

[MS-ESP]: Excel Web Services Protocol

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. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.
- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

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

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

Revision Summary

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

| Date | Revision History | Revision Class | Comments |
|-------------|-------------------------|-----------------------|--|
| 10/08/2012 | 3.0.1 | Editorial | Changed language and formatting in the technical content. |
| 02/11/2013 | 3.0.1 | No change | No changes to the meaning, language, or formatting of the technical content. |
| 07/30/2013 | 3.0.1 | No change | No changes to the meaning, language, or formatting of the technical content. |
| 11/18/2013 | 3.1 | Minor | Clarified the meaning of the technical content. |
| 02/10/2014 | 3.1 | No change | No changes to the meaning, language, or formatting of the technical content. |
| 04/30/2014 | 3.2 | Minor | Clarified the meaning of the technical content. |
| 07/31/2014 | 3.3 | Minor | Clarified the meaning of the technical content. |

Table of Contents

| | |
|---|-----------|
| 1 Introduction | 13 |
| 1.1 Glossary | 13 |
| 1.2 References | 15 |
| 1.2.1 Normative References | 15 |
| 1.2.2 Informative References | 16 |
| 1.3 Overview | 16 |
| 1.3.1 Published Items and Results Returned by this Protocol | 17 |
| 1.4 Relationship to Other Protocols | 17 |
| 1.5 Prerequisites/Preconditions | 18 |
| 1.6 Applicability Statement | 18 |
| 1.7 Versioning and Capability Negotiation | 18 |
| 1.8 Vendor-Extensible Fields | 18 |
| 1.9 Standards Assignments | 18 |
| 2 Messages | 19 |
| 2.1 Transport | 19 |
| 2.2 Common Message Syntax | 19 |
| 2.2.1 Namespaces | 19 |
| 2.2.2 Messages | 19 |
| 2.2.3 Elements | 20 |
| 2.2.4 Complex Types | 20 |
| 2.2.4.1 ArrayOfAnyType | 20 |
| 2.2.4.2 ArrayOfStatus | 20 |
| 2.2.4.3 ArrayOfString | 21 |
| 2.2.4.4 RangeCoordinates | 21 |
| 2.2.4.5 Status | 22 |
| 2.2.4.6 TypeExporter | 22 |
| 2.2.5 Simple Types | 22 |
| 2.2.5.1 CellError | 23 |
| 2.2.5.2 RangeReference | 23 |
| 2.2.5.3 SheetName | 24 |
| 2.2.5.4 WorkbookType | 24 |
| 2.2.6 Attributes | 25 |
| 2.2.7 Groups | 25 |
| 2.2.8 Attribute Groups | 25 |
| 2.2.9 Common Data Structures | 25 |
| 2.2.9.1 Faults | 25 |
| 2.2.9.2 Common List of Subcodes | 35 |
| 3 Protocol Details | 37 |
| 3.1 Server Details | 38 |
| 3.1.1 Abstract Data Model | 38 |
| 3.1.2 Timers | 38 |
| 3.1.2.1 Session Timeout | 38 |
| 3.1.2.2 Request Timeout | 38 |
| 3.1.3 Initialization | 38 |
| 3.1.4 Message Processing Events and Sequencing Rules | 38 |
| 3.1.4.1 Calculate | 40 |
| 3.1.4.1.1 Messages | 41 |
| 3.1.4.1.1.1 CalculateSoapIn | 41 |

| | |
|---|----|
| 3.1.4.1.1.2 CalculateSoapOut | 41 |
| 3.1.4.1.2 Elements..... | 41 |
| 3.1.4.1.2.1 Calculate | 41 |
| 3.1.4.1.2.2 CalculateResponse..... | 42 |
| 3.1.4.1.3 Complex Types | 42 |
| 3.1.4.1.4 Simple Types..... | 42 |
| 3.1.4.1.5 Attributes..... | 42 |
| 3.1.4.1.6 Groups..... | 42 |
| 3.1.4.1.7 Attribute Groups | 42 |
| 3.1.4.2 CalculateA1..... | 42 |
| 3.1.4.2.1 Messages | 43 |
| 3.1.4.2.1.1 CalculateA1SoapIn..... | 43 |
| 3.1.4.2.1.2 CalculateA1SoapOut | 44 |
| 3.1.4.2.2 Elements..... | 44 |
| 3.1.4.2.2.1 CalculateA1 | 44 |
| 3.1.4.2.2.2 CalculateA1Response | 44 |
| 3.1.4.2.3 Complex Types | 45 |
| 3.1.4.2.4 Simple Types..... | 45 |
| 3.1.4.2.5 Attributes..... | 45 |
| 3.1.4.2.6 Groups..... | 45 |
| 3.1.4.2.7 Attribute Groups | 45 |
| 3.1.4.3 CalculateWorkbook | 45 |
| 3.1.4.3.1 Messages | 46 |
| 3.1.4.3.1.1 CalculateWorkbookSoapIn | 46 |
| 3.1.4.3.1.2 CalculateWorkbookSoapOut | 46 |
| 3.1.4.3.2 Elements..... | 46 |
| 3.1.4.3.2.1 CalculateWorkbook | 46 |
| 3.1.4.3.2.2 CalculateWorkbookResponse | 47 |
| 3.1.4.3.3 Complex Types | 47 |
| 3.1.4.3.4 Simple Types..... | 47 |
| 3.1.4.3.4.1 CalculateType | 47 |
| 3.1.4.3.5 Attributes..... | 48 |
| 3.1.4.3.6 Groups..... | 48 |
| 3.1.4.3.7 Attribute Groups | 48 |
| 3.1.4.4 CancelRequest | 48 |
| 3.1.4.4.1 Messages | 49 |
| 3.1.4.4.1.1 CancelRequestSoapIn | 49 |
| 3.1.4.4.1.2 CancelRequestSoapOut | 49 |
| 3.1.4.4.2 Elements..... | 49 |
| 3.1.4.4.2.1 CancelRequest | 49 |
| 3.1.4.4.2.2 CancelRequestResponse | 50 |
| 3.1.4.4.3 Complex Types | 50 |
| 3.1.4.4.4 Simple Types..... | 50 |
| 3.1.4.4.5 Attributes..... | 50 |
| 3.1.4.4.6 Groups..... | 50 |
| 3.1.4.4.7 Attribute Groups | 50 |
| 3.1.4.5 CloseWorkbook | 50 |
| 3.1.4.5.1 Messages | 51 |
| 3.1.4.5.1.1 CloseWorkbookSoapIn | 51 |
| 3.1.4.5.1.2 CloseWorkbookSoapOut | 51 |
| 3.1.4.5.2 Elements..... | 51 |
| 3.1.4.5.2.1 CloseWorkbook | 51 |
| 3.1.4.5.2.2 CloseWorkbookResponse | 52 |

| | | |
|-------------|----------------------------------|----|
| 3.1.4.5.3 | Complex Types | 52 |
| 3.1.4.5.4 | Simple Types..... | 52 |
| 3.1.4.5.5 | Attributes..... | 52 |
| 3.1.4.5.6 | Groups..... | 52 |
| 3.1.4.5.7 | Attribute Groups | 52 |
| 3.1.4.6 | EnsureWorkbookModel | 52 |
| 3.1.4.6.1 | Messages | 53 |
| 3.1.4.6.1.1 | EnsureWorkbookModelSoapIn | 53 |
| 3.1.4.6.1.2 | EnsureWorkbookModelSoapOut | 53 |
| 3.1.4.6.2 | Elements..... | 53 |
| 3.1.4.6.2.1 | EnsureWorkbookModel | 53 |
| 3.1.4.6.2.2 | EnsureWorkbookModelResponse..... | 54 |
| 3.1.4.6.3 | Complex Types | 54 |
| 3.1.4.6.3.1 | WorkbookModelInfo | 54 |
| 3.1.4.6.4 | Simple Types..... | 55 |
| 3.1.4.6.5 | Attributes..... | 55 |
| 3.1.4.6.6 | Groups..... | 55 |
| 3.1.4.6.7 | Attribute Groups | 55 |
| 3.1.4.7 | GetApiVersion | 55 |
| 3.1.4.7.1 | Messages | 55 |
| 3.1.4.7.1.1 | GetApiVersionSoapIn | 56 |
| 3.1.4.7.1.2 | GetApiVersionSoapOut | 56 |
| 3.1.4.7.2 | Elements..... | 56 |
| 3.1.4.7.2.1 | GetApiVersion | 56 |
| 3.1.4.7.2.2 | GetApiVersionResponse..... | 56 |
| 3.1.4.7.3 | Complex Types | 57 |
| 3.1.4.7.4 | Simple Types..... | 57 |
| 3.1.4.7.5 | Attributes..... | 57 |
| 3.1.4.7.6 | Groups..... | 57 |
| 3.1.4.7.7 | Attribute Groups | 57 |
| 3.1.4.8 | GetCell | 57 |
| 3.1.4.8.1 | Messages | 58 |
| 3.1.4.8.1.1 | GetCellSoapIn | 58 |
| 3.1.4.8.1.2 | GetCellSoapOut..... | 58 |
| 3.1.4.8.2 | Elements..... | 58 |
| 3.1.4.8.2.1 | GetCell..... | 58 |
| 3.1.4.8.2.2 | GetCellResponse..... | 59 |
| 3.1.4.8.3 | Complex Types | 59 |
| 3.1.4.8.4 | Simple Types..... | 59 |
| 3.1.4.8.5 | Attributes..... | 60 |
| 3.1.4.8.6 | Groups..... | 60 |
| 3.1.4.8.7 | Attribute Groups | 60 |
| 3.1.4.9 | GetCellA1 | 60 |
| 3.1.4.9.1 | Messages | 61 |
| 3.1.4.9.1.1 | GetCellA1SoapIn | 61 |
| 3.1.4.9.1.2 | GetCellA1SoapOut | 61 |
| 3.1.4.9.2 | Elements..... | 61 |
| 3.1.4.9.2.1 | GetCellA1 | 61 |
| 3.1.4.9.2.2 | GetCellA1Response..... | 62 |
| 3.1.4.9.3 | Complex Types | 62 |
| 3.1.4.9.4 | Simple Types..... | 62 |
| 3.1.4.9.5 | Attributes..... | 62 |
| 3.1.4.9.6 | Groups..... | 63 |

| | | |
|--------------|-------------------------------------|----|
| 3.1.4.9.7 | Attribute Groups | 63 |
| 3.1.4.10 | GetChartImageUrl..... | 63 |
| 3.1.4.10.1 | Messages | 63 |
| 3.1.4.10.1.1 | GetChartImageUrlSoapIn..... | 63 |
| 3.1.4.10.1.2 | GetChartImageUrlSoapOut..... | 63 |
| 3.1.4.10.2 | Elements..... | 64 |
| 3.1.4.10.2.1 | GetChartImageUrl..... | 64 |
| 3.1.4.10.2.2 | GetChartImageUrlResponse | 64 |
| 3.1.4.10.3 | Complex Types | 65 |
| 3.1.4.10.3.1 | Size | 65 |
| 3.1.4.10.4 | Simple Types | 65 |
| 3.1.4.10.5 | Attributes..... | 65 |
| 3.1.4.10.6 | Groups..... | 65 |
| 3.1.4.10.7 | Attribute Groups..... | 65 |
| 3.1.4.11 | GetPublishedItemNames | 65 |
| 3.1.4.11.1 | Messages | 66 |
| 3.1.4.11.1.1 | GetPublishedItemNamesSoapIn..... | 66 |
| 3.1.4.11.1.2 | GetPublishedItemNamesSoapOut..... | 66 |
| 3.1.4.11.2 | Elements..... | 66 |
| 3.1.4.11.2.1 | GetPublishedItemNames..... | 67 |
| 3.1.4.11.2.2 | GetPublishedItemNamesResponse | 67 |
| 3.1.4.11.3 | Complex Types | 67 |
| 3.1.4.11.3.1 | ArrayOfWorkbookItem..... | 67 |
| 3.1.4.11.3.2 | WorkbookItem | 68 |
| 3.1.4.11.4 | Simple Types | 68 |
| 3.1.4.11.4.1 | ItemType..... | 68 |
| 3.1.4.11.5 | Attributes..... | 69 |
| 3.1.4.11.6 | Groups..... | 69 |
| 3.1.4.11.7 | Attribute Groups..... | 69 |
| 3.1.4.12 | GetRange..... | 69 |
| 3.1.4.12.1 | Messages | 70 |
| 3.1.4.12.1.1 | GetRangeSoapIn | 70 |
| 3.1.4.12.1.2 | GetRangeSoapOut | 70 |
| 3.1.4.12.2 | Elements..... | 71 |
| 3.1.4.12.2.1 | GetRange | 71 |
| 3.1.4.12.2.2 | GetRangeResponse | 71 |
| 3.1.4.12.3 | Complex Types | 72 |
| 3.1.4.12.4 | Simple Types | 72 |
| 3.1.4.12.5 | Attributes..... | 72 |
| 3.1.4.12.6 | Groups..... | 72 |
| 3.1.4.12.7 | Attribute Groups..... | 72 |
| 3.1.4.13 | GetRangeA1..... | 72 |
| 3.1.4.13.1 | Messages | 73 |
| 3.1.4.13.1.1 | GetRangeA1SoapIn..... | 73 |
| 3.1.4.13.1.2 | GetRangeA1SoapOut..... | 74 |
| 3.1.4.13.2 | Elements..... | 74 |
| 3.1.4.13.2.1 | GetRangeA1..... | 74 |
| 3.1.4.13.2.2 | GetRangeA1Response | 74 |
| 3.1.4.13.3 | Complex Types | 75 |
| 3.1.4.13.4 | Simple Types | 75 |
| 3.1.4.13.5 | Attributes..... | 75 |
| 3.1.4.13.6 | Groups..... | 75 |
| 3.1.4.13.7 | Attribute Groups..... | 75 |

| | | |
|--------------|--------------------------------------|----|
| 3.1.4.14 | GetSessionInformation | 75 |
| 3.1.4.14.1 | Messages | 76 |
| 3.1.4.14.1.1 | GetSessionInformationSoapIn | 76 |
| 3.1.4.14.1.2 | GetSessionInformationSoapOut | 76 |
| 3.1.4.14.2 | Elements | 76 |
| 3.1.4.14.2.1 | GetSessionInformation | 76 |
| 3.1.4.14.2.2 | GetSessionInformationResponse | 77 |
| 3.1.4.14.3 | Complex Types | 77 |
| 3.1.4.14.4 | Simple Types | 77 |
| 3.1.4.14.5 | Attributes | 77 |
| 3.1.4.14.6 | Groups | 77 |
| 3.1.4.14.7 | Attribute Groups | 77 |
| 3.1.4.15 | GetSheetNames | 77 |
| 3.1.4.15.1 | Messages | 78 |
| 3.1.4.15.1.1 | GetSheetNamesSoapIn | 78 |
| 3.1.4.15.1.2 | GetSheetNamesSoapOut | 78 |
| 3.1.4.15.2 | Elements | 78 |
| 3.1.4.15.2.1 | GetSheetNames | 79 |
| 3.1.4.15.2.2 | GetSheetNamesResponse | 79 |
| 3.1.4.15.3 | Complex Types | 79 |
| 3.1.4.15.3.1 | ArrayOfSheetInfo | 80 |
| 3.1.4.15.3.2 | SheetInfo | 80 |
| 3.1.4.15.4 | Simple Types | 80 |
| 3.1.4.15.4.1 | SheetType | 80 |
| 3.1.4.15.4.2 | SheetVisibility | 81 |
| 3.1.4.15.5 | Attributes | 81 |
| 3.1.4.15.6 | Groups | 81 |
| 3.1.4.15.7 | Attribute Groups | 81 |
| 3.1.4.16 | GetWorkbook | 81 |
| 3.1.4.16.1 | Messages | 82 |
| 3.1.4.16.1.1 | GetWorkbookSoapIn | 82 |
| 3.1.4.16.1.2 | GetWorkbookSoapOut | 82 |
| 3.1.4.16.2 | Elements | 83 |
| 3.1.4.16.2.1 | GetWorkbook | 83 |
| 3.1.4.16.2.2 | GetWorkbookResponse | 83 |
| 3.1.4.16.3 | Complex Types | 83 |
| 3.1.4.16.4 | Simple Types | 84 |
| 3.1.4.16.5 | Attributes | 84 |
| 3.1.4.16.6 | Groups | 84 |
| 3.1.4.16.7 | Attribute Groups | 84 |
| 3.1.4.17 | GetWorkbookConnections | 84 |
| 3.1.4.17.1 | Messages | 84 |
| 3.1.4.17.1.1 | GetWorkbookConnectionsSoapIn | 84 |
| 3.1.4.17.1.2 | GetWorkbookConnectionsSoapOut | 85 |
| 3.1.4.17.2 | Elements | 85 |
| 3.1.4.17.2.1 | GetWorkbookConnections | 85 |
| 3.1.4.17.2.2 | GetWorkbookConnectionsResponse | 85 |
| 3.1.4.17.3 | Complex Types | 86 |
| 3.1.4.17.4 | Simple Types | 86 |
| 3.1.4.17.5 | Attributes | 86 |
| 3.1.4.17.6 | Groups | 86 |
| 3.1.4.17.7 | Attribute Groups | 86 |
| 3.1.4.18 | NewWorkbook | 86 |

| | | |
|--------------|--------------------------------------|----|
| 3.1.4.18.1 | Messages | 87 |
| 3.1.4.18.1.1 | NewWorkbookSoapIn | 87 |
| 3.1.4.18.1.2 | NewWorkbookSoapOut | 87 |
| 3.1.4.18.2 | Elements | 87 |
| 3.1.4.18.2.1 | NewWorkbook | 87 |
| 3.1.4.18.2.2 | NewWorkbookResponse | 88 |
| 3.1.4.18.3 | Complex Types | 88 |
| 3.1.4.18.4 | Simple Types | 88 |
| 3.1.4.18.5 | Attributes | 88 |
| 3.1.4.18.6 | Groups | 88 |
| 3.1.4.18.7 | Attribute Groups | 88 |
| 3.1.4.19 | OpenWorkbook | 89 |
| 3.1.4.19.1 | Messages | 90 |
| 3.1.4.19.1.1 | OpenWorkbookSoapIn | 90 |
| 3.1.4.19.1.2 | OpenWorkbookSoapOut | 90 |
| 3.1.4.19.2 | Elements | 90 |
| 3.1.4.19.2.1 | OpenWorkbook | 91 |
| 3.1.4.19.2.2 | OpenWorkbookResponse | 91 |
| 3.1.4.19.3 | Complex Types | 91 |
| 3.1.4.19.4 | Simple Types | 92 |
| 3.1.4.19.5 | Attributes | 92 |
| 3.1.4.19.6 | Groups | 92 |
| 3.1.4.19.7 | Attribute Groups | 92 |
| 3.1.4.20 | OpenWorkbookEx | 92 |
| 3.1.4.20.1 | Messages | 92 |
| 3.1.4.20.1.1 | OpenWorkbookExSoapIn | 92 |
| 3.1.4.20.1.2 | OpenWorkbookExSoapOut | 93 |
| 3.1.4.20.2 | Elements | 93 |
| 3.1.4.20.2.1 | OpenWorkbookEx | 93 |
| 3.1.4.20.2.2 | OpenWorkbookExResponse | 93 |
| 3.1.4.20.3 | Complex Types | 94 |
| 3.1.4.20.4 | Simple Types | 94 |
| 3.1.4.20.5 | Attributes | 94 |
| 3.1.4.20.6 | Groups | 94 |
| 3.1.4.20.7 | Attribute Groups | 94 |
| 3.1.4.21 | OpenWorkbookForEditing | 94 |
| 3.1.4.21.1 | Messages | 95 |
| 3.1.4.21.1.1 | OpenWorkbookForEditingSoapIn | 96 |
| 3.1.4.21.1.2 | OpenWorkbookForEditingSoapOut | 96 |
| 3.1.4.21.2 | Elements | 96 |
| 3.1.4.21.2.1 | OpenWorkbookForEditing | 96 |
| 3.1.4.21.2.2 | OpenWorkbookForEditingResponse | 97 |
| 3.1.4.21.3 | Complex Types | 97 |
| 3.1.4.21.4 | Simple Types | 97 |
| 3.1.4.21.5 | Attributes | 97 |
| 3.1.4.21.6 | Groups | 97 |
| 3.1.4.21.7 | Attribute Groups | 97 |
| 3.1.4.22 | Refresh | 97 |
| 3.1.4.22.1 | Messages | 98 |
| 3.1.4.22.1.1 | RefreshSoapIn | 98 |
| 3.1.4.22.1.2 | RefreshSoapOut | 98 |
| 3.1.4.22.2 | Elements | 98 |
| 3.1.4.22.2.1 | Refresh | 99 |

| | | |
|--------------|-------------------------------------|-----|
| 3.1.4.22.2 | RefreshResponse | 99 |
| 3.1.4.22.3 | Complex Types | 99 |
| 3.1.4.22.4 | Simple Types | 99 |
| 3.1.4.22.5 | Attributes | 99 |
| 3.1.4.22.6 | Groups..... | 99 |
| 3.1.4.22.7 | Attribute Groups..... | 99 |
| 3.1.4.23 | RefreshEx | 100 |
| 3.1.4.23.1 | Messages | 100 |
| 3.1.4.23.1.1 | RefreshExSoapIn | 100 |
| 3.1.4.23.1.2 | RefreshExSoapOut | 100 |
| 3.1.4.23.2 | Elements | 100 |
| 3.1.4.23.2.1 | RefreshEx | 101 |
| 3.1.4.23.2.2 | RefreshExResponse..... | 101 |
| 3.1.4.23.3 | Complex Types | 101 |
| 3.1.4.23.4 | Simple Types | 101 |
| 3.1.4.23.5 | Attributes | 101 |
| 3.1.4.23.6 | Groups..... | 101 |
| 3.1.4.23.7 | Attribute Groups..... | 102 |
| 3.1.4.24 | SaveWorkbook | 102 |
| 3.1.4.24.1 | Messages | 102 |
| 3.1.4.24.1.1 | SaveWorkbookSoapIn | 102 |
| 3.1.4.24.1.2 | SaveWorkbookSoapOut | 103 |
| 3.1.4.24.2 | Elements | 103 |
| 3.1.4.24.2.1 | SaveWorkbook | 103 |
| 3.1.4.24.2.2 | SaveWorkbookResponse | 103 |
| 3.1.4.24.3 | Complex Types | 104 |
| 3.1.4.24.4 | Simple Types | 104 |
| 3.1.4.24.5 | Attributes | 104 |
| 3.1.4.24.6 | Groups..... | 104 |
| 3.1.4.24.7 | Attribute Groups..... | 104 |
| 3.1.4.25 | SaveWorkbookCopy | 104 |
| 3.1.4.25.1 | Messages | 105 |
| 3.1.4.25.1.1 | SaveWorkbookCopySoapIn | 105 |
| 3.1.4.25.1.2 | SaveWorkbookCopySoapOut | 105 |
| 3.1.4.25.2 | Elements | 105 |
| 3.1.4.25.2.1 | SaveWorkbookCopy | 105 |
| 3.1.4.25.2.2 | SaveWorkbookCopyResponse | 106 |
| 3.1.4.25.3 | Complex Types | 106 |
| 3.1.4.25.4 | Simple Types | 106 |
| 3.1.4.25.4.1 | SaveOptions..... | 107 |
| 3.1.4.25.5 | Attributes | 107 |
| 3.1.4.25.6 | Groups..... | 107 |
| 3.1.4.25.7 | Attribute Groups..... | 107 |
| 3.1.4.26 | SetCalculationOptions..... | 107 |
| 3.1.4.26.1 | Messages | 108 |
| 3.1.4.26.1.1 | SetCalculationOptionsSoapIn..... | 108 |
| 3.1.4.26.1.2 | SetCalculationOptionsSoapOut | 108 |
| 3.1.4.26.2 | Elements | 108 |
| 3.1.4.26.2.1 | SetCalculationOptions | 108 |
| 3.1.4.26.2.2 | SetCalculationOptionsResponse | 109 |
| 3.1.4.26.3 | Complex Types | 109 |
| 3.1.4.26.4 | Simple Types | 109 |
| 3.1.4.26.4.1 | WorkbookCalculation..... | 109 |

| | | |
|--------------|-----------------------------|-----|
| 3.1.4.26.5 | Attributes | 110 |
| 3.1.4.26.6 | Groups..... | 110 |
| 3.1.4.26.7 | Attribute Groups..... | 110 |
| 3.1.4.27 | SetCell | 110 |
| 3.1.4.27.1 | Messages | 111 |
| 3.1.4.27.1.1 | SetCellSoapIn | 111 |
| 3.1.4.27.1.2 | SetCellSoapOut | 111 |
| 3.1.4.27.2 | Elements..... | 111 |
| 3.1.4.27.2.1 | SetCell | 111 |
| 3.1.4.27.2.2 | SetCellResponse | 112 |
| 3.1.4.27.3 | Complex Types | 112 |
| 3.1.4.27.4 | Simple Types | 112 |
| 3.1.4.27.5 | Attributes | 112 |
| 3.1.4.27.6 | Groups..... | 113 |
| 3.1.4.27.7 | Attribute Groups..... | 113 |
| 3.1.4.28 | SetCellA1..... | 113 |
| 3.1.4.28.1 | Messages | 114 |
| 3.1.4.28.1.1 | SetCellA1SoapIn..... | 114 |
| 3.1.4.28.1.2 | SetCellA1SoapOut..... | 114 |
| 3.1.4.28.2 | Elements..... | 114 |
| 3.1.4.28.2.1 | SetCellA1..... | 114 |
| 3.1.4.28.2.2 | SetCellA1Response | 115 |
| 3.1.4.28.3 | Complex Types | 115 |
| 3.1.4.28.4 | Simple Types | 115 |
| 3.1.4.28.5 | Attributes | 115 |
| 3.1.4.28.6 | Groups..... | 115 |
| 3.1.4.28.7 | Attribute Groups..... | 115 |
| 3.1.4.29 | SetParameters..... | 116 |
| 3.1.4.29.1 | Messages | 117 |
| 3.1.4.29.1.1 | SetParametersSoapIn..... | 117 |
| 3.1.4.29.1.2 | SetParametersSoapOut | 117 |
| 3.1.4.29.2 | Elements..... | 117 |
| 3.1.4.29.2.1 | SetParameters | 117 |
| 3.1.4.29.2.2 | SetParametersResponse | 118 |
| 3.1.4.29.3 | Complex Types | 118 |
| 3.1.4.29.3.1 | ArrayOfParameterInfo | 118 |
| 3.1.4.29.3.2 | ParameterInfo | 118 |
| 3.1.4.29.4 | Simple Types | 119 |
| 3.1.4.29.5 | Attributes | 119 |
| 3.1.4.29.6 | Groups..... | 119 |
| 3.1.4.29.7 | Attribute Groups..... | 119 |
| 3.1.4.30 | SetRange..... | 119 |
| 3.1.4.30.1 | Messages | 120 |
| 3.1.4.30.1.1 | SetRangeSoapIn..... | 120 |
| 3.1.4.30.1.2 | SetRangeSoapOut..... | 120 |
| 3.1.4.30.2 | Elements..... | 121 |
| 3.1.4.30.2.1 | SetRange..... | 121 |
| 3.1.4.30.2.2 | SetRangeResponse | 121 |
| 3.1.4.30.3 | Complex Types | 122 |
| 3.1.4.30.4 | Simple Types | 122 |
| 3.1.4.30.5 | Attributes | 122 |
| 3.1.4.30.6 | Groups..... | 122 |
| 3.1.4.30.7 | Attribute Groups..... | 122 |

| | | |
|--------------|--|------------|
| 3.1.4.31 | SetRangeA1 | 122 |
| 3.1.4.31.1 | Messages | 123 |
| 3.1.4.31.1.1 | SetRangeA1SoapIn | 123 |
| 3.1.4.31.1.2 | SetRangeA1SoapOut | 123 |
| 3.1.4.31.2 | Elements | 123 |
| 3.1.4.31.2.1 | SetRangeA1 | 123 |
| 3.1.4.31.2.2 | SetRangeA1Response..... | 124 |
| 3.1.4.31.3 | Complex Types | 124 |
| 3.1.4.31.4 | Simple Types | 124 |
| 3.1.4.31.5 | Attributes | 124 |
| 3.1.4.31.6 | Groups..... | 124 |
| 3.1.4.31.7 | Attribute Groups..... | 124 |
| 3.1.5 | Timer Events | 125 |
| 3.1.6 | Other Local Events | 125 |
| 4 | Protocol Examples..... | 126 |
| 4.1 | Getting the Value of a Worksheet Cell | 126 |
| 4.2 | Refreshing Data and Retrieving a Workbook..... | 129 |
| 5 | Security..... | 141 |
| 5.1 | Security Considerations for Implementers..... | 141 |
| 5.2 | Index of Security Parameters | 141 |
| 6 | Appendix A: Full WSDL | 142 |
| 7 | Appendix B: Product Behavior | 171 |
| 8 | Change Tracking..... | 173 |
| 9 | Index | 176 |

1 Introduction

The Excel Web Services Protocol is a Web service protocol that is used to read from and perform basic operations on workbooks hosted on a protocol server.

Sections 1.8, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in [\[RFC2119\]](#). Sections 1.5 and 1.9 are also normative but does not contain those terms. All other sections and examples in this specification are informative.

1.1 Glossary

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

- Hypertext Transfer Protocol (HTTP)**
- Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)**
- SOAP**
- SOAP action**
- SOAP body**
- SOAP fault**
- Universal Naming Convention (UNC)**
- XML namespace**

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

- active sheet**
- array formula**
- AutoFilter**
- automatic calculation mode**
- calculate**
- calculation mode**
- cell**
- chart sheet**
- checked out**
- column**
- confirmation**
- connected data**
- cube**
- data connection**
- data culture**
- data provider**
- data table**
- default filter value**
- dimension**
- document library**
- edit session**
- farm**
- filter**
- formula**
- grow**
- Information Rights Management (IRM)**
- item**
- level**
- license key**
- list**

locked
manual calculation mode
manual filter
merged cell
named object
named range
new workbook session
non-contiguous range
number format
OLAP
Open Item permission
permission
PivotTable
PivotTable member
PivotTable page field
published
published item
published items snapshot
published sheet
query
range
refresh
row
session
session identifier
sheet
site
slicer
snapshot
table
trusted location
UI culture
Uniform Resource Identifier (URI)
Uniform Resource Locator (URL)
user-defined function (UDF)
view session
view-only mode
visible
web service
Web Services Description Language (WSDL)
workbook
workbook file
workbook parameter
worksheet
WSDL message
WSDL operation
XML namespace prefix
XML schema

The following terms are specific to this document:

cell error value: Any of a number of special values that are returned as a result of an unsuccessful formula calculation.

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.

[ISO/IEC29500-1:2011] ISO/IEC, "Information Technology -- Document description and processing languages -- Office Open XML File Formats -- Part 1: Fundamentals and Markup Language Reference", ISO/IEC 29500-1:2011, 2011, http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=59575

[ISO/IEC29500-1:2012] ISO/IEC, "Information Technology -- Document description and processing languages -- Office Open XML File Formats -- Part 1: Fundamentals and Markup Language Reference", ISO/IEC 29500-1:2012, http://www.iso.org/iso/home/store/catalogue_ics/catalogue_detail_ics.htm?csnumber=61750

[MS-ODBCSTR] Microsoft Corporation, "[ODBC Connection String Structure Specification](#)".

[MS-ODCFF] Microsoft Corporation, "[Office Data Connection File Format Structure Specification](#)".

[MS-OFFMACRO2] Microsoft Corporation, "[Office Macro-Enabled File Format Version 2 Structure Specification](#)".

[MS-OLEDBSTR] Microsoft Corporation, "[OLEDB Connection String Structure Specification](#)".

[MS-XLSB] Microsoft Corporation, "[Excel Binary File Format \(.xlsb\) Structure Specification](#)".

[ODF] OASIS, "Open Document Format for Office Applications (OpenDocument) v1.1", 1 February 2007, <http://docs.oasis-open.org/office/v1.1/OS/OpenDocument-v1.1-html/OpenDocument-v1.1.html>

[RFC1766] Alvestrand, H., "Tags for the Identification of Languages", RFC 1766, March 1995, <http://www.ietf.org/rfc/rfc1766.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>

[RFC2396] Berners-Lee, T., Fielding, R., and Masinter, L., "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998, <http://www.ietf.org/rfc/rfc2396.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>

[RFC3986] Berners-Lee, T., Fielding, R., and Masinter, L., "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005, <http://www.ietf.org/rfc/rfc3986.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., Beech, D., Maloney, M., and Mendelsohn, N., Eds., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, <http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/>

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

1.2.2 Informative References

[MSDN-ESO] Microsoft Corporation, "Excel Services Overview", <http://msdn.microsoft.com/en-us/library/ms546696.aspx>

[MSDN-UFES] Microsoft Corporation, "Unsupported Features in Excel Services", <http://msdn.microsoft.com/en-us/library/ms496823.aspx>

[MSDN-WDCAEWS] Microsoft Corporation, "Walkthrough: Developing a Custom Application Using Excel Web Services", <http://msdn.microsoft.com/en-us/library/ms519100.aspx>

[MSFT-ACRR] Microsoft Corporation, "About Cell and Range References", <http://office.microsoft.com/en-us/help/HP051983231033.aspx>

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

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

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

1.3 Overview

This protocol is a Web service based protocol that is used to access **workbooks** controlled by a protocol server. The protocol specified in this document is stateful and relies on **sessions (2)** to maintain state. Workbooks hosted on a protocol server are used to present, store, manipulate, and **calculate** tabular data. For an overview of the specific implementation of a workbook server system that uses this protocol, see [\[MSDN-ESO\]](#).

The protocol provides the following sets of functionality that can be performed on workbooks:

- Open existing workbooks for viewing and editing and close open workbooks.
- Create new workbooks.

- Save workbooks.
- Retrieve the full contents of the current open copy.
- Recalculation.
- **Refresh connected data.**
- Set specific values into a **worksheet cell** or **range**.
- Get specific values from a worksheet cell or range.
- Cancel a request.
- Get the names of all the **published items** in a workbook.
- Get a **URL** to an image of a chart in a workbook.
- Get information about a session.
- Get the version of the protocol server.

1.3.1 Published Items and Results Returned by this Protocol

A protocol server can have functionality where only portions of the workbook are available for use by protocol clients. The parts of the workbook that are accessible from the protocol server are said to be published. Some of the operations in this protocol could fail if the workbook item being exposed or changed is not published and the user does not have sufficient permissions to view items that have not been published.

1.4 Relationship to Other Protocols

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

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

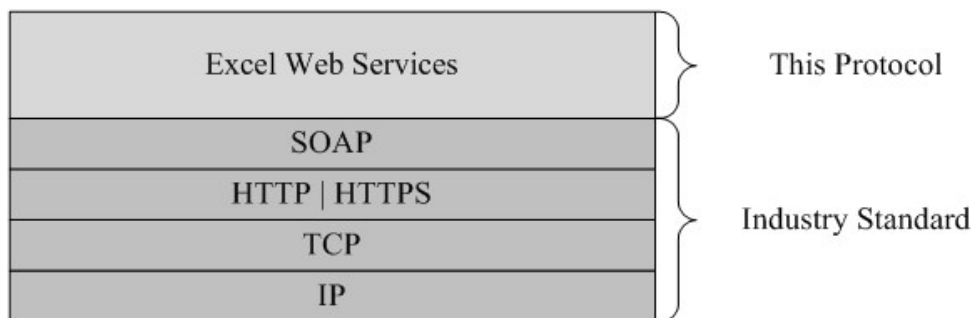


Figure 1: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a **site (2)** that is identified by a URL that is known by protocol clients. The protocol server endpoint is formed by appending "_vti_bin/ExcelService.asmx" to the URL of the site (2), for example http://www.contoso.com/Repository/_vti_bin/ExcelService.asmx.

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

1.6 Applicability Statement

This protocol can be used at any time to access a simple interface that supports basic operations that leverage server side calculation of workbooks is needed. It is appropriate to use this as a public-facing interface to workbooks hosted on a server.

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 SOAP messages, as described in section [2.1](#).
- **Protocol versions:** This protocol is not versioned.
- **Capability negotiation:** This protocol does not support capability 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 behavior 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 HTTP. Protocol servers **MAY** additionally support SOAP over HTTPS for securing communication with clients.

Protocol messages **MUST** be formatted as specified either in [\[SOAP1.1\]](#) section 4 or in [\[SOAP1.2/1\]](#) section 5. Protocol server faults **MUST** be returned either by using HTTP Status Codes as specified in [\[RFC2616\]](#) section 10, or by using **SOAP faults** as specified either in [\[SOAP1.1\]](#), section 4.4 or in [\[SOAP1.2/1\]](#) section 5.4. The version of the SOAP fault returned **MUST** correspond to the version of SOAP used for the request **WSDL message**.

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 |
|--------|---|--|
| http | http://schemas.xmlsoap.org/wsdl/http/ | |
| soap | http://schemas.xmlsoap.org/wsdl/soap/ | [SOAP1.1] |
| soap12 | http://schemas.xmlsoap.org/wsdl/soap12/ | [SOAP1.2/1] [SOAP1.2/2] |
| tns | http://schemas.microsoft.com/office/excel/server/webservices | |
| wsdl | http://schemas.xmlsoap.org/wsdl/ | [WSDL] |
| xs | http://www.w3.org/2001/XMLSchema | [XMLSCHEMA1] [XMLSCHEMA2] |

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 |
|-------------------------|--|
| ArrayOfAnyType | An array of anyType elements defined as follows. |
| ArrayOfStatus | An array of status information. |
| ArrayOfString | An array of string elements. |
| RangeCoordinates | Specifies a range on a workbook. |
| Status | The status information returned by the protocol server defined as follows. |
| TypeExporter | MUST NOT be used. |

2.2.4.1 ArrayOfAnyType

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

An array of **anyType** elements defined as follows.

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

anyType: An **anyType** value.

2.2.4.2 ArrayOfStatus

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

An array of status information.

- The protocol server MUST use the **Status** complex type to return informative status messages to the protocol client for operations that succeed.
- The protocol client MUST NOT perform any logic based on the specific values in the **Name** or **Message** elements returned in the **Status** type.

This complex type is defined as follows.

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

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

Status: The status information returned by the protocol server.

2.2.4.3 ArrayOfString

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

An array of string elements.

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

string: A string value.

2.2.4.4 RangeCoordinates

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

Specifies a range on a workbook.

- **RangeCoordinates** MUST be considered empty if **Row**, **Column**, **Height**, and **Width** are all 0. Otherwise, **RangeCoordinates** MUST NOT be considered empty.
- If **Height** is 0, then **Row**, **Column**, and **Width** MUST be 0.
- If **Width** is 0, then **Row**, **Column**, and **Height** MUST be 0.

This complex type is defined as follows.

```
<xs:complexType name="RangeCoordinates" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Row" type="xs:int"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Column" type="xs:int"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Height" type="xs:int"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Width" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

Row: The **row (2)** index. MUST be greater than or equal to 0 and less than or equal to 1,048,575.

Column: The **column (2)** index. MUST be greater than or equal to 0 and less than or equal to 16,383.

Height: The number of rows (2). MUST be greater than or equal to 0. The sum of **Row** and **Height** MUST NOT be greater than 1,048,576.

Width: The number of columns (2). MUST be greater than or equal to 0. The sum of **Column** and **Width** MUST NOT be greater than 16,384.

2.2.4.5 Status

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

The status information returned by the protocol server defined as follows.

```
<xs:complexType name="Status" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Severity" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Message" type="xs:string"/>
  </xs:sequence>
</xs:complexType>
```

Name: The name of the status message. MUST be present. SHOULD be any subcode specified in section [2.2.9.1](#), but MAY be any other string.

Severity: The severity of the status message. MUST contain one of the values described in the following table.

| Value | Meaning |
|-------------|--|
| Error | The status is of severity error. |
| Warning | The status is of severity warning. |
| Information | The status is of severity information. |

Message: The message description.

2.2.4.6 TypeExporter

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

MUST NOT be used.

It is defined as follows.

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

CellError: MUST NOT be used.

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 |
|-----------------------|---|
| CellError | The CellError type is an enumeration of a set of cell error values that a worksheet cell can contain. |
| RangeReference | A string that is used by the protocol server to identify a range on a sheet (2) in a workbook, in some of the operations specified by this protocol. |
| SheetName | A string that is used by the protocol server to find a sheet (1) in a workbook in some of the operations specified in this protocol. |
| WorkbookType | An enumeration that is used in the GetWorkbook and SaveWorkbookCopy operations to specify the form of the requested workbook. |

2.2.5.1 CellError

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

The **CellError** type is an enumeration of a set of cell error values that a worksheet cell can contain.

It is defined as follows.

```
<xs:simpleType name="CellError" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Div0"/>
    <xs:enumeration value="NA"/>
    <xs:enumeration value="Name"/>
    <xs:enumeration value="Null"/>
    <xs:enumeration value="Num"/>
    <xs:enumeration value="Ref"/>
    <xs:enumeration value="Value"/>
  </xs:restriction>
</xs:simpleType>
```

The following table specifies the allowable values for the **CellError** simple type.

| Value | Meaning |
|-------|--|
| Div0 | Worksheet cell contains a number that is divided by 0. |
| NA | Worksheet cell contains a value not available to a formula . |
| Name | Worksheet cell contains text not recognized by a formula. |
| Null | Worksheet cell contains an empty intersection of two areas. |
| Num | Worksheet cell contains invalid numeric values in a formula. |
| Ref | Worksheet cell contains a reference that is not valid. |
| Value | Worksheet cell contains a formula that includes worksheet cells that contain different data types. |

2.2.5.2 RangeReference

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

A string that is used by the protocol server to identify a range on a sheet (2) in a workbook, in some of the operations specified by this protocol.

It has the following specification:

- MUST NOT be longer than 32768 characters.
- SHOULD use the A1 reference style as defined in [\[MSFT-ACRR\]](#), but MAY use any format supported by the server to identify the range.

This simple type is defined as follows.

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

2.2.5.3 SheetName

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

A string that is used by the protocol server to find a sheet (1) in a workbook in some of the operations specified in this protocol.

It has the following specification:

- MUST NOT be longer than 31 characters.
- If empty, the protocol server MUST use the first sheet (1) in the workbook unless specified otherwise in the operation.
- If no sheet (1) in the workbook matches the requested **SheetName** simple type, the protocol server MUST return a SOAP fault response, which SHOULD contain the **InvalidSheetName** subcode but MAY contain any other subcode.

If the user does not have the **Open Item permission** for the workbook used by the session (2) , when **SheetName** is used by the **GetCell**, **GetCellA1**, **GetRange**, or **GetRangeA1** operations, the server MUST return a SOAP fault response, which MUST contain the **SpecifiedItemNotFound** subcode, if the matching sheet (1) is not a **visible** sheet (1).

This simple type is defined as follows.

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

2.2.5.4 WorkbookType

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

An enumeration that is used in the **GetWorkbook** and **SaveWorkbookCopy** operations to specify the form of the requested workbook.

This enumeration is defined as follows.

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



```

<xs:restriction base="xs:string">
  <xs:enumeration value="FullWorkbook"/>
  <xs:enumeration value="FullSnapshot"/>
  <xs:enumeration value="PublishedItemsSnapshot"/>
</xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for the **WorkbookType** simple type.

| Value | Meaning |
|------------------------|--|
| FullWorkbook | The protocol server returns the workbook. |
| FullSnapshot | The protocol server returns a snapshot that contains the entire contents of the workbook. |
| PublishedItemsSnapshot | The protocol server returns a published items snapshot . |

2.2.6 Attributes

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

2.2.7 Groups

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

2.2.8 Attribute Groups

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

2.2.9 Common Data Structures

This protocol allows protocol servers to notify protocol clients of application-level faults by 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.

If [\[SOAP1.2/1\]](#) or [\[SOAP1.2/2\]](#) is used for transport, the protocol server MUST use the **Value** element inside the **Subcode** element to pass the error code, as specified in [\[SOAP1.2/1\]](#), Section 5.4.1.3. Expected subcode values are specified in each operation. Common expected subcodes are specified in the [Common List of Subcodes](#).

2.2.9.1 Faults

The following table describes the error codes.

| Value | Meaning |
|------------------|--|
| Blank | MUST NOT be used. |
| InternalError | An unexpected error occurred in the protocol server. |
| NotMemberOfRole | The user does not have permissions to complete the operation. |
| EncryptionTooLow | The protocol server is configured to accept only HTTPS, |

| Value | Meaning |
|-------------------------------------|---|
| | and the protocol client has tried to connect with a non HTTPS. |
| ConfirmDataRefresh | A connected data refresh confirmation is requested by the protocol client before connected data refresh on a workbook. |
| ExternalDataRefreshFailed | Connected data refresh failed. |
| ExternalDataRefreshFailedReal | Connected data refresh failed.<1> |
| ExternalDataRefreshFailedReadOnly | The query of one of the external sources failed (error message for users with read- or view-only permissions). |
| NoAccessToOdcReadOnly | The query of one of the external sources failed due to no access to the ODC file (error message for users with read- or view-only permissions). |
| ErrorConnectingToDataSourceReadOnly | The query of one of the external sources failed due to failure to connect to the data source (error message for users with read- or view-only permissions). |
| InvalidUserNameOrPasswordReadOnly | The query of one of the external sources failed due to invalid credentials (error message for users with read- or view-only permissions). |
| InvalidUserNameOrPassword | The query of one of the external sources failed due to invalid credentials. |
| ExternalDataNotAllowedOnTL | The query of one of the external sources failed because external data connections are not allowed on the Trusted Location. |
| InvalidUriLocalConnectionPath | The query of one of the external sources failed due to a local path for the ODC file. |
| DclNotTrusted | The query of one of the external sources failed because the specified DCL is not trusted. |
| OdcFileDoesNotExist | The query of one of the external sources failed because the specified ODC file does not exist. |
| ErrorAccessingSsoFromEcs | The query of one of the external sources failed because the ECS is unable to access SSO. |
| SsoError | The query of one of the external sources failed due to an SSO error. |
| ErrorConnectingToDataSource | The query of one of the external sources failed due to the inability to connect to the specified data source. |
| UnattendedAccountNotConfigured | The query of one of the external sources failed because the unattended account is not configured. |
| IncorrectAuthUsed | The query of one of the external sources failed due to a credentials delegation error. |
| ExternalDataRefreshFailed_Details | Used by native code to report problems with a |

| Value | Meaning |
|---------------------------------------|--|
| | connection. |
| PivotNoOverlapAllowed | A PivotTable grew to the point where it would overlap another PivotTable. |
| PivotOverlapList | A PivotTable grew to the point where it would overlap a list (2) . |
| PivotCubeNotUnique | An OLAP query found more than one cube of the same name. |
| NoDataItemsOlap | An OLAP query found nothing to display. |
| OlapSchemaFail | An OLAP query was unable to fetch the schema of the cube. |
| InvalidPivot | A PivotTable is invalid. |
| OlapNoMoreDetail | A PivotTable has reached the lowest level of detail. |
| DrillActionInvalid | A PivotTable member cannot be expanded any further. |
| DriverNotOlap | The database does not support OLAP. |
| CubeMissing | The OLAP query was unable to locate the cube in the database. |
| SkipLine | Not all data received from the OLAP query could be displayed. |
| CDimMismatch | The OLAP data provider returned an unexpected number of dimensions (2) . |
| DataSetAxisTooBig | The row (2) or column (2) area of the PivotTable is too large. |
| CubeIntegrityError | There was a problem with the structure of the OLAP cube. |
| PivotNotExpandedtoProtectCellContents | The PivotTable would have filled worksheet cells that already contain values, so the operation was canceled. |
| PivotTruncatedSheetEdge | The PivotTable was truncated by a worksheet edge. |
| DdlGrpNotSup | The PivotTable operation is not supported by the OLAP data provider. |
| PivotFormulasCollided | Some calculated PivotTable members were removed from the PivotTable because new PivotTable members of the same name appeared during refresh. |
| NoRetItems | The query of the PivotTable page field did not return any items. |
| OlapConnectVisualModeProvider | The PivotTable operation is not supported by the OLAP data provider. |
| PivotNoSourceData | The PivotTable was saved without the underlying data. |

| Value | Meaning |
|---|--|
| PivotFilteringError | The PivotTable could not be filtered. |
| PivotTableIncorrectDateFormat | An invalid date was used to filter the PivotTable. |
| PivotTableEndingLargerThanStarting | An invalid pair of starting and ending values was used to filter the PivotTable. |
| PivotTableIncorrectNumberFormat | A non-numeric value was used to filter the PivotTable, when a numeric value was required. |
| PivotTableOutOfRange | An out of range value was used to filter the PivotTable. |
| MergedCellOverlap | The PivotTable could not grow over part of a merged cell . |
| PivotSourceNotAvailable | The PivotTable could not be refreshed because its source is missing. |
| UnexpectedPivotError | The PivotTable has encountered an unexpected error. |
| EditModeCannotAcquireLockUnknownReason | An attempt to acquire or refresh a lock on a file failed. |
| EditModeCannotAcquireLock | An attempt to acquire or refresh a lock on a file failed because the file is already locked . |
| EditModeCannotAcquireLockUnknownUser | An attempt to acquire or refresh a lock on a file failed because the file is locked by an unknown user. |
| EditModeWorkbookCheckedOutRequired | An attempt to acquire or refresh a lock on a file failed because the file is not checked out . |
| EditModeWorkbookCheckedOutToLocalDrafts | An attempt to acquire or refresh a lock on a file failed because the file is checked out to a local drafts folder. |
| SiteTooBusy | MUST NOT be used. |
| SiteCollectionMemoryQuotaExceeded | MUST NOT be used. |
| SessionMemoryQuotaExceeded | MUST NOT be used. |
| SiteCollectionChartQuotaExceeded | MUST NOT be used. |
| RequestLimitsExceeded | Too many cells requested. |
| SavingUnsupportedInViewMode | An attempt was made to save a workbook from a view session . |
| SavingWorkbookNoWorkbookName | An attempt was made to save a workbook from a new workbook session . |
| FileSaveGeneric | The save operation failed. |
| SavingWorkbookInvalidPermissions | The user does not have permissions to save to the specified location. |
| SavingWorkbookMaximumSizeExceeded | An attempt to save a workbook failed because the workbook exceeds the maximum permitted size. |
| SavingWorkbookLockedOrCheckedOut | An attempt to save a workbook failed because the file is |

| Value | Meaning |
|---|--|
| | locked or checked out. |
| SavingWorkbookNotCheckedOut | An attempt to save a workbook failed because the file is not checked out. |
| SavingWorkbookAccessDenied | An attempt to save a workbook failed because the user does not have permission to overwrite an existing file. |
| SavingWorkbookInvalidFileType | MUST NOT be used. |
| SavingWorkbookFileNameExists | An attempt to save a workbook failed because the file already exists and the option to overwrite existing files was not specified. |
| SavingWorkbookNotTrustedLocation | An attempt to save a workbook failed because of the security settings on the protocol server. |
| SavingWorkbookInvalidLocation | An attempt to save a workbook failed because the specified file location is invalid. |
| SavingWorkbookInvalidFileName | An attempt to save a workbook failed because the specified file name is invalid. |
| SavingWorkbookTrustedLocationSettingWarning | An attempt to save a workbook failed because of the security settings on the protocol server. |
| SavingWorkbookAlmostTooLarge | MUST NOT be used. |
| SavingWorkbookTooLarge | MUST NOT be used. |
| NewWorkbookInvalidLocation | An attempt to create a new workbook failed because the specified location is invalid. |
| FileOpenSecuritySettings | Cannot open workbook because of security settings. |
| FileOpenNotFound | The specified workbook file does not exist. |
| FileOpenAccessDenied | The user does not have access to the workbook file. |
| FileCorrupt | The workbook file is in a format not recognized by the protocol server. |
| OpenWorkbookTooLarge | The protocol server failed to create a session (2) for a workbook because it is too large. |
| OpenWorkbookLocationNotTrusted | The protocol server failed to create a session (2) for a workbook because of the security settings on the protocol server. |
| OpenWorkbookAccessDenied | The protocol server failed to create a session (2) for a workbook because the user does not have sufficient permissions. |
| WorkbookNotSupported | The protocol server failed to create a session (2) for this workbook because it has unsupported content. |
| WorkbookNotSupportedViewRight | The protocol server failed to create a session (2) for this workbook because it has unsupported content and the user does not have Open Item permission. |

| Value | Meaning |
|---|--|
| UnsupportedFeatureViewWarning | MUST NOT be used. |
| UnsupportedFeatureSaveCopyWarning | MUST NOT be used. |
| ExternalWorkbookLinksDisabled | MUST NOT be used. |
| WorkbookLockedRead | An attempt to open a workbook failed because a file lock prevents it from being read. |
| EditModeCannotEditNonSharePointWorkbook | An attempt to open an edit session failed because the file is not in a document library . |
| EditModeAccessDenied | An attempt to open a workbook in an edit session failed because the user does not have sufficient permissions to edit the workbook file . |
| GenericFileOpenError | An unspecified error occurred when opening the workbook file. |
| IRMedWorkbook | An attempt to open a workbook failed because the workbook is in an Information Rights Management (IRM) document library. |
| ParameterInvalid | An attempt to open a workbook failed because the workbook contains one or more invalid workbook parameters . |
| ParameterLimitExceeded | An attempt to open a workbook failed because the workbook contains too many workbook parameters. |
| WorkbookInternalLimit | The workbook cannot be loaded by the protocol server because of a file-size limitation. |
| OldFormatConversionFailed | MUST NOT be used. |
| EditModeWorkbookExternalData | An attempt to open a workbook as an edit session failed because it contains connected data or user-defined functions (UDFs) , and the workbook file has not been checked out. |
| FormulaEditingNotEnabled | The operation failed because it tried to set a formula into a workbook parameter. |
| DimensionAndArrayMismatch | An attempt to set a range failed because the dimensions (2) of the specified array of cells do not match the specified range. |
| SetRangeFailure | The operation failed to set a value into a cell. |
| ObjectTypeNotSupported | The operation failed because it tried to update a cell with an object of an unsupported type. |
| ParameterDoesNotExist | The specified workbook parameter does not exist. |
| FormulaParseFailed | The specified formula could not be parsed. |
| SetParameterFailure | An unspecified error occurred when attempting to set a workbook parameter. |

| Value | Meaning |
|--------------------------------|---|
| CannotEditPivotTable | The operation failed because it tried to set a cell in a PivotTable. |
| InsDelCellsNonblankCells | The operation failed because it would require shifting non-blank cells off the sheet (1). |
| InsDelCellsArrayFmla | The operation failed because it tried to change part of an array formula . |
| InsDelCellsPivotTable | The operation failed because it tried to change part of a PivotTable. |
| InsDelCellsTable | The operation failed because it tried to change part of a table . |
| InsDelCellsMergedCells | The operation failed because it tried to change part of a merged cell. |
| InsDelCellsDatatable | The operation failed because it tried to change part of a data table (1) . |
| InsDelCellsRowsForWholeColumn | The operation failed because it tried to insert or delete a whole row (2), but the specified range is a whole column (2). |
| InsDelCellsColumnsForWholeRow | The operation failed because it tried to insert or delete a whole column (2), but the specified range is a whole row (2). |
| InvalidSheetName | The operation failed because the specified sheet (1) name does not exist. |
| RangeParseError | The operation failed because the specified range is invalid. |
| SpecifiedRangeNotFound | The operation failed because the specified range was not found. |
| RangeDefinitionError | The operation failed because the specified range is not a valid sheet (2) location. |
| DiscontiguousRangeNotSupported | The operation failed because a non-contiguous range was specified. |
| RangeRequestAreaExceeded | The requested range was too large. |
| SheetRangeMismatch | The sheet (2) provided as the sheet (2) argument is not the same as the sheet (2) specified in the range argument. |
| ImageGenFailed | The protocol server failed to create an image. |
| ApiInvalidArgument | The operation failed because an argument that passed was invalid. |
| ApiInvalidCoordinate | MUST NOT be used. |
| ConfirmationUnsupported | MUST NOT be used. |

| Value | Meaning |
|---|---|
| ApiInvalidDimensionReturned | MUST NOT be used. |
| ApiMissingSiteContext | MUST NOT be used. |
| Ewr_IllegalPageSize | MUST NOT be used. |
| Ewr_WorkbookNotSpecified | MUST NOT be used. |
| Ewr_LoadProgress | MUST NOT be used. |
| Ewr_FileCorrupt | MUST NOT be used. |
| Ewr_WorkbookNotSupported | MUST NOT be used. |
| Ewr_GenericFileOpenError | MUST NOT be used. |
| Ewr_InvalidParameterGeneric | MUST NOT be used. |
| Ewr_InvalidParameterMandatoryMissing | MUST NOT be used. |
| Ewr_InvalidParameterUnexpectedFormat | MUST NOT be used. |
| Ewr_InvalidParameterOutOfRange | MUST NOT be used. |
| Ewr_InvalidParameterOperationNotAllowed | MUST NOT be used. |
| Ewr_ParameterInvalid | MUST NOT be used. |
| Ewr_ParameterLimitExceeded | MUST NOT be used. |
| Ewr_ParameterDoesNotExist | MUST NOT be used. |
| Ewr_SetParameterFailure | MUST NOT be used. |
| Ewr_SetParameterFailureOnOpen | MUST NOT be used. |
| Ewr_ParameterReadFromDocPropsError | MUST NOT be used. |
| Ewr_FormulaEditingNotEnabled | MUST NOT be used. |
| Ewr_InvalidVisibleItem | MUST NOT be used. |
| Ewr_EditSaveAsOverCurrentWorkbook | MUST NOT be used. |
| Ewr_RequestTimedOut | MUST NOT be used. |
| Ewr_ServerCommunicationError | MUST NOT be used. |
| Ewr_ServerAuthenticationError | MUST NOT be used. |
| Mewa_PageStateInitFailed | MUST NOT be used. |
| Mewa_CriticalParameterMissing | MUST NOT be used. |
| Mewa_GridViewRenderingFailed | MUST NOT be used. |
| Mewa_FeatureNotActive | MUST NOT be used. |
| OperationNotSupportedInViewOnly | The operation failed because it is not supported in a |

| Value | Meaning |
|---------------------------------|---|
| | view session. |
| OperationOnInvalidState | An attempt was made to open a session (2) that already has a workbook assigned. |
| InvalidOrTimedOutSession | The operation failed because the session (2) is invalid or timed out. |
| ScratchPadChangesLost | A view session (2) that is not in view-only mode timed out while the user was attempting to convert it to an edit session. |
| MaxSessionsPerUserExceeded | The user has exceeded the maximum allowed session (2) count by the protocol server. |
| OperationCanceled | The user canceled a request that was being processed by a session (2). |
| MaxRequestDurationExceeded | The request failed because it took too long to process. |
| NoOccurrencesFound | The protocol server could not match the sought string to any worksheet cell relevant to the search. |
| EcsInvalidNamedObject | An invalid named object was requested. |
| NoPublishedItems | The workbook does not have any published items . |
| RetryError | The operation could not be finished at this time, but the request may succeed if it is sent again by the protocol client. |
| StopOpenOnRefreshFailure | An attempt to open a workbook failed because connected data refresh failed. |
| ProductExpired | The protocol server returns this error when the application license key is invalid. |
| SortActionInvalid | Sorting an AutoFilter failed because it contains merged cells that are not all the same size. |
| SortActionNotSupportedWithPivot | Sorting an AutoFilter failed because it contains a PivotTable. |
| CircularReferencesFound | MUST NOT be used. |
| SessionNotFound | MUST NOT be used. |
| NewWorkbookSessionTimeout | A new workbook session has timed out. |
| EditXLSFile | MUST NOT be used. |
| UnsupportedFeatureEditWarning | Edit session is not supported for this workbook. |
| EditModeSessionExists | The view session could not be saved because the workbook is open in an edit session. |
| NewerWorkbookExists | The view session could not be saved because the workbook has been modified since the session (2) was opened. |

| Value | Meaning |
|------------------------------------|---|
| ListEntryNotFound_1 | MUST NOT be used. |
| ListEntryNotFound_2 | MUST NOT be used. |
| ListEntryAlreadyExists_1 | MUST NOT be used. |
| ListEntryAlreadyExists_2 | MUST NOT be used. |
| InvalidArgumentValue | MUST NOT be used. |
| MissingArgumentName | MUST NOT be used. |
| InvalidArgumentName | MUST NOT be used. |
| ArgumentBadInRemove | MUST NOT be used. |
| ManagementUIListEntryAlreadyExists | MUST NOT be used. |
| ManagementUIListEntryNotFound | MUST NOT be used. |
| ManagementUIInternalError | MUST NOT be used. |
| UnsupportedUICulture | The UI culture is not supported by the protocol server. |
| UnsupportedDataCulture | The data culture is not supported by the protocol server. |
| SpecifiedItemNotFound | An attempt to set a range of cells in a workbook failed because the user does not have Open Item permission. |
| EcsSnapshotAccessDenied | The protocol server cannot return the workbook or a snapshot, because the user does not have Open Item permission. |
| SaveFailed | A save operation failed. |
| WrongEcs | An attempt was made to access an edit session that is owned by another instance of a protocol server in the farm . |
| PasteDataLimitHit | The string exceeds the maximum permitted length. |
| UndoRedoFail | MUST NOT be used. |
| CollabLocaleMismatch | The user joined an edit session whose data culture does not match the user's data culture. |
| LocaleRestored | The data culture of an edit session has been restored to the user's data culture. |
| InvalidEditSession | The edit session is invalid. |
| InvalidEditSessionId | MUST NOT be used. |
| UnableToLoadEditSession | Unable to load an edit session. |
| SavingWorkbookFailed | Failed to save workbook. |
| RefreshLockFailed | An attempt to refresh a lock on a workbook file failed. |

| Value | Meaning |
|-------------------------|--------------------------------------|
| InvalidChartName | An invalid chart name was specified. |
| RetryOutOfSync | The PivotTable operation failed. |
| OperationFailed | A workbook operation failed. |
| OpenInExcelActiveXError | MUST NOT be used. |
| Count | MUST NOT be used. |

2.2.9.2 Common List of Subcodes

In the event of an application error during an operation the protocol server returns a SOAP fault response, which SHOULD [<2>](#) contain one of the following subcodes:

| Subcode |
|--|
| InternalError |
| OperationFailed |
| NotMemberOfRole |
| EncryptionTooLow |
| ConfirmDataRefresh |
| OperationOnInvalidState |
| InvalidOrTimedOutSession |
| MaxSessionsPerUserExceeded |
| OperationCanceled |
| MaxRequestDurationExceeded |
| RetryError |
| ProductExpired |
| NewWorkbookSessionTimeout |
| SpecifiedItemNotFound |
| CollabLocaleMismatch |
| InvalidEditSession |
| UnableToLoadEditSession |
| SpecifiedRangeNotFound |
| EditModeAccessDenied |
| EditModeCannotAcquireLockUnknownReason |
| RequestLimitsExceeded |

| Subcode |
|----------------------|
| SavingWorkbookFailed |
| RefreshLockFailed |
| ApiInvalidArgument |

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**.

The client side of this protocol simply passes calls through, and no additional timers or other state is required. Calls made by the higher-layer protocol or application are passed directly to the transport, and the results returned by the transport are passed directly back to the higher-layer protocol or application.

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

This protocol allows protocol servers to perform implementation-specific localization of text in various messages. Except where specified, the localization of this text is an implementation-specific behavior of the protocol server and not significant for interoperability.

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

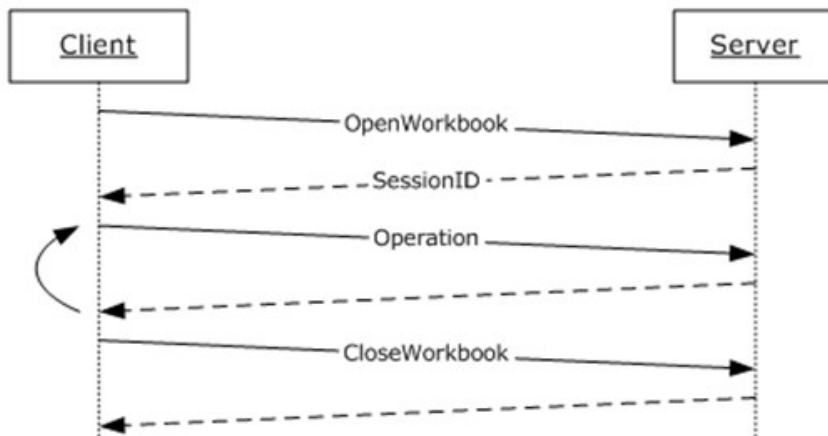


Figure 2: High-level sequence diagram

Session Behavior

View Sessions

A view session is created on a successful **OpenWorkbook** or **OpenWorkbookEx** request. A view session stops responding to operations after a successful call to the **CloseWorkbook** operation or after the view session has timed out.

New Workbook Sessions

A new workbook session is created on a successful **NewWorkbook** request. A new workbook session stops responding to operations after a successful call to the **CloseWorkbook** operation or after the new workbook session has timed out.

Edit Sessions

An edit session is created on a successful **OpenWorkbookForEditing** request, unless one already exists for the same **workbookPath**, as specified in [OpenWorkbookForEditing](#) (section 3.1.4.21). A new participant is added to the edit session on a successful **OpenWorkbookForEditing** request.

A participant that makes a call to **CloseWorkbook** is removed from the edit session if the operation completes successfully. An edit session stops responding to operations when all of its participants are removed.

Request Behavior

A request is created when a **Web service** operation starts running on the server, and ends when the Web service operation finishes, when it is successfully cancelled by a **CancelRequest** call, or after the request has timed out.

3.1 Server Details

3.1.1 Abstract Data Model

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

Session: An entity that gives context to operations done against a workbook. A session (2) is associated with a workbook state, which can be modified by state changing requests.

Request: An entity that represents a running request on the protocol server.

3.1.2 Timers

3.1.2.1 Session Timeout

The protocol server MAY<3> time sessions (2) out. When a session (2) times out, the protocol server MUST NOT serve any requests to the timed out session (2).

3.1.2.2 Request Timeout

The protocol server MAY<4> time requests out. When a request times out, the protocol server SHOULD<5> return a SOAP fault response, with the subcode **MaxRequestDurationExceeded**.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

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

| Operation | Description |
|-----------|---|
| Calculate | This operation is used to calculate a specific range in the workbook. |

| Operation | Description |
|-------------------------------|---|
| CalculateA1 | This operation is used to calculate a specific range in the workbook. |
| CalculateWorkbook | This operation is used to calculate the workbook. |
| CancelRequest | This operation is used to cancel an operation that is currently running on the protocol server. |
| CloseWorkbook | This operation is used to close an existing session (2) or leave an edit session on the protocol server. |
| EnsureWorkbookModel | Loads workbook data model and returns information about the model. |
| GetApiVersion | This operation is used to get the version of the protocol server. |
| GetCell | This operation is used to retrieve the contents of a specific cell in the workbook. |
| GetCellA1 | This operation is used to retrieve the contents of a specific cell in the workbook. |
| GetChartImageUrl | This operation returns the URL of a chart in a workbook. |
| GetPublishedItemNames | This operation returns a list of the published items in a workbook. |
| GetRange | This operation is used to retrieve the contents of a specific range in the workbook. |
| GetRangeA1 | This operation is used to retrieve the contents of a named range in the workbook. |
| GetSessionInformation | This operation retrieves information about a session (2) on the protocol server. |
| GetSheetNames | The operation returns information about the sheets in a workbook. |
| GetWorkbook | This operation is used to retrieve a workbook file. |
| GetWorkbookConnections | Returns a list of data connection (2) names in the workbook. |
| NewWorkbook | This operation is used to create a new workbook session. |
| OpenWorkbook | This operation is used to create a new view session, giving access to a workbook on the protocol server. |
| OpenWorkbookEx | This operation is used to create a new view session, giving access to a workbook on the protocol server. |
| OpenWorkbookForEditing | This operation is used to create a new edit session, giving access to a workbook on the protocol server. |
| Refresh | This operation is used to refresh connected data on a workbook. |
| RefreshEx | This operation is used to refresh connected data on a workbook. |
| SaveWorkbook | This operation is used to save a workbook associated with an open session (2) to the same location from which it was opened. |
| SaveWorkbookCopy | This operation is used to save a workbook associated with an open session (2) to a specified location, and in a specified format. |

| Operation | Description |
|------------------------------|---|
| SetCalculationOptions | This operation is used to set the workbook calculation mode . |
| SetCell | This operation is used to set a value to a specific worksheet cell in the workbook. |
| SetCellA1 | This operation is used to set a value to a specific worksheet cell in the workbook. |
| SetParameters | This operation to sets values of workbook parameters in a workbook. |
| SetRange | This operation is used to set values to a range of worksheet cells on a workbook. |
| SetRangeA1 | This operation is used to set values to a range of worksheet cells on a workbook. |

3.1.4.1 Calculate

This operation is used to calculate a specific range in the workbook.

The following is the WSDL port type specification of the **Calculate WSDL operation**.

```
<wsdl:operation name="Calculate" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:CalculateSoapIn"/>
  <wsdl:output message="tns:CalculateSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **CalculateSoapIn** request message, and the protocol server responds with a **CalculateSoapOut** response message as follows:

1. If the protocol determines that the workbook does not need to be calculated, it MUST succeed the operation and it MUST ignore the **sheetName** and **rangeCoordinates** parameters.
2. **sheetName** MUST NOT be empty if **rangeCoordinates** is not empty, as defined in [RangeCoordinates](#). If **sheetName** is empty, the whole workbook MUST be calculated.
3. If **rangeCoordinates** is empty, as defined in section [2.2.4.4](#), and the **sheetName** is not empty, the whole sheet (2), which was specified in **sheetName**, MUST be calculated.
4. If **rangeCoordinates** is not empty, the specified range MUST be calculated.
5. If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|---------------------------------------|
| DiscontiguousRangeNotSupported |
| EcsInvalidNamedObject |
| InvalidSheetName |
| RangeDefinitionError |

| |
|---------------------------------|
| Subcode |
| RangeParseError |
| RangeRequestAreaExceeded |
| SheetRangeMismatch |

3.1.4.1.1 Messages

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

| Message | Description |
|-------------------------|--|
| CalculateSoapIn | The request WSDL message for the Calculate WSDL operation. |
| CalculateSoapOut | The response WSDL message for the Calculate WSDL operation. |

3.1.4.1.1.1 CalculateSoapIn

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

The **SOAP action** value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/Calculate
```

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

3.1.4.1.1.2 CalculateSoapOut

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

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

3.1.4.1.2 Elements

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

| Element | Description |
|--------------------------|--|
| Calculate | The input data for the Calculate WSDL operation. |
| CalculateResponse | The result data for the Calculate WSDL operation. |

3.1.4.1.2.1 Calculate

The **Calculate** element specifies the input data for the **Calculate** WSDL operation.

```
<xs:element name="Calculate" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```

        <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="rangeCoordinates"
type="tns:RangeCoordinates"/>
    </xs:sequence>
</xs:complexType>
</xs:element>

```

sessionId: A **session identifier (2)**. MUST be present in the request message. MUST NOT be longer than 1023 characters. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

sheetName: The name of the sheet (2) to calculate. MUST be present. MUST NOT be longer than 31 characters.

rangeCoordinates: The range to calculate on the specified sheet (2). MUST be present.

3.1.4.1.2.2 CalculateResponse

The **CalculateResponse** element specifies the result data for the **Calculate** WSDL operation.

```

<xs:element name="CalculateResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

status: An array of status information as specified in **ArrayOfStatus** (section [2.2.4.2](#)).

3.1.4.1.3 Complex Types

None.

3.1.4.1.4 Simple Types

None.

3.1.4.1.5 Attributes

None.

3.1.4.1.6 Groups

None.

3.1.4.1.7 Attribute Groups

None.

3.1.4.2 CalculateA1

This operation is used to calculate a specific range in the workbook.

The following is the WSDL port type specification of the **CalculateA1** WSDL operation.

```
<wsdl:operation name="CalculateA1" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:CalculateA1SoapIn"/>
  <wsdl:output message="tns:CalculateA1SoapOut"/>
</wsdl:operation>
```

The protocol client sends a **CalculateA1SoapIn** request message, and the protocol server responds with a **CalculateA1SoapOut** response message, as follows:

1. If the protocol determines that the workbook does not need to be calculated, it MUST succeed the operation and it MUST ignore the **sheetName** and **rangeName** parameters.
2. **sheetName** MUST be present and MUST NOT be empty, if **rangeName** is not empty. If **sheetName** is not present or is empty, the whole workbook MUST be calculated.
3. **rangeName** MUST be present. If **rangeName** is empty, and **sheetName** is present and not empty, the whole sheet (2), which was specified in **sheetName**, MUST be calculated.
4. If **rangeName** is not empty, the specified range MUST be calculated.
5. If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|---------------------------------------|
| DiscontiguousRangeNotSupported |
| EcsInvalidNamedObject |
| InvalidSheetName |
| RangeDefinitionError |
| RangeParseError |
| RangeRequestAreaExceeded |
| SheetRangeMismatch |

3.1.4.2.1 Messages

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

| Message | Description |
|---------------------------|--|
| CalculateA1SoapIn | The request WSDL message for the CalculateA1 WSDL operation. |
| CalculateA1SoapOut | The response WSDL message for the CalculateA1 WSDL operation. |

3.1.4.2.1.1 CalculateA1SoapIn

The request WSDL message for the **CalculateA1** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/CalculateA1
```

The SOAP body contains the **CalculateA1** element.

3.1.4.2.1.2 CalculateA1SoapOut

The response WSDL message for the **CalculateA1** WSDL operation.

The SOAP body contains the **CalculateA1Response** element.

3.1.4.2.2 Elements

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

| Element | Description |
|----------------------------|--|
| CalculateA1 | The input data for the CalculateA1 WSDL operation. |
| CalculateA1Response | The result data for the CalculateA1 WSDL operation. |

3.1.4.2.2.1 CalculateA1

The **CalculateA1** element specifies the input data for the **CalculateA1** WSDL operation.

```
<xs:element name="CalculateA1" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>  
      <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>  
      <xs:element minOccurs="0" maxOccurs="1" name="rangeName" type="xs:string"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

sessionId: A session (2) identifier. MUST be present in the request message. MUST NOT be longer than 1023 characters. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

sheetName: The name of the sheet (2) to calculate. MUST be present and MUST NOT be empty, if **rangeName** is not empty. MUST NOT be longer than 31 characters.

rangeName: The name of the range to calculate. MUST be present. MUST NOT be longer than 255 characters. See section [2.2.5.2](#).

3.1.4.2.2.2 CalculateA1Response

The **CalculateA1Response** element specifies the result data for the **CalculateA1** WSDL operation.

```
<xs:element name="CalculateA1Response" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>
```

```

    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.2.3 Complex Types

None.

3.1.4.2.4 Simple Types

None.

3.1.4.2.5 Attributes

None.

3.1.4.2.6 Groups

None.

3.1.4.2.7 Attribute Groups

None.

3.1.4.3 CalculateWorkbook

This operation is used to calculate the workbook.

The following is the WSDL port type specification of the **CalculateWorkbook** WSDL operation.

```

<wsdl:operation name="CalculateWorkbook" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:CalculateWorkbookSoapIn"/>
  <wsdl:output message="tns:CalculateWorkbookSoapOut"/>
</wsdl:operation>

```

The protocol client sends a **CalculateWorkbookSoapIn** request message and the protocol server responds with a **CalculateWorkbookSoapOut** response message as follows:

If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|---------------------------------------|
| DiscontiguousRangeNotSupported |
| EcsInvalidNamedObject |
| InvalidSheetName |

| |
|---------------------------------|
| Subcode |
| RangeDefinitionError |
| RangeParseError |
| RangeRequestAreaExceeded |
| SheetRangeMismatch |

3.1.4.3.1 Messages

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

| Message | Description |
|---------------------------------|--|
| CalculateWorkbookSoapIn | The request WSDL message for the CalculateWorkbook WSDL operation. |
| CalculateWorkbookSoapOut | The response WSDL message for the CalculateWorkbook WSDL operation. |

3.1.4.3.1.1 CalculateWorkbookSoapIn

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

The SOAP action value is:

`http://schemas.microsoft.com/office/excel/server/webservices/CalculateWorkbook`

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

3.1.4.3.1.2 CalculateWorkbookSoapOut

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

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

3.1.4.3.2 Elements

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

| Element | Description |
|----------------------------------|--|
| CalculateWorkbook | The input data for the CalculateWorkbook WSDL operation. |
| CalculateWorkbookResponse | The result data for the CalculateWorkbook WSDL operation. |

3.1.4.3.2.1 CalculateWorkbook

The **CalculateWorkbook** element specifies the input data for the **CalculateWorkbook** WSDL operation.

```

<xs:element name="CalculateWorkbook" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="calculateType" type="tns:CalculateType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

sessionId: A session (2) identifier. MUST be present in the request message. MUST NOT be longer than 1023 characters. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

calculateType: The type of calculation to perform, as defined in section [3.1.4.3.4.1](#).

3.1.4.3.2 CalculateWorkbookResponse

The **CalculateWorkbookResponse** element specifies the result data for the **CalculateWorkbook** WSDL operation.

```

<xs:element name="CalculateWorkbookResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.3.3 Complex Types

None.

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 |
|----------------------|---|
| CalculateType | The type of calculation to be executed. |

3.1.4.3.4.1 CalculateType

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

The type of calculation to be executed.

```

<xs:simpleType name="CalculateType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Recalculate"/>
    <xs:enumeration value="CalculateFull"/>
  </xs:restriction>
</xs:simpleType>

```

```
</xs:restriction>
</xs:simpleType>
```

The following table specifies the allowable values for the **CalculateType** simple type.

| Value | Meaning |
|---------------|---|
| Recalculate | The protocol server MUST calculate all formulas in the workbook that need to be calculated. |
| CalculateFull | The protocol server MUST calculate all formulas. |

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 CancelRequest

This operation is used to cancel an operation that is currently running on the protocol server.

The following is the WSDL port type specification of the **CancelRequest** WSDL operation.

```
<wsdl:operation name="CancelRequest" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:CancelRequestSoapIn"/>
  <wsdl:output message="tns:CancelRequestSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **CancelRequestSoapIn** request message, and the protocol server responds with a **CancelRequestSoapOut** response message, as follows:

- The protocol server SHOULD try to cancel all requests currently running on the protocol server against the session (2) specified in **sessionId**.
- The following MUST NOT cause an application error in the **CancelRequest** operation:
 - An operation is not currently running on the session (2).
 - A currently running operation on the session (2) cannot be cancelled.
- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

3.1.4.4.1 Messages

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

| Message | Description |
|-----------------------------|--|
| CancelRequestSoapIn | The request WSDL message for the CancelRequest WSDL operation. |
| CancelRequestSoapOut | The response WSDL message for the CancelRequest WSDL operation. |

3.1.4.4.1.1 CancelRequestSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/CancelRequest
```

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

3.1.4.4.1.2 CancelRequestSoapOut

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

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

3.1.4.4.2 Elements

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

| Element | Description |
|------------------------------|--|
| CancelRequest | The input data for the CancelRequest WSDL operation. |
| CancelRequestResponse | The result data for the CancelRequest WSDL operation. |

3.1.4.4.2.1 CancelRequest

The **CancelRequest** element specifies the input data for the **CancelRequest** WSDL operation.

```
<xs:element name="CancelRequest" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

sessionId: A session identifier (2) MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**,

OpenWorkbookForEditing, or **NewWorkbook** operations, or but MAY instead be obtained in any other way.

3.1.4.4.2 CancelRequestResponse

The **CancelRequestResponse** element specifies the result data for the **CancelRequest** WSDL operation.

```
<xs:element name="CancelRequestResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

status: An array of status information as specified in **ArrayOfStatus** (section [2.2.4.2](#)).

3.1.4.4.3 Complex Types

None.

3.1.4.4.4 Simple Types

None.

3.1.4.4.5 Attributes

None.

3.1.4.4.6 Groups

None.

3.1.4.4.7 Attribute Groups

None.

3.1.4.5 CloseWorkbook

This operation is used to close an existing session (2) or leave an edit session on the protocol server.

The following is the WSDL port type specification of the **CloseWorkbook** WSDL operation.

```
<wsdl:operation name="CloseWorkbook" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:CloseWorkbookSoapIn"/>
  <wsdl:output message="tns:CloseWorkbookSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **CloseWorkbookSoapIn** request message and the protocol server responds with a **CloseWorkbookSoapOut** response message as follows:

If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

3.1.4.5.1 Messages

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

| Message | Description |
|-----------------------------|--|
| CloseWorkbookSoapIn | The request WSDL message for the CloseWorkbook WSDL operation. |
| CloseWorkbookSoapOut | The response WSDL message for the CloseWorkbook WSDL operation. |

3.1.4.5.1.1 CloseWorkbookSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/CloseWorkbook
```

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

3.1.4.5.1.2 CloseWorkbookSoapOut

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

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

3.1.4.5.2 Elements

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

| Element | Description |
|------------------------------|--|
| CloseWorkbook | The input data for the CloseWorkbook WSDL operation. |
| CloseWorkbookResponse | The result data for the CloseWorkbook WSDL operation. |

3.1.4.5.2.1 CloseWorkbook

The **CloseWorkbook** element specifies the input data for the **CloseWorkbook** WSDL operation.

```
<xs:element name="CloseWorkbook" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operations, or but MAY instead be obtained in any other way.

3.1.4.5.2.2 CloseWorkbookResponse

The **CloseWorkbookResponse** element specifies the result data for the **CloseWorkbook** WSDL operation.

```
<xs:element name="CloseWorkbookResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

status: MUST NOT be present.

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.

3.1.4.6 EnsureWorkbookModel

Loads workbook data model and returns information about the model.

The following is the WSDL port type specification of the **EnsureWorkbookModel** WSDL operation.

```
<wsdl:operation name="EnsureWorkbookModel" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:EnsureWorkbookModelSoapIn"/>
  <wsdl:output message="tns:EnsureWorkbookModelSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **EnsureWorkbookModelSoapIn** request message, and the protocol server responds with a **EnsureWorkbookModelSoapOut** response message as follows:

- The protocol server loads workbook data model and returns information about the loaded model in the **WorkbookModelInfo** element.
- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

3.1.4.6.1 Messages

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

| Message | Description |
|-----------------------------------|--|
| EnsureWorkbookModelSoapIn | The request WSDL message for the EnsureWorkbookModel WSDL operation. |
| EnsureWorkbookModelSoapOut | The response WSDL message for the EnsureWorkbookModel WSDL operation. |

3.1.4.6.1.1 EnsureWorkbookModelSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/EnsureWorkbookModel
```

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

3.1.4.6.1.2 EnsureWorkbookModelSoapOut

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

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

3.1.4.6.2 Elements

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

| Element | Description |
|------------------------------------|--|
| EnsureWorkbookModel | The input data for the EnsureWorkbookModel WSDL operation. |
| EnsureWorkbookModelResponse | The result data for the EnsureWorkbookModel WSDL operation. |

3.1.4.6.2.1 EnsureWorkbookModel

The **EnsureWorkbookModel** element specifies the input data for the **EnsureWorkbookModel** WSDL operation.

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

```

    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

sessionId: A session identifier (2). MUST be present. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operations, but MAY instead be obtained in any other way.

3.1.4.6.2.2 EnsureWorkbookModelResponse

The **EnsureWorkbookModelResponse** element specifies the result data for the **EnsureWorkbookModel** WSDL operation.

```

<xs:element name="EnsureWorkbookModelResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="EnsureWorkbookModelResult"
type="tns:WorkbookModelInfo"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

EnsureWorkbookModelResult: WorkbookModelInfo element as specified in section [3.1.4.6.3.1](#)

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.6.3 Complex Types

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

| Complex type | Description |
|--------------------------|--|
| WorkbookModelInfo | Specifies server, database, version of the data model, and session timeout (section 3.1.2.1). |

3.1.4.6.3.1 WorkbookModelInfo

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

Specifies **server**, **database**, **version** of the data model, and session timeout (section [3.1.2.1](#)).

```

<xs:complexType name="WorkbookModelInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Server" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Database" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="SessionInteractivityTimeout"
type="xs:int"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Version" type="xs:int"/>
  </xs:sequence>

```

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

Server: The server where the data model was loaded.

Database: The database on the server where the data model was loaded.

SessionInteractivityTimeout: Session timeout (section [3.1.2.1](#)).

Version: Version of the data model.

3.1.4.6.4 Simple Types

None.

3.1.4.6.5 Attributes

None.

3.1.4.6.6 Groups

None.

3.1.4.6.7 Attribute Groups

None.

3.1.4.7 GetApiVersion

This operation is used to get the version of the protocol server.

The following is the WSDL port type specification of the **GetApiVersion** WSDL operation.

```
<wsdl:operation name="GetApiVersion" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:GetApiVersionSoapIn"/>
  <wsdl:output message="tns:GetApiVersionSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **GetApiVersionSoapIn** request message, and the protocol server responds with a **GetApiVersionSoapOut** response message as follows:

- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

3.1.4.7.1 Messages

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

| Message | Description |
|-----------------------------|--|
| GetApiVersionSoapIn | The request WSDL message for the GetApiVersion WSDL operation. |
| GetApiVersionSoapOut | The response WSDL message for the GetApiVersion WSDL operation. |

3.1.4.7.1.1 GetApiVersionSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/GetApiVersion
```

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

3.1.4.7.1.2 GetApiVersionSoapOut

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

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

3.1.4.7.2 Elements

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

| Element | Description |
|------------------------------|--|
| GetApiVersion | The input data for the GetApiVersion WSDL operation. |
| GetApiVersionResponse | The result data for the GetApiVersion WSDL operation. |

3.1.4.7.2.1 GetApiVersion

The **GetApiVersion** element specifies the input data for the **GetApiVersion** WSDL operation.

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

3.1.4.7.2.2 GetApiVersionResponse

The **GetApiVersionResponse** element specifies the result data for the **GetApiVersion** WSDL operation.

```
<xs:element name="GetApiVersionResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" maxOccurs="1" name="GetApiVersionResult" type="xs:string"/>  
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

GetApiVersionResult: The version of the server. MUST be present.

status: MUST NOT be present.

3.1.4.7.3 Complex Types

None.

3.1.4.7.4 Simple Types

None.

3.1.4.7.5 Attributes

None.

3.1.4.7.6 Groups

None.

3.1.4.7.7 Attribute Groups

None.

3.1.4.8 GetCell

This operation is used to retrieve the contents of a specific cell in the workbook.

The following is the WSDL port type specification of the **GetCell** WSDL operation.

```
<wsdl:operation name="GetCell" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:GetCellSoapIn"/>
  <wsdl:output message="tns:GetCellSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **GetCellSoapIn** request message, and the protocol server responds with a **GetCellSoapOut** response message, as follows:

1. The row (2), column (2) on the worksheet specified by **sheetName**, specify the location of the sheet (2) cells to retrieve.
2. If **formatted** is **false**, the worksheet cell returned as the **GetCellResult** element MUST be either not present or one of the following types: **boolean**, **double**, **string**, or **CellError** (section [2.2.5.1](#)).
3. If **formatted** is true, the worksheet cell returned as the **GetCellResult** element MUST be type **string**, and MUST be formatted using a **number format** according to how the worksheet cell is number formatted in the workbook.
4. If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|---------------------------------------|
| DiscontiguousRangeNotSupported |
| EcsInvalidNamedObject |

| |
|---------------------------------|
| Subcode |
| InvalidSheetName |
| RangeDefinitionError |
| RangeParseError |
| RangeRequestAreaExceeded |
| SheetRangeMismatch |

3.1.4.8.1 Messages

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

| Message | Description |
|-----------------------|--|
| GetCellSoapIn | The request WSDL message for the GetCell WSDL operation. |
| GetCellSoapOut | The response WSDL message for the GetCell WSDL operation. |

3.1.4.8.1.1 GetCellSoapIn

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

The SOAP action value is:

`http://schemas.microsoft.com/office/excel/server/webservices/GetCell`

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

3.1.4.8.1.2 GetCellSoapOut

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

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

3.1.4.8.2 Elements

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

| Element | Description |
|------------------------|--|
| GetCell | The input data for the GetCell WSDL operation. |
| GetCellResponse | The result data for the GetCell WSDL operation. |

3.1.4.8.2.1 GetCell

The **GetCell** element specifies the input data for the **GetCell** WSDL operation.

```

<xs:element name="GetCell" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="row" type="xs:int"/>
      <xs:element minOccurs="1" maxOccurs="1" name="column" type="xs:int"/>
      <xs:element minOccurs="1" maxOccurs="1" name="formatted" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

sheetName: The name of the worksheet that contains the cell. See section [2.2.5.3](#). MUST be present. MUST NOT be empty.

row: The row (2) index of the cell. MUST be greater than or equal to 0 and less than or equal to 1,048,575.

column: The column (2) index of the cell. MUST be greater than or equal to 0 and less than or equal to 16,383

formatted: Specifies whether **GetCellResult** will be number formatted.

3.1.4.8.2.2 GetCellResponse

The **GetCellResponse** element specifies the result data for the **GetCell** WSDL operation.

```

<xs:element name="GetCellResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetCellResult"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

GetCellResult: The returned cell value.

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.8.3 Complex Types

None.

3.1.4.8.4 Simple Types

None.

3.1.4.8.5 Attributes

None.

3.1.4.8.6 Groups

None.

3.1.4.8.7 Attribute Groups

None.

3.1.4.9 GetCellA1

This operation is used to retrieve the contents of a specific cell in the workbook.

The following is the WSDL port type specification of the **GetCellA1** WSDL operation.

```
<wsdl:operation name="GetCellA1" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:GetCellA1SoapIn"/>
  <wsdl:output message="tns:GetCellA1SoapOut"/>
</wsdl:operation>
```

The protocol client sends a **GetCellA1SoapIn** request message, and the protocol server responds with a **GetCellA1SoapOut** response message, as follows:

- The **rangeName** specifies the location of the sheet (2) cell to retrieve. The sheet (2) that contains the sheet (2) cell to retrieve is determined in the following manner.
- If **rangeName** specifies a sheet (2), that is the sheet (2) which MUST be used.
- Otherwise, the sheet (2) specified in **sheetName** MUST be used, as specified in section [2.2.5.3](#).
- If neither **sheetName** nor **rangeName** specify a sheet (2), the protocol server MAY [<6>](#) use any sheet (1) in the workbook.
- If **formatted** is false, the sheet (2) cell returned as the **GetCellA1Result** element MUST be either not present or one of the following types: **boolean**, **double**, **string**, or **CellError** (section [2.2.5.1](#)).
- If **formatted** is true, the sheet (2) cell returned as the **GetCellA1Result** element MUST be type **string**, and MUST be number formatted according to how the sheet (2) cell is number formatted in the workbook.
- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|---------------------------------------|
| DiscontiguousRangeNotSupported |
| EcsInvalidNamedObject |

| |
|---------------------------------|
| Subcode |
| InvalidSheetName |
| RangeDefinitionError |
| RangeParseError |
| RangeRequestAreaExceeded |
| SheetRangeMismatch |

3.1.4.9.1 Messages

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

| Message | Description |
|-------------------------|--|
| GetCellA1SoapIn | The request WSDL message for the GetCellA1 WSDL operation. |
| GetCellA1SoapOut | The response WSDL message for the GetCellA1 WSDL operation. |

3.1.4.9.1.1 GetCellA1SoapIn

The request WSDL message for the **GetCellA1** WSDL operation.

The SOAP action value is:

`http://schemas.microsoft.com/office/excel/server/webservices/GetCellA1`

The SOAP body contains the **GetCellA1** element.

3.1.4.9.1.2 GetCellA1SoapOut

The response WSDL message for the **GetCellA1** WSDL operation.

The SOAP body contains the **GetCellA1Response** element.

3.1.4.9.2 Elements

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

| Element | Description |
|--------------------------|--|
| GetCellA1 | The input data for the GetCellA1 WSDL operation. |
| GetCellA1Response | The result data for the GetCellA1 WSDL operation. |

3.1.4.9.2.1 GetCellA1

The **GetCellA1** element specifies the input data for the **GetCellA1** WSDL operation.

```

<xs:element name="GetCellA1" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="rangeName" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="formatted" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

sheetName: The name of the sheet (2) that contains the cell. See section [2.2.5.3](#) for specification. MUST be present.

rangeName: The name of the range as specified in section [2.2.5.2](#). MUST be present. MUST NOT be longer than 255 characters.

formatted: Specifies whether **GetCellA1Result** will be number formatted.

3.1.4.9.2.2 GetCellA1Response

The **GetCellA1Response** element specifies the result data for the **GetCellA1** WSDL operation.

```

<xs:element name="GetCellA1Response" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetCellA1Result"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

GetCellA1Result: The returned cell value.

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.9.3 Complex Types

None.

3.1.4.9.4 Simple Types

None.

3.1.4.9.5 Attributes

None.

3.1.4.9.6 Groups

None.

3.1.4.9.7 Attribute Groups

None.

3.1.4.10 GetChartImageUrl

This operation returns the URL of a chart in a workbook.

The following is the WSDL port type specification of the **GetChartImageUrl** WSDL operation.

```
<wsdl:operation name="GetChartImageUrl" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:GetChartImageUrlSoapIn"/>
  <wsdl:output message="tns:GetChartImageUrlSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **GetChartUriSoapIn** request message, and the protocol server responds with a **GetChartUriSoapOut** response message as follows:

- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

3.1.4.10.1 Messages

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

| Message | Description |
|--------------------------------|---|
| GetChartImageUrlSoapIn | The request WSDL message for the GetChartImageUrl WSDL operation. |
| GetChartImageUrlSoapOut | The response WSDL message for the GetChartImageUrl WSDL operation. |

3.1.4.10.1.1 GetChartImageUrlSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/GetChartImageUrl
```

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

3.1.4.10.1.2 GetChartImageUrlSoapOut

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

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

3.1.4.10.2 Elements

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

| Element | Description |
|---------------------------------|---|
| GetChartImageUrl | The input data for the GetChartImageUrl WSDL operation. |
| GetChartImageUrlResponse | The result data for the GetChartImageUrl WSDL operation. |

3.1.4.10.2.1 GetChartImageUrl

The **GetChartImageUrl** element specifies the input data for the **GetChartImageUrl** WSDL operation.

```
<xs:element name="GetChartImageUrl" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="size" type="tns:Size"/>
      <xs:element minOccurs="0" maxOccurs="1" name="chartName" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

sessionId: A session identifier (2) . MUST be present. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operations, but MAY instead be obtained in any other way.

size: The size of the chart image as defined in section [3.1.4.10.3.1](#). If not present, the original chart size will be used.

chartName: The string identifier for a chart. MUST be present. MUST be non-empty.

3.1.4.10.2.2 GetChartImageUrlResponse

The **GetChartImageUrlResponse** element specifies the result data for the **GetChartImageUrl** WSDL operation.

```
<xs:element name="GetChartImageUrlResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetChartImageUrlResult"
type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetChartImageUrlResult: The URL of the requested chart.

status: An array of **Status** elements as specified in section [2.2.4.2](#).

3.1.4.10.3 Complex Types

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

| Complex type | Description |
|--------------|--------------------------------------|
| Size | Specifies the size of a chart image. |

3.1.4.10.3.1 Size

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

Specifies the size of a chart image defined as follows.

```
<xs:complexType name="Size" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="1" maxOccurs="1" name="Width" type="xs:int"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Height" type="xs:int"/>
  </xs:sequence>
</xs:complexType>
```

Width: Specifies the width of the image to be returned. MUST be an integer greater than or equal to 4 and less than or equal to 4096, or -1. If -1, the original width of the chart will be used. Must be -1 if **Height** is -1.

Height: Specifies the height of the image to be returned. MUST be an integer greater than or equal to 4 and less than or equal to 4096, or -1. If -1, the original height of the chart will be used. MUST be -1 if **Width** is -1.

3.1.4.10.4 Simple Types

None.

3.1.4.10.5 Attributes

None.

3.1.4.10.6 Groups

None.

3.1.4.10.7 Attribute Groups

None.

3.1.4.11 GetPublishedItemNames

This operation returns a list of the published items in a workbook.

The following is the WSDL port type specification of the **GetPublishedItemNames** WSDL operation.

```
<wsdl:operation name="GetPublishedItemNames" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
```

```

    <wsdl:input message="tns:GetPublishedItemNamesSoapIn"/>
    <wsdl:output message="tns:GetPublishedItemNamesSoapOut"/>
</wsdl:operation>

```

The protocol client sends a **GetPublishedItemNamesSoapIn** request message, and the protocol server responds with a **GetPublishedItemNamesSoapOut** response message as follows:

- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

3.1.4.11.1 Messages

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

| Message | Description |
|-------------------------------------|--|
| GetPublishedItemNamesSoapIn | The request WSDL message for the GetPublishedItemNames WSDL operation. |
| GetPublishedItemNamesSoapOut | The response WSDL message for the GetPublishedItemNames WSDL operation. |

3.1.4.11.1.1 GetPublishedItemNamesSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/GetPublishedItemNames
```

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

3.1.4.11.1.2 GetPublishedItemNamesSoapOut

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

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

3.1.4.11.2 Elements

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

| Element | Description |
|--------------------------------------|--|
| GetPublishedItemNames | The input data for the GetPublishedItemNames WSDL operation. |
| GetPublishedItemNamesResponse | The result data for the GetPublishedItemNames WSDL operation. |

3.1.4.11.2.1 GetPublishedItemNames

The **GetPublishedItemNames** element specifies the input data for the **GetPublishedItemNames** WSDL operation.

```
<xs:element name="GetPublishedItemNames" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

sessionId: A session identifier (2). MUST be present. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operations, but MAY instead be obtained in any other way.

3.1.4.11.2.2 GetPublishedItemNamesResponse

The **GetPublishedItemNamesResponse** element specifies the result data for the **GetPublishedItemNames** WSDL operation.

```
<xs:element name="GetPublishedItemNamesResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetPublishedItemNamesResult"
type="tns:ArrayOfWorkbookItem"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetPublishedItemNamesResult: An array of **WorkbookItem** elements as specified in section [3.1.4.11.3.1](#). MUST be present.

status: An array of **Status** elements as specified in section [2.2.4.2](#).

3.1.4.11.3 Complex Types

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

| Complex type | Description |
|----------------------------|--|
| ArrayOfWorkbookItem | An array of WorkbookItem elements. |
| WorkbookItem | Specifies the name and type of a published item. |

3.1.4.11.3.1 ArrayOfWorkbookItem

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

An array of **WorkbookItem** elements defined as follows.

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

WorkbookItem: A **WorkbookItem** as specified in section [3.1.4.11.3.2](#).

3.1.4.11.3.2 WorkbookItem

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

Specifies the name and type of a published item and is defined as follows.

```
<xs:complexType name="WorkbookItem" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Type" type="tns:ItemType"/>
  </xs:sequence>
</xs:complexType>
```

Name: The name of the published item. MUST be present. MUST NOT be empty.

Type: The type of the published item as specified in section [3.1.4.11.4.1](#).

3.1.4.11.4 Simple Types

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

| Simple type | Description |
|-----------------|---|
| ItemType | Specifies the type of a published item. |

3.1.4.11.4.1 ItemType

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

Specifies the type of a published item and is defined as follows.

```
<xs:simpleType name="ItemType" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Chart"/>
    <xs:enumeration value="PivotTable"/>
    <xs:enumeration value="Table"/>
    <xs:enumeration value="NamedRange"/>
  </xs:restriction>
</xs:simpleType>
```

The following table specifies the allowable values for the **ItemType** simple type.

| Value | Meaning |
|------------|---|
| Chart | Specifies that the published item is a chart. |
| PivotTable | Specifies that the published item is a PivotTable. |
| Table | Specifies that the published item is a list (2). |
| NamedRange | Specifies that the published item is a named range. |

3.1.4.11.5 Attributes

None.

3.1.4.11.6 Groups

None.

3.1.4.11.7 Attribute Groups

None.

3.1.4.12 GetRange

This operation is used to retrieve the contents of a specific range in the workbook.

The following is the WSDL port type specification of the **GetRange** WSDL operation.

```
<wsdl:operation name="GetRange" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:GetRangeSoapIn"/>
  <wsdl:output message="tns:GetRangeSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **GetRangeSoapIn** request message, and the protocol server responds with a **GetRangeSoapOut** response message, as follows:

1. The **rangeCoordinates** on the sheet (2) specified by **sheetName**, specify the location of the sheet (2) cells to retrieve.
2. Each element inside the **GetRangeResult** element MUST be of type **ArrayOfAnyType**. Each element inside **GetRangeResult** is a row (2) of sheet (1) cells returned from the protocol server. The number of rows (2) returned MUST be equal to the number of rows (2) requested in the **rangeCoordinates** element.
3. Each element inside each of the **ArrayOfAnyType** type nested inside **GetRangeResult** MUST be a single workbook sheet (2) cell. The number of elements inside each such nested **ArrayOfAnyType** MUST be equal to the number of columns (2) requested in the **rangeCoordinates** element.
4. If **formatted** is false, all sheet (2) cells returned as part of the **GetRangeResult** element MUST either be empty and have the `xsi:nil` attribute set to true or be one of the following types: **Boolean**, **double**, **string**, or **CellError** (section [2.2.5.1](#)).

5. If **formatted** is true, all sheet cells returned as part of the **GetRangeResult** element MUST be type **string**, and MUST be number formatted according to how the sheet (1) cell is number formatted in the workbook.
6. If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|---------------------------------------|
| DiscontiguousRangeNotSupported |
| EcsInvalidNamedObject |
| InvalidSheetName |
| RangeDefinitionError |
| RangeParseError |
| RangeRequestAreaExceeded |
| SheetRangeMismatch |

3.1.4.12.1 Messages

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

| Message | Description |
|------------------------|---|
| GetRangeSoapIn | The request WSDL message for the GetRange WSDL operation. |
| GetRangeSoapOut | The response WSDL message for the GetRange WSDL operation. |

3.1.4.12.1.1 GetRangeSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/GetRange
```

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

3.1.4.12.1.2 GetRangeSoapOut

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

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

3.1.4.12.2 Elements

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

| Element | Description |
|-------------------------|---|
| GetRange | The input data for the GetRange WSDL operation. |
| GetRangeResponse | The result data for the GetRange WSDL operation. |

3.1.4.12.2.1 GetRange

The **GetRange** element specifies the input data for the **GetRange** WSDL operation.

```
<xs:element name="GetRange" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="rangeCoordinates"
type="tns:RangeCoordinates"/>
      <xs:element minOccurs="1" maxOccurs="1" name="formatted" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

sheetName: The name of the sheet (2) that contains the range. See section [2.2.5.3](#) for specification. MUST be present. MUST NOT be empty.

rangeCoordinates: The range to retrieve.

- MUST be present.
- MUST NOT be empty, as specified in section [2.2.4.4](#).
- The product of **Height** and **Width** MUST NOT be greater than 1,000,000.

formatted: Specifies whether **GetRangeResult** will be number formatted.

3.1.4.12.2.2 GetRangeResponse

The **GetRangeResponse** element specifies the result data for the **GetRange** WSDL operation.

```
<xs:element name="GetRangeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetRangeResult"
type="tns:ArrayOfAnyType"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

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

GetRangeResult: The returned range value.

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.12.3 Complex Types

None.

3.1.4.12.4 Simple Types

None.

3.1.4.12.5 Attributes

None.

3.1.4.12.6 Groups

None.

3.1.4.12.7 Attribute Groups

None.

3.1.4.13 GetRangeA1

This operation is used to retrieve the contents of a named range in the workbook.

The following is the WSDL port type specification of the **GetRangeA1** WSDL operation.

```
<wsdl:operation name="GetRangeA1" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:GetRangeA1SoapIn"/>
  <wsdl:output message="tns:GetRangeA1SoapOut"/>
</wsdl:operation>
```

The protocol client sends a **GetRangeA1SoapIn** request message, and the protocol server responds with a **GetRangeA1SoapOut** response message, as follows:

- The **rangeName** specifies the named range of the cells on the sheet (2) to retrieve. The sheet (2) that contains the sheet (2) cells to retrieve is determined in the following manner:
- If **rangeName** specifies a sheet (2), that is the sheet (2) that MUST be used.
- Otherwise, the sheet (2) specified in **sheetName** MUST be used, as specified in section [2.2.5.3](#). If neither **sheetName** nor **rangeName** specify a sheet (2), the protocol server MAY [<7>](#) use any sheet (2) in the workbook.
- Each element inside the **GetRangeA1Result** element MUST be of type **ArrayOfAnyType**. Each element inside the **GetRangeA1Result** is a row (2) of sheet (1) cells returned from the protocol server. The number of rows (2) MUST be equal to the rows (2) in the named range specified by **rangeName**.

- Each element inside each of the **ArrayOfAnyType** type nested inside **GetRangeA1Result** MUST be a single workbook sheet (2) cell. The number of elements inside each such nested **ArrayOfAnyType** MUST be equal to the number of columns (2) in the named range specified by **rangeName**.
- If **formatted** is false, all sheet (2) cells returned as part of the **GetRangeA1Result** element MUST either be empty and have the `xsi:nil` attribute set to true or be one of the following types: **Boolean**, **double**, **string**, or [CellError](#) (section [2.2.5.1](#)).
- If **formatted** is true, all sheet (2) cells returned as part of the **GetRangeA1Result** element MUST be type **string**, and MUST be number formatted according to how the sheet (2) cell is number formatted in the workbook.
- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|---------------------------------------|
| DiscontiguousRangeNotSupported |
| EcsInvalidNamedObject |
| InvalidSheetName |
| RangeDefinitionError |
| RangeParseError |
| RangeRequestAreaExceeded |
| SheetRangeMismatch |

3.1.4.13.1 Messages

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

| Message | Description |
|--------------------------|---|
| GetRangeA1SoapIn | The request WSDL message for the GetRangeA1 WSDL operation. |
| GetRangeA1SoapOut | The response WSDL message for the GetRangeA1 WSDL operation. |

3.1.4.13.1.1 GetRangeA1SoapIn

The request WSDL message for the **GetRangeA1** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/GetRangeA1
```

The SOAP body contains the **GetRangeA1** element.

3.1.4.13.1.2 GetRangeA1SoapOut

The response WSDL message for the **GetRangeA1** WSDL operation.

The SOAP body contains the **GetRangeA1Response** element.

3.1.4.13.2 Elements

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

| Element | Description |
|---------------------------|---|
| GetRangeA1 | The input data for the GetRangeA1 WSDL operation. |
| GetRangeA1Response | The result data for the GetRangeA1 WSDL operation. |

3.1.4.13.2.1 GetRangeA1

The **GetRangeA1** element specifies the input data for the **GetRangeA1** WSDL operation.

```
<xs:element name="GetRangeA1" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="rangeName" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="formatted" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

sheetName: The name of the sheet (1) on which the **rangeName** is defined. See section [2.2.5.3](#) for specification. MUST be present.

rangeName: The name of the range as specified in section [2.2.5.2](#). MUST be present. MUST NOT be longer than 255 characters.

formatted: Specifies whether **GetRangeA1Result** will be number formatted.

3.1.4.13.2.2 GetRangeA1Response

The **GetRangeA1Response** element specifies the result data for the **GetRangeA1** WSDL operation.

```
<xs:element name="GetRangeA1Response" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetRangeA1Result"
        type="tns:ArrayOfAnyType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```
<xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
</xs:sequence>
</xs:complexType>
</xs:element>
```

GetRangeA1Result: The returned range value.

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.13.3 Complex Types

None.

3.1.4.13.4 Simple Types

None.

3.1.4.13.5 Attributes

None.

3.1.4.13.6 Groups

None.

3.1.4.13.7 Attribute Groups

None.

3.1.4.14 GetSessionInformation

This operation retrieves information about a session (2) on the protocol server.

The following is the WSDL port type specification of the **GetSessionInformation** WSDL operation.

```
<wsdl:operation name="GetSessionInformation" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:GetSessionInformationSoapIn"/>
  <wsdl:output message="tns:GetSessionInformationSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **GetSessionInformationSoapIn** request message, and the protocol server responds with a **GetSessionInformationSoapOut** response message, as follows:

- The protocol server MUST return the UI culture and the data culture for the specified session (2). These are the languages specified in the [OpenWorkbook](#) operation or the languages that the protocol server fell back to, as specified in [OpenWorkbook](#).
- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in [Common List of Subcodes](#), but MAY instead contain any subcode specified in [List of Subcodes](#) or any other subcode.

3.1.4.14.1 Messages

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

| Message | Description |
|-------------------------------------|--|
| GetSessionInformationSoapIn | The request WSDL message for the GetSessionInformation WSDL operation. |
| GetSessionInformationSoapOut | The response WSDL message for the GetSessionInformation WSDL operation. |

3.1.4.14.1.1 GetSessionInformationSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/GetSessionInformation
```

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

3.1.4.14.1.2 GetSessionInformationSoapOut

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

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

3.1.4.14.2 Elements

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

| Element | Description |
|--------------------------------------|--|
| GetSessionInformation | The input data for the GetSessionInformation WSDL operation. |
| GetSessionInformationResponse | The result data for the GetSessionInformation WSDL operation. |

3.1.4.14.2.1 GetSessionInformation

The **GetSessionInformation** element specifies the input data for the **GetSessionInformation** WSDL operation.

```
<xs:element name="GetSessionInformation" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

sessionId: A session identifier (2). MUST be present. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operations, but MAY instead be obtained in any other way.

3.1.4.14.2.2 GetSessionInformationResponse

The **GetSessionInformationResponse** element specifies the result data for the **GetSessionInformation** WSDL operation.

```
<xs:element name="GetSessionInformationResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="serverVersion" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="uiCultureNameUsed" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="dataCultureNameUsed" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

serverVersion: The version of the server. MUST be present.

uiCultureNameUsed: The UI culture. MUST be present. MUST be a language tag, as specified in [\[RFC1766\]](#) section 2.

dataCultureNameUsed: The data culture. MUST be present. MUST be a language tag, as specified in [\[RFC1766\]](#) section 2.

status: An array of **Status** elements as specified in section [2.2.4.2](#).

3.1.4.14.3 Complex Types

None.

3.1.4.14.4 Simple Types

None.

3.1.4.14.5 Attributes

None.

3.1.4.14.6 Groups

None.

3.1.4.14.7 Attribute Groups

None.

3.1.4.15 GetSheetNames

The operation returns information about the sheets in a workbook.

The following is the WSDL port type specification of the **GetSheetNames** WSDL operation.

```
<wsdl:operation name="GetSheetNames" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:GetSheetNamesSoapIn"/>
  <wsdl:output message="tns:GetSheetNamesSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **GetSheetNamesSoapIn** request message, and the protocol server responds with a **GetSheetNamesSoapOut** response message, as follows:

- If the session (2) is a view session in view-only mode, the number of elements inside **GetSheetNamesResult** MUST be equal to the number of visible **published sheets** in the workbook.
- If the session (2) is not a view session in view-only mode, the number of elements inside **GetSheetNamesResult** MUST be equal to the number of loaded sheets (1) in the workbook.
- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

3.1.4.15.1 Messages

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

| Message | Description |
|-----------------------------|--|
| GetSheetNamesSoapIn | The request WSDL message for the GetSheetNames WSDL operation. |
| GetSheetNamesSoapOut | The response WSDL message for the GetSheetNames WSDL operation. |

3.1.4.15.1.1 GetSheetNamesSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/GetSheetNames
```

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

3.1.4.15.1.2 GetSheetNamesSoapOut

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

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

3.1.4.15.2 Elements

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

| Element | Description |
|------------------------------|--|
| GetSheetNames | The input data for the GetSheetNames WSDL operation. |
| GetSheetNamesResponse | The result data for the GetSheetNames WSDL operation. |

3.1.4.15.2.1 GetSheetNames

The **GetSheetNames** element specifies the input data for the **GetSheetNames** WSDL operation.

```
<xs:element name="GetSheetNames" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

sessionId: A session identifier (2). MUST be present. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

3.1.4.15.2.2 GetSheetNamesResponse

The **GetSheetNamesResponse** element specifies the result data for the **GetSheetNames** WSDL operation.

```
<xs:element name="GetSheetNamesResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetSheetNamesResult"
type="tns:ArrayOfSheetInfo"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetSheetNamesResult: An array of **SheetInfo** elements as specified in section [3.1.4.15.3.1](#). MUST be present.

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.15.3 Complex Types

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

| Complex type | Description |
|-------------------------|---|
| ArrayOfSheetInfo | An array of SheetInfo elements. |
| SheetInfo | Specifies the name , type , and visible condition of a sheet (1). |

3.1.4.15.3.1 ArrayOfSheetInfo

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

An array of **SheetInfo** elements defined as follows.

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

SheetInfo: A **SheetInfo** as specified in section [3.1.4.15.3.2](#).

3.1.4.15.3.2 SheetInfo

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

Specifies the **name**, **type**, and visible condition of a sheet (1) and is defined as follows.

```
<xs:complexType name="SheetInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Type" type="tns:SheetType"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Visibility" type="tns:SheetVisibility"/>
  </xs:sequence>
</xs:complexType>
```

Name: The name of the sheet (1). MUST be present.

Type: The type of the sheet (1) as specified in section [3.1.4.15.4.1](#).

Visibility: The visible condition of the sheet (1) as specified in section [3.1.4.15.4.2](#).

3.1.4.15.4 Simple Types

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

| Simple type | Description |
|------------------------|---|
| SheetType | Specifies the type of a sheet (1). |
| SheetVisibility | Specifies the visible condition of a sheet (1). |

3.1.4.15.4.1 SheetType

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

Specifies the type of a sheet (1) and is defined as follows.

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



```

    <xs:enumeration value="Worksheet"/>
    <xs:enumeration value="Chartsheet"/>
  </xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for the **SheetType** simple type.

| Value | Meaning |
|------------|------------------------|
| Worksheet | A worksheet. |
| Chartsheet | A chart sheet . |

3.1.4.15.4.2 SheetVisibility

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

Specifies the visible condition of a sheet (1) and is defined as follows.

```

<xs:simpleType name="SheetVisibility" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Visible"/>
    <xs:enumeration value="Hidden"/>
    <xs:enumeration value="VeryHidden"/>
  </xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for the **SheetVisibility** simple type.

| Value | Meaning |
|------------|---|
| Visible | Specifies that the sheet (1) is visible. |
| Hidden | Specifies that the sheet (1) is not visible, but can be unhidden via the client UI. |
| VeryHidden | Specifies that the sheet (1) is not visible and cannot be unhidden via the client UI. |

3.1.4.15.5 Attributes

None.

3.1.4.15.6 Groups

None.

3.1.4.15.7 Attribute Groups

None.

3.1.4.16 GetWorkbook

This operation is used to retrieve a workbook file.

The following is the WSDL port type specification of the **GetWorkbook** WSDL operation.

```

<wsdl:operation name="GetWorkbook" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:GetWorkbookSoapIn"/>
  <wsdl:output message="tns:GetWorkbookSoapOut"/>
</wsdl:operation>

```

The protocol client sends a **GetWorkbookSoapIn** request message, and the protocol server responds with a **GetWorkbookSoapOut** response message, as follows:

- If **workbookType** is **FullWorkbook** or **FullSnapshot** and the user does not have Open Item permission, the protocol server returns a SOAP fault with subcode **EcsSnapshotAccessDenied**.
- If the protocol server cannot retrieve the workbook file in the form specified by **workbookType**, the protocol server returns a SOAP fault with subcode **SaveFailed**.
- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section 2.2.9.2, but MAY instead contain any subcode specified in section 2.2.9.1 or any other subcode.

| Subcode |
|--------------------------------|
| EcsSnapshotAccessDenied |
| SaveFailed |

3.1.4.16.1 Messages

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

| Message | Description |
|---------------------------|--|
| GetWorkbookSoapIn | The request WSDL message for the GetWorkbook WSDL operation. |
| GetWorkbookSoapOut | The response WSDL message for the GetWorkbook WSDL operation. |

3.1.4.16.1.1 GetWorkbookSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/GetWorkbook
```

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

3.1.4.16.1.2 GetWorkbookSoapOut

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

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

3.1.4.16.2 Elements

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

| Element | Description |
|----------------------------|--|
| GetWorkbook | The input data for the GetWorkbook WSDL operation. |
| GetWorkbookResponse | The result data for the GetWorkbook WSDL operation. |

3.1.4.16.2.1 GetWorkbook

The **GetWorkbook** element specifies the input data for the **GetWorkbook** WSDL operation.

```
<xs:element name="GetWorkbook" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="workbookType" type="tns:WorkbookType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

sessionId: A session (2) identifier. MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

workbookType: The form of the requested workbook as specified in section [2.2.5.4](#).

3.1.4.16.2.2 GetWorkbookResponse

The **GetWorkbookResponse** element specifies the result data for the **GetWorkbook** WSDL operation.

```
<xs:element name="GetWorkbookResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetWorkbookResult"
type="xs:base64Binary"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

GetWorkbookResult: A workbook file. MUST be present.

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.16.3 Complex Types

None.

3.1.4.16.4 Simple Types

None.

3.1.4.16.5 Attributes

None.

3.1.4.16.6 Groups

None.

3.1.4.16.7 Attribute Groups

None.

3.1.4.17 GetWorkbookConnections

Returns a list of data connection (2) names in the workbook.

The following is the WSDL port type specification of the **GetWorkbookConnections** WSDL operation.

```
<wsdl:operation name="GetWorkbookConnections" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:GetWorkbookConnectionsSoapIn"/>
  <wsdl:output message="tns:GetWorkbookConnectionsSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **GetWorkbookConnectionsSoapIn** request WSDL message, and the protocol server responds with a **GetWorkbookConnectionsSoapOut** response WSDL message, as follows:

- The protocol server returns a list of the data connection (2) names.
- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

3.1.4.17.1 Messages

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

| Message | Description |
|--------------------------------------|---|
| GetWorkbookConnectionsSoapIn | The request WSDL message for the GetWorkