisioGraphicsServiceFaultFault" namespace=""/>
    </wsdl:fault>
  </wsdl:operation>
  <wsdl:operation name="GetRasterPageItem">
    <soap12:operation
soapAction="http://tempuri.org/IVisioGraphicsService/GetRasterPageItem" style="document"/>
    <wsdl:input>
      <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>

```

```
<wsdl:fault name="VisioGraphicsServiceFaultFault">
  <soap12:fault use="literal" name="VisioGraphicsServiceFaultFault" namespace=""/>
</wsdl:fault>
</wsdl:operation>
<wsdl:operation name="SaveComments">
  <soap12:operation soapAction="http://tempuri.org/IVisioGraphicsService/SaveComments"
style="document"/>
  <wsdl:input>
    <soap12:body use="literal"/>
  </wsdl:input>
  <wsdl:output>
    <soap12:body use="literal"/>
  </wsdl:output>
  <wsdl:fault name="VisioGraphicsServiceFaultFault">
    <soap12:fault use="literal" name="VisioGraphicsServiceFaultFault" namespace=""/>
  </wsdl:fault>
</wsdl:operation>
</wsdl:binding>
</wsdl:definitions>
```

7 Appendix B: Full XML Schema

For ease of implementation, the following sections provide the full XML schema for this protocol.

Schema name	Prefix	Section
http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer	tns	7.1
http://schemas.microsoft.com/2003/10/Serialization/Arrays	q1	7.2
http://schemas.microsoft.com/2003/10/Serialization/	tns1	7.3
http://tempuri.org/	tns2	7.4

7.1

<http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer> Schema

```
<?xml version="1.0"?>
<xs:schema
  xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer"
  elementFormDefault="qualified"
  targetNamespace="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServer"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import
    schemaLocation="http://www.example.com/bfa9dba966bd4821ac2fca057487aaa8/VisioGraphicsService.svc?xsd=xsd1"
    namespace="http://schemas.microsoft.com/2003/10/Serialization/" />
  <xs:import
    schemaLocation="http://www.example.com/bfa9dba966bd4821ac2fca057487aaa8/VisioGraphicsService.svc?xsd=xsd4"
    namespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays/" />
  <xs:complexType name="RasterDiagramRequestContract">
    <xs:complexContent mixed="false">
      <xs:extension base="tns:BaseRequestContract">
        <xs:sequence>
          <xs:element minOccurs="0" name="Resolution" type="xs:int"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="RasterDiagramRequestContract" nillable="true" type="tns:RasterDiagramRequestContract"/>
  <xs:complexType name="BaseRequestContract">
    <xs:sequence>
      <xs:element minOccurs="0" name="DataSources" nillable="true" type="tns:ArrayOfAddonDataSource"/>
      <xs:element minOccurs="0" name="DisableRefresh" type="xs:boolean"/>
      <xs:element minOccurs="0" name="FileUri" nillable="true" type="xs:anyURI"/>
      <xs:element minOccurs="0" name="PageNumber" type="xs:int"/>
      <xs:element minOccurs="0" name="TimeLastModified" type="xs:dateTime"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="BaseRequestContract" nillable="true" type="tns:BaseRequestContract"/>
  <xs:complexType name="ArrayOfAddonDataSource">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="unbounded" name="AddonDataSource" nillable="true" type="tns:AddonDataSource"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>
```

```

    </xs:sequence>
  </xs:complexType>
  <xs:element name="ArrayOfAddonDataSource" nillable="true"
type="tns:ArrayOfAddonDataSource"/>
  <xs:complexType name="AddonDataSource">
    <xs:sequence>
      <xs:element minOccurs="0" name="ConnectionString" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Data" nillable="true">
        <xs:complexType>
          <xs:sequence>
            <xs:element ref="xs:schema"/>
            <xs:any/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element minOccurs="0" name="Id" type="xs:int"/>
      <xs:element minOccurs="0" name="QueryString" nillable="true" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="AddonDataSource" nillable="true" type="tns:AddonDataSource"/>
  <xs:complexType name="RasterDiagram">
    <xs:complexContent mixed="false">
      <xs:extension base="tns:DiagramBase">
        <xs:sequence>
          <xs:element minOccurs="0" name="AreaMaps" nillable="true" type="xs:string"/>
          <xs:element minOccurs="0" name="Image" nillable="true" type="xs:base64Binary"/>
          <xs:element minOccurs="0" name="ShapeInfo" nillable="true" type="xs:string"/>
        </xs:sequence>
      </xs:extension>
    </xs:complexContent>
  </xs:complexType>
  <xs:element name="RasterDiagram" nillable="true" type="tns:RasterDiagram"/>
  <xs:complexType name="DiagramBase">
    <xs:sequence>
      <xs:element minOccurs="0" name="CacheAge" type="xs:int"/>
      <xs:element minOccurs="0" name="DataSources" nillable="true"
type="tns:ArrayOfAddonDataSource"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="DiagramBase" nillable="true" type="tns:DiagramBase"/>
  <xs:complexType name="VisioGraphicsServiceFault">
    <xs:sequence>
      <xs:element minOccurs="0" name="Details" nillable="true" type="xs:string"/>
      <xs:element minOccurs="0" name="Error" type="tns:FaultCode"/>
      <xs:element minOccurs="0" name="Flags" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="VisioGraphicsServiceFault" nillable="true"
type="tns:VisioGraphicsServiceFault"/>
  <xs:simpleType name="FaultCode">
    <xs:restriction base="xs:string">
      <xs:enumeration value="DataConnectivityError"/>
      <xs:enumeration value="DataSourceConnectionError"/>
      <xs:enumeration value="DataSourcePermissionError"/>
      <xs:enumeration value="EmptyRasterDiagramRequest"/>
      <xs:enumeration value="EmptyVectorDiagramRequest"/>
      <xs:enumeration value="OfficeDataConnectionFileNotFound"/>
      <xs:enumeration value="OfficeDataConnectionFileRetrievalError"/>
      <xs:enumeration value="OfficeDataConnectionFileAccessDenied"/>
    </xs:restriction>
  </xs:simpleType>

```

```

<xs:enumeration value="IncompatibleDomain"/>
<xs:enumeration value="WssViewAccessDenied"/>
<xs:enumeration value="WssListAccessDenied"/>
<xs:enumeration value="InvalidPageNumber"/>
<xs:enumeration value="InvalidResolution"/>
<xs:enumeration value="InvalidFileUrl"/>
<xs:enumeration value="RasterizationError"/>
<xs:enumeration value="InvalidFileExtension"/>
<xs:enumeration value="EmptyServerData"/>
<xs:enumeration value="OpenServerFileFailed"/>
<xs:enumeration value="NoReadPermissionToFile"/>
<xs:enumeration value="KeyNotFoundFromCollection"/>
<xs:enumeration value="ExceededMaxDiagramSize"/>
<xs:enumeration value="IncompatibleVersion"/>
<xs:enumeration value="DataProviderNotTrusted"/>
<xs:enumeration value="DataBindingConflict"/>
<xs:enumeration value="UnableToGetExternalDataSourceCredentials"/>
<xs:enumeration value="UnattendedAccountTypeMismatch"/>
<xs:enumeration value="RecalcTimedOut"/>
<xs:enumeration value="LicenseExpired"/>
<xs:enumeration value="VSDXFileLoadGenericError"/>
<xs:enumeration value="VSDXFileLoadPermissionError"/>
<xs:enumeration value="VSDXFileSizeError"/>
<xs:enumeration value="WebPartConfigError"/>
</xs:restriction>
</xs:simpleType>
<xs:element name="FaultCode" nillable="true" type="tns:FaultCode"/>
<xs:complexType name="VectorDiagramRequestContract">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:BaseRequestContract">
      <xs:sequence/>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:element name="VectorDiagramRequestContract" nillable="true"
type="tns:VectorDiagramRequestContract"/>
<xs:complexType name="VectorDiagram">
  <xs:complexContent mixed="false">
    <xs:extension base="tns:DiagramBase">
      <xs:sequence>
        <xs:element minOccurs="0" name="OutputPage" nillable="true"
type="tns:RenderedPage"/>
      </xs:sequence>
    </xs:extension>
  </xs:complexContent>
</xs:complexType>
<xs:element name="VectorDiagram" nillable="true" type="tns:VectorDiagram"/>
<xs:complexType name="RenderedPage">
  <xs:sequence>
    <xs:element minOccurs="0" name="Resources" nillable="true"
type="tns:ArrayOfDiagramResource"/>
    <xs:element minOccurs="0" name="ShapeInfo" nillable="true" type="xs:string"/>
    <xs:element minOccurs="0" name="Xaml" nillable="true" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="RenderedPage" nillable="true" type="tns:RenderedPage"/>
<xs:complexType name="ArrayOfDiagramResource">
  <xs:sequence>

```



```

        <xs:element minOccurs="0" maxOccurs="unbounded" name="DiagramResource" nillable="true"
type="tns:DiagramResource"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="ArrayOfDiagramResource" nillable="true"
type="tns:ArrayOfDiagramResource"/>
<xs:complexType name="DiagramResource">
    <xs:sequence>
        <xs:element minOccurs="0" name="ByteData" nillable="true" type="xs:base64Binary"/>
        <xs:element minOccurs="0" name="Name" nillable="true" type="xs:string"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="DiagramResource" nillable="true" type="tns:DiagramResource"/>
<xs:complexType name="RasterPageRequest">
    <xs:sequence>
        <xs:element minOccurs="0" name="DataModeName" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="DataSources" nillable="true"
type="tns:ArrayOfAddonDataSource"/>
        <xs:element minOccurs="0" name="EntityTag" nillable="true" type="tns:RasterPageTag"/>
        <xs:element minOccurs="0" name="FileUrl" nillable="true" type="xs:anyURI"/>
        <xs:element minOccurs="0" name="PageId" type="xs:long"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="RasterPageRequest" nillable="true" type="tns:RasterPageRequest"/>
<xs:complexType name="RasterPageTag">
    <xs:sequence>
        <xs:element minOccurs="0" name="DataModeName" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="FileETag" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="OutputETag" nillable="true" type="xs:string"/>
        <xs:element minOccurs="0" name="SequenceId" nillable="true" type="xs:string"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="RasterPageTag" nillable="true" type="tns:RasterPageTag"/>
<xs:complexType name="RasterPageResponse">
    <xs:sequence>
        <xs:element minOccurs="0" name="DataSources" nillable="true"
type="tns:ArrayOfAddonDataSource"/>
        <xs:element minOccurs="0" name="EntityTag" nillable="true" type="tns:RasterPageTag"/>
        <xs:element minOccurs="0" name="PageInfo" nillable="true" type="tns:RasterPageInfo"/>
        <xs:element minOccurs="0" name="ResponseCode" type="tns:RasterPageResponseCode"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="RasterPageResponse" nillable="true" type="tns:RasterPageResponse"/>
<xs:complexType name="RasterPageInfo">
    <xs:sequence>
        <xs:element minOccurs="0" name="DefaultImage" type="xs:int"/>
        <xs:element xmlns:q1="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
minOccurs="0" name="ZoomValues" nillable="true" type="q1:ArrayOfint"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="RasterPageInfo" nillable="true" type="tns:RasterPageInfo"/>
<xs:simpleType name="RasterPageResponseCode">
    <xs:restriction base="xs:string">
        <xs:enumeration value="NotModified"/>
        <xs:enumeration value="OK"/>
        <xs:enumeration value="DataSources"/>
    </xs:restriction>
</xs:simpleType>

```

```

    <xs:element name="RasterPageResponseCode" nillable="true"
type="tns:RasterPageResponseCode"/>
    <xs:complexType name="RasterPageItemRequest">
        <xs:sequence>
            <xs:element minOccurs="0" name="EntityTag" nillable="true"
type="tns:RasterPageItemTag"/>
            <xs:element minOccurs="0" name="FileUrl" nillable="true" type="xs:anyURI"/>
            <xs:element minOccurs="0" name="ItemName" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="SequenceId" nillable="true" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="RasterPageItemRequest" nillable="true" type="tns:RasterPageItemRequest"/>
    <xs:complexType name="RasterPageItemTag">
        <xs:sequence>
            <xs:element minOccurs="0" name="DataModeName" nillable="true" type="xs:string"/>
            <xs:element minOccurs="0" name="FileETag" nillable="true" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="RasterPageItemTag" nillable="true" type="tns:RasterPageItemTag"/>
    <xs:complexType name="RasterPageItemResponse">
        <xs:sequence>
            <xs:element minOccurs="0" name="EntityTag" nillable="true"
type="tns:RasterPageItemTag"/>
            <xs:element minOccurs="0" name="ItemData" nillable="true" type="xs:base64Binary"/>
            <xs:element minOccurs="0" name="ResponseCode" type="tns:RasterPageItemResponseCode"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="RasterPageItemResponse" nillable="true"
type="tns:RasterPageItemResponse"/>
    <xs:simpleType name="RasterPageItemResponseCode">
        <xs:restriction base="xs:string">
            <xs:enumeration value="NotModified"/>
            <xs:enumeration value="OK"/>
        </xs:restriction>
    </xs:simpleType>
    <xs:element name="RasterPageItemResponseCode" nillable="true"
type="tns:RasterPageItemResponseCode"/>
</xs:schema>

```

7.2 <http://schemas.microsoft.com/2003/10/Serialization/Arrays> Schema

```

<?xml version="1.0"?>
<xs:schema xmlns:q1="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/2003/10/Serialization/Arrays"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:complexType name="ArrayOfint">
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="unbounded" name="int" type="xs:int"/>
        </xs:sequence>
    </xs:complexType>
    <xs:element name="ArrayOfint" nillable="true" type="q1:ArrayOfint"/>
</xs:schema>

```

7.3 <http://schemas.microsoft.com/2003/10/Serialization/> Schema

```

<?xml version="1.0"?>

```

```

<xs:schema xmlns:tns1="http://schemas.microsoft.com/2003/10/Serialization/"
attributeFormDefault="qualified" elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/2003/10/Serialization/"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="anyType" nillable="true" type="xs:anyType"/>
  <xs:element name="anyURI" nillable="true" type="xs:anyURI"/>
  <xs:element name="base64Binary" nillable="true" type="xs:base64Binary"/>
  <xs:element name="boolean" nillable="true" type="xs:boolean"/>
  <xs:element name="byte" nillable="true" type="xs:byte"/>
  <xs:element name="dateTime" nillable="true" type="xs:dateTime"/>
  <xs:element name="decimal" nillable="true" type="xs:decimal"/>
  <xs:element name="double" nillable="true" type="xs:double"/>
  <xs:element name="float" nillable="true" type="xs:float"/>
  <xs:element name="int" nillable="true" type="xs:int"/>
  <xs:element name="long" nillable="true" type="xs:long"/>
  <xs:element name="QName" nillable="true" type="xs:QName"/>
  <xs:element name="short" nillable="true" type="xs:short"/>
  <xs:element name="string" nillable="true" type="xs:string"/>
  <xs:element name="unsignedByte" nillable="true" type="xs:unsignedByte"/>
  <xs:element name="unsignedInt" nillable="true" type="xs:unsignedInt"/>
  <xs:element name="unsignedLong" nillable="true" type="xs:unsignedLong"/>
  <xs:element name="unsignedShort" nillable="true" type="xs:unsignedShort"/>
  <xs:element name="char" nillable="true" type="tns1:char"/>
  <xs:simpleType name="char">
    <xs:restriction base="xs:int"/>
  </xs:simpleType>
  <xs:element name="duration" nillable="true" type="tns1:duration"/>
  <xs:simpleType name="duration">
    <xs:restriction base="xs:duration">
      <xs:pattern value="\-?P(\d*D)?(T(\d*H)?(\d*M)?(\d*(\.\d*)?S)?)?"/>
      <xs:minInclusive value="-P10675199DT2H48M5.4775808S"/>
      <xs:maxInclusive value="P10675199DT2H48M5.4775807S"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="guid" nillable="true" type="tns1:guid"/>
  <xs:simpleType name="guid">
    <xs:restriction base="xs:string">
      <xs:pattern value="[\da-fA-F]{8}-[\da-fA-F]{4}-[\da-fA-F]{4}-[\da-fA-F]{4}-[\da-fA-F]{12}"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:attribute name="FactoryType" type="xs:QName"/>
  <xs:attribute name="Id" type="xs:ID"/>
  <xs:attribute name="Ref" type="xs:IDREF"/>
</xs:schema>

```

7.4 http://tempuri.org/ Schema

```

<?xml version="1.0"?>
<xs:schema elementFormDefault="qualified" targetNamespace="http://tempuri.org/"
xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:import
schemaLocation="http://www.example.com/bfa9dba966bd4821ac2fca057487aaa8/VisioGraphicsService.
svc?xsd=xsd2"
namespace="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er"/>
  <xs:element name="GetRasterDiagram">
    <xs:complexType>

```

```

    <xs:sequence>
      <xs:element
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" minOccurs="0" name="RasterDiagramRequestContract" nillable="true"
type="tns:RasterDiagramRequestContract"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="GetRasterDiagramResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" minOccurs="0" name="GetRasterDiagramResult" nillable="true" type="tns:RasterDiagram"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="GetVectorDiagram">
  <xs:complexType>
    <xs:sequence>
      <xs:element
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" minOccurs="0" name="vectorDiagramRequest" nillable="true"
type="tns:VectorDiagramRequestContract"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="GetVectorDiagramResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" minOccurs="0" name="GetVectorDiagramResult" nillable="true" type="tns:VectorDiagram"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="GetRasterPage">
  <xs:complexType>
    <xs:sequence>
      <xs:element
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" minOccurs="0" name="RasterPageRequestContract" nillable="true"
type="tns:RasterPageRequest"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="GetRasterPageResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" minOccurs="0" name="GetRasterPageResult" nillable="true" type="tns:RasterPageResponse"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="GetRasterPageItem">
  <xs:complexType>
    <xs:sequence>
      <xs:element
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ

```

```

er" minOccurs="0" name="rasterPageItemRequestContract" nillable="true"
type="tns:RasterPageItemRequest"/>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="GetRasterPageItemResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element
xmlns:tns="http://schemas.datacontract.org/2004/07/Microsoft.Office.Visio.Server.GraphicsServ
er" minOccurs="0" name="GetRasterPageItemResult" nillable="true"
type="tns:RasterPageItemResponse"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="SaveComments">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="fileUrl" nillable="true" type="xs:anyURI"/>
        <xs:element minOccurs="0" name="commentData" nillable="true" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="SaveCommentsResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" name="SaveCommentsResult" nillable="true"
type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
</xs:schema>

```

8 Appendix C: 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® SharePoint® Server 2010
- Microsoft® SharePoint® Server 15 Technical Preview

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

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

[<1> Section 2.2.9.1:](#) This element and its complex types are available only in SharePoint Server 15 Technical Preview.

[<2> Section 2.2.9.2:](#) This element and its complex types are available only in SharePoint Server 15 Technical Preview.

[<3> Section 2.2.9.3:](#) This element and its complex types are available only in SharePoint Server 15 Technical Preview.

[<4> Section 2.2.9.4:](#) This element and its complex types are available only in SharePoint Server 15 Technical Preview.

[<5> Section 3.1.1:](#) This model is available only in SharePoint Server 15 Technical Preview.

[<6> Section 3.1.2:](#) This timer is available only in SharePoint Server 15 Technical Preview.

[<7> Section 3.1.4:](#) This operation is available only in SharePoint Server 15 Technical Preview.

[<8> Section 3.1.4:](#) This operation is available only in SharePoint Server 15 Technical Preview.

[<9> Section 3.1.4:](#) This operation is available only in SharePoint Server 15 Technical Preview.

[<10> Section 3.1.4.3:](#) This operation is available only in SharePoint Server 15 Technical Preview.

[<11> Section 3.1.4.4:](#) This operation is available only in SharePoint Server 15 Technical Preview.

[<12> Section 3.1.4.5:](#) This operation is available only in SharePoint Server 15 Technical Preview.

[<13> Section 3.1.5:](#) This timer event is available only in SharePoint Server 15 Technical Preview.

9 Change Tracking

This section identifies changes that were made to the [MS-VGSP] protocol document between the June 2011 and January 2012 releases. Changes are classified as New, Major, Minor, Editorial, or No change.

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

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

- A document revision that incorporates changes to interoperability requirements or functionality.
- An extensive rewrite, addition, or deletion of major portions of content.
- The removal of a document from the documentation set.
- Changes made for template compliance.

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

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

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

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

- New content added.
- Content updated.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.

- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.
- New content added for template compliance.
- Content updated for template compliance.
- Content removed for template compliance.
- Obsolete document removed.

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

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

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.
- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

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

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
1.1 Glossary	Added the term "drawing".	Y	New content added.
1.1 Glossary	Added the term "embedded object".	Y	New content added.
1.1 Glossary	Added the term "hyperlink".	Y	New content added.
1.1 Glossary	Added the term "WSDL message".	Y	New content added.
1.1 Glossary	Added the term "XML namespace".	Y	New content added.
1.1 Glossary	Updated SOAP version.	N	Content updated.
1.1 Glossary	Added the term "XML namespace prefix".	Y	New content added.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
1.1 Glossary	Added the term "zoom level".	Y	New content added.
1.1 Glossary	Added the term "pixel".	Y	New content added.
1.2 References	Revised.	N	Content updated for template compliance.
1.2.1 Normative References	Added a normative reference to Visio Graphics Service VSDX File Format [MS-VSDX].	Y	New content added.
1.2.1 Normative References	Added a normative reference to The application/json Media Type for JavaScript Object Notation [RFC4627].	Y	New content added.
1.2.1 Normative References	Removed normative reference to Web Services Addressing 1.0 - Core [WSA1.0 Core].	Y	Obsolete document removed.
1.2.1 Normative References	Removed normative reference to Web Services Addressing 1.0 - Metadata [WSA1.0 Metadata].	Y	Obsolete document removed.
1.2.1 Normative References	Removed normative reference to Web Services Addressing 1.0 - WSDL Binding [WSA1.0].	Y	Obsolete document removed.
1.2.1 Normative References	Removed normative reference to Web Services Addressing (WS-Addressing)	Y	Obsolete document

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
	[WSAddressing].		removed.
1.2.1 Normative References	Removed normative reference to Web Services Metadata Exchange (WS-MetadataExchange) [WS-MetadataExchange].	Y	Obsolete document removed.
1.2.1 Normative References	Removed normative reference to Web Services Policy 1.5 - Framework [WS-Policy1.5].	Y	Obsolete document removed.
1.2.1 Normative References	Removed normative reference to Web Services Security: SOAP Message Security 1.0 (WS-Security 2004) [WSSE 1.0].	Y	Obsolete document removed.
1.2.2 Informative References	Added an informative reference to GZIP file format specification version 4.3 [RFC1952].	Y	New content added.
1.2.2 Informative References	Added an informative reference to UTF-8, A Transformation Format of ISO 10646 [RFC3629].	Y	New content added.
1.3 Protocol Overview (Synopsis)	Provided description of new comment functionality.	Y	Content updated.
1.3.1 Web Drawing	Updated description of a web drawing by including a distinction between the two file formats.	Y	Content updated.
1.3.2 Drawing Page	Updated description to include new file format.	Y	Content updated.
1.3.2.1 Foreground Page	Added description of a foreground page in	Y	New content

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
	the VSDX web drawings.		added due to protocol revision.
1.3.3 Shape	Clarified the definition of shape by including formatting and shape data.	Y	Content updated.
1.3.3 Shape	Updated description to include new file format.	Y	Content updated.
1.3.4 Shape Data	Added description of shape data and references to the file format specifications of this element.	Y	New content added.
1.3.5 Comment	Added description of comments and references to the file format specifications of this element.	Y	New content added.
1.6 Applicability Statement	Documented new comment functionality.	Y	Content updated.
2.1 Transport	Revised description of transport.	N	Editorially updated.
2.2.1 Namespaces	Updated list of namespaces used in the protocol.	Y	Content updated.
2.2.4 Complex Types	Modified the descriptions of complex types.	Y	Content updated due to protocol revision.
2.2.4.1 AddonDataSource	Modified; added namespace and removed the annotation DataSet.	Y	Protocol syntax updated due to protocol revision.
2.2.4.2 ArrayOfAddonDataSource	Modified; added namespace.	Y	Protocol syntax

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
			updated due to protocol revision.
2.2.4.3 BaseRequestContract	Clarified behavior of complex type only applies to VGSFF.	N	Content updated .
2.2.4.3 BaseRequestContract	Modified; added namespace.	Y	Protocol syntax updated due to protocol revision.
2.2.4.4 DiagramBase	Clarified behavior of complex type only applies to VGSFF.	N	Content updated .
2.2.4.4 DiagramBase	Modified; added namespace.	Y	Protocol syntax updated due to protocol revision.
2.2.4.5 VisioGraphicsServiceFault	Clarified behavior of flags only applies to VGSFF.	N	Content updated .
2.2.4.5 VisioGraphicsServiceFault	Modified; added namespace.	Y	Protocol syntax updated due to protocol revision.
2.2.5.1 FaultCode	Added new fault codes used in new operations: DataSourceConnectionError, DataSourcePermissionError, VSDXFileLoadGenericError, VSDXFileLoadPermissionError, VSDXFileSizeError, WebPartConfigError.	Y	Protocol syntax updated due to protocol revision.
2.2.9 Common Data Structures	Added description of data structures used	Y	New content

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
	in VSDX web drawings.		added due to protocol revision.
2.2.9.1 ShapeInfo	Added description of ShapeInfo XML fragment used in VSDX web drawings.	Y	New content added due to protocol revision.
2.2.9.1.1 Page	Added description of Page complex type used in VSDX web drawings.	Y	New protocol syntax added due to protocol revision.
2.2.9.1.2 Pages	Added description of Pages complex type used in VSDX web drawings.	Y	Protocol syntax updated due to protocol revision.
2.2.9.1.3 PageInfo	Added description of PageInfo complex type used in VSDX web drawings.	Y	New protocol syntax added due to protocol revision.
2.2.9.1.4 ShapeInfo	Added description of ShapeInfo complex type used in VSDX web drawings.	Y	New protocol syntax added due to protocol revision.
2.2.9.1.5 Hyperlinks	Added description of Hyperlinks complex type used in VSDX web drawings.	Y	New content added due to protocol revision.
2.2.9.1.6 Hyperlink	Added description of Hyperlink complex type used in VSDX web drawings.	Y	New content added due to

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
			protocol revision.
2.2.9.1.7 ShapeDataItems	Added description of ShapeDataItems complex type used in VSDX web drawings.	Y	New content added due to protocol revision.
2.2.9.1.8 ShapeData	Added description of ShapeData complex type used in VSDX web drawings.	Y	New content added due to protocol revision.
2.2.9.1.9 Shapes	Added description of Shapes complex type used in VSDX web drawings.	Y	New content added due to protocol revision.
2.2.9.1.10 Shape	Added description of Shape complex type used in VSDX web drawings.	Y	New content added due to protocol revision.
2.2.9.1.11 Path	Added description of Path complex type used in VSDX web drawings.	Y	New content added due to protocol revision.
2.2.9.2 Comments	Added description of Comments XML fragment used in VSDX web drawings.	Y	New content added due to protocol revision.
2.2.9.2.1 Comments	Added description of Comments complex type used in VSDX web drawings.	Y	New protocol syntax added due to protocol revision.
2.2.9.2.2	Added description of	Y	New

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
CommentEntry	CommentEntry complex type used in VSDX web drawings.		protocol syntax added due to protocol revision.
2.2.9.3 CommentData	Added description of CommentData XML fragment used in VSDX web drawings.	Y	New content added due to protocol revision.
2.2.9.3.1 Comment	Added description of Comment complex type used in VSDX web drawings.	Y	New protocol syntax added due to protocol revision.
2.2.9.4 CommentResult	Added description of CommentResult XML fragment used in VSDX web drawings.	Y	New content added due to protocol revision.
2.2.9.4.1 Comment	Added description of Comment complex type used in VSDX web drawings.	Y	New protocol syntax added due to protocol revision.
3 Protocol Details	Removed client description that does not apply anymore.	Y	Content removed.
3.1 Common Details	Updated description for new protocol operations.	Y	Content updated due to protocol revision.
3.1.1 Abstract Data Model	Added description of data model for VSDX web drawings.	Y	Content updated due to protocol revision.
3.1.2	Added description of	Y	Content

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
Timers	new timers used in new operations, GetRasterPage and GetRasterPageItem.		updated due to protocol revision.
3.1.4 Message Processing Events and Sequencing Rules	Modified: added description of message sequence and restrictions used in new operations: GetRasterPage, GetRasterPageItem and SaveComments.	Y	Content updated due to protocol revision.
3.1.4.1 GetRasterDiagram	Modified: added namespaces and revised prefixes, while clarifying operation only applies to VGSFF.	Y	Protocol syntax updated due to protocol revision.
3.1.4.1.1 Messages	Added table with message descriptions.	N	New content added for template compliance.
3.1.4.1.1.2 IVisioGraphicsService GetRasterDiagram OutputMessage	Removed the SOAP action value.	Y	Protocol syntax removed due to protocol revision.
3.1.4.1.1.3 IVisioGraphicsService GetRasterDiagram VisioGraphicsService FaultFault FaultMessage	Removed the SOAP action value.	Y	Protocol syntax removed due to protocol revision.
3.1.4.1.2 Elements	Added table with element descriptions.	N	New content added for template compliance.
3.1.4.1.2.1	Modified: added	Y	Protocol

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
GetRasterDiagram	namespace and revised element.		syntax updated due to protocol revision.
3.1.4.1.2.2 GetRasterDiagramResponse	Modified: added namespace and revised element.	Y	Protocol syntax updated due to protocol revision.
3.1.4.1.3 Complex Types	Added table with complex type descriptions.	N	New content added for template compliance.
3.1.4.1.3.1 RasterDiagramRequestContract	Modified: added namespace, while clarifying requirement of resolution element.	Y	Protocol syntax updated due to protocol revision.
3.1.4.1.3.2 RasterDiagram	Modified: added namespace, while clarifying elements of ShapeInfo only apply to VGSFF.	Y	Protocol syntax updated due to protocol revision.
3.1.4.2 GetVectorDiagram	Modified: added namespaces and revised prefixes, while clarifying operation only applies to VGSFF.	Y	Protocol syntax updated due to protocol revision.
3.1.4.2.1 Messages	Added table with message descriptions.	N	New content added for template compliance.
3.1.4.2.1.2 IVisioGraphicsService_GetVectorDiagram_OutputMessage	Removed the SOAP action value.	Y	Protocol syntax

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
			removed due to protocol revision.
3.1.4.2.1.3 IVisioGraphicsService_GetVectorDiagram VisioGraphicsServiceFaultFault FaultMessage	Removed the SOAP action value.	Y	Protocol syntax removed due to protocol revision.
3.1.4.2.2 Elements	Added table with element descriptions.	N	New content added for template compliance.
3.1.4.2.2.1 GetVectorDiagram	Modified: added namespace and revised prefix.	Y	Protocol syntax updated due to protocol revision.
3.1.4.2.2.2 GetVectorDiagramResponse	Modified: added namespace and revised prefix.	Y	Protocol syntax updated due to protocol revision.
3.1.4.2.3 Complex Types	Added table with complex type descriptions.	N	New content added for template compliance.
3.1.4.2.3.1 VectorDiagramRequestContract	Modified: added namespace.	Y	Protocol syntax updated due to protocol revision.
3.1.4.2.3.2 VectorDiagram	Modified: added namespace.	Y	Protocol syntax updated

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
			due to protocol revision.
3.1.4.2.3.3 RenderedPage	Clarified elements of ShapeInfo only apply to VGSFF.	N	Content updated.
3.1.4.2.3.3 RenderedPage	Modified: added namespace.	Y	Protocol syntax updated due to protocol revision.
3.1.4.2.3.4 ArrayOfDiagramResource	Modified: added namespace.	Y	Protocol syntax updated due to protocol revision.
3.1.4.2.3.5 DiagramResource	Modified: added namespace.	Y	Protocol syntax updated due to protocol revision.
3.1.4.3 GetRasterPage	Added new operation GetRasterPage.	Y	New protocol syntax added due to protocol revision.
3.1.4.3.1 Messages	Added summary table of messages for the GetRasterPage operation.	Y	New content added due to protocol revision.
3.1.4.3.1.1 IVisioGraphicsService_GetRasterPage_InputMessage	Added new message for the GetRasterPage operation.	Y	New protocol syntax added due to protocol revision.
3.1.4.3.1.2 IVisioGraphicsService_GetRasterPage_OutputMessage	Added new message for the GetRasterPage operation.	Y	New protocol

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
	operation.		syntax added due to protocol revision.
3.1.4.3.1.3 IVisioGraphicsService_GetRasterPage_VisioGraphicsServiceFaultFault_FaultMessage	Added new message for the GetRasterPage operation.	Y	New protocol syntax added due to protocol revision.
3.1.4.3.2 Elements	Added summary table of elements for the GetRasterPage operation.	Y	New content added due to protocol revision.
3.1.4.3.2.1 GetRasterPage	Added new element for the GetRasterPage operation.	Y	New protocol syntax added due to protocol revision.
3.1.4.3.2.2 GetRasterPageResponse	Added new element for the GetRasterPage operation.	Y	New protocol syntax added due to protocol revision.
3.1.4.3.3 Complex Types	Added summary table of complex types for the GetRasterPage operation.	Y	New content added due to protocol revision.
3.1.4.3.3.1 RasterPageRequest	Added new complex type for the GetRasterPage operation.	Y	New protocol syntax added due to protocol revision.
3.1.4.3.3.2 RasterPageResponse	Added new complex type for the	Y	New protocol

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
	GetRasterPage operation.		syntax added due to protocol revision.
3.1.4.3.3.3 RasterPageTag	Added new complex type for the GetRasterPage operation.	Y	New protocol syntax added due to protocol revision.
3.1.4.3.3.4 RasterPageInfo	Added new complex type for the GetRasterPage operation.	Y	New protocol syntax added due to protocol revision.
3.1.4.3.4 Simple Types	Added summary table of simple types for the GetRasterPage operation.	Y	New content added due to protocol revision.
3.1.4.3.4.1 RasterPageResponseCode	Added new simple type for the GetRasterPage operation.	Y	New protocol syntax added due to protocol revision.
3.1.4.4 GetRasterPageItem	Added description for GetRasterPageItem operation.	Y	New content added.
3.1.4.4.1 Messages	Added description for messages of GetRasterPageItem operation.	Y	New content added.
3.1.4.4.1.1 IVisioGraphicsService GetRasterPageItem InputMessage	Added new message for the GetRasterPageItem operation.	Y	New content added.
3.1.4.4.1.2 IVisioGraphicsService GetRasterPageItem OutputMessage	Added new message for the GetRasterPageItem operation.	Y	New content added.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
	operation.		
3.1.4.4.1.3 IVisioGraphicsService_GetRasterPageItem_VisioGraphicsServiceFaultFault_FaultMessage	Added new message for the GetRasterPageItem operation.	Y	New content added.
3.1.4.4.2 Elements	Added description for elements of GetRasterPageItem operation.	Y	New content added.
3.1.4.4.2.1 GetRasterPageItem	Added description of GetRasterPageItem element.	Y	New content added.
3.1.4.4.2.2 GetRasterPageItemResponse	Added description of GetRasterPageItemResponse element.	Y	New content added.
3.1.4.4.3 Complex Types	Added description for complex types of GetRasterPageItem operation.	Y	New content added.
3.1.4.4.3.1 RasterPageItemRequest	Added description for RasterPageItemRequest complex type.	Y	New content added.
3.1.4.4.3.2 RasterPageItemTag	Added description for RasterPageItemTag complex type.	Y	New content added.
3.1.4.4.3.3 RasterPageItemResponse	Added description for RasterPageItemResponse complex type.	Y	New content added.
3.1.4.4.4 Simple Types	Added description for simple types of GetRasterPageItem operation.	Y	New content added.
3.1.4.4.4.1 RasterPageItemResponseCode	Added description for RasterPageItemResponseCode simple type.	Y	New content added.
3.1.4.5 SaveComments	Added description for SaveComments operation.	Y	New content added.
3.1.4.5.1 Messages	Added description for messages of SaveComments operation.	Y	New content added.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
3.1.4.5.1.1 IVisioGraphicsService SaveComments InputMessage	Added new message for the SaveComments operation.	Y	New content added.
3.1.4.5.1.2 IVisioGraphicsService SaveComments OutputMessage	Added new message for the SaveComments operation.	Y	New content added.
3.1.4.5.1.3 IVisioGraphicsService SaveComments VisioGraphicsServiceFault FaultMessage	Added new message for the SaveComments operation.	Y	New content added.
3.1.4.5.2 Elements	Added description for elements of SaveComments operation.	Y	New content added.
3.1.4.5.2.1 SaveComments	Added description of SaveComments element.	Y	New content added.
3.1.4.5.2.2 SaveCommentsResponse	Added description of SaveCommentsResponse element.	Y	New content added.
3.1.5 Timer Events	Added description of timer events.	Y	New content added.
4.1 GetRasterDiagram	Updated example code with correct rasterDiagramRequest Contract tag.	Y	New content added.
4.2 GetVectorDiagram	Updated example code with correct vectorDiagramRequestContract tag.	Y	New content added.
4.3 GetVectorDiagram with AddonDataSource	Updated example code with correct vectorDiagramRequestContract tag.	Y	New content added.
6 Appendix A: Full WSDL	Updated section with WSDL for updated protocol.	Y	New content added.
7 Appendix B: Full XML Schema	Updated section with Full XML schemas for updated protocol.	Y	New content added.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
8 Appendix C: Product Behavior	Added new product version.	Y	New content added.
	Added the sections Complex Types, Simple Types, Attributes and Groups for each of the operations.	N	New content added for template compliance.

10 Index

A

[AddonDataSource complex type](#) 14
[Applicability](#) 11
[ArrayOfAddonDataSource complex type](#) 14
[Attribute groups](#) 20
[Attributes](#) 19

B

[BaseRequestContract complex type](#) 15

C

[Capability negotiation](#) 11
[Change tracking](#) 111
Client
 [overview](#) 32
[Complex types](#) 13
 [AddonDataSource](#) 14
 [ArrayOfAddonDataSource](#) 14
 [BaseRequestContract](#) 15
 [DiagramBase](#) 16
 [VisioGraphicsServiceFault](#) 16

D

[DiagramBase complex type](#) 16
[Drawing page](#) 10

E

Examples
 [GetRasterDiagram](#) 63
 [GetVectorDiagram](#) 69
 [GetVectorDiagram with AddonDataSource](#) 71
 [overview](#) 63

F

[FaultCode simple type](#) 17
[Fields - vendor-extensible](#) 11
[Full WSDL](#) 89

G

[GetRasterDiagram example](#) 63
[GetVectorDiagram example](#) 69
[GetVectorDiagram with AddonDataSource example](#) 71
[Glossary](#) 7
[Groups](#) 19

I

[Implementer - security considerations](#) 88
[Index of security parameters](#) 88
[Informative references](#) 9

[Introduction](#) 7

M

Messages
 [AddonDataSource complex type](#) 14
 [ArrayOfAddonDataSource complex type](#) 14
 [attribute groups](#) 20
 [attributes](#) 19
 [BaseRequestContract complex type](#) 15
 [complex types](#) 13
 [DiagramBase complex type](#) 16
 [elements](#) 13
 [enumerated](#) 13
 [FaultCode simple type](#) 17
 [groups](#) 19
 [namespaces](#) 12
 [simple types](#) 17
 [syntax](#) 12
 [transport](#) 12
 [VisioGraphicsServiceFault complex type](#) 16

N

[Namespaces](#) 12
[Normative references](#) 8

O

Operations
 [ShapeInfo](#) 28
 [Overview \(synopsis\)](#) 9
 [drawing page](#) 10
 [shape](#) 10
 [Web drawing](#) 9

P

[Parameters - security index](#) 88
[Preconditions](#) 11
[Prerequisites](#) 11
[Product behavior](#) 110

R

[References](#) 8
 [informative](#) 9
 [normative](#) 8
[Relationship to other protocols](#) 10

S

Security
 [implementer considerations](#) 88
 [parameter index](#) 88
Server
 [overview](#) 32
 [ShapeInfo operation](#) 28

[Shape](#) 10
[Shape data](#) 10
[Simple types](#) 17
 [FaultCode](#) 17
[Standards assignments](#) 11
Structures
 [shape data](#) 10
Syntax
 [messages - overview](#) 12

T

[Tracking changes](#) 111
[Transport](#) 12
Types
 [complex](#) 13
 [simple](#) 17

V

[Vendor-extensible fields](#) 11
[Versioning](#) 11
[VisioGraphicsServiceFault complex type](#) 16

W

[Web drawing](#) 9
[WSDL](#) 89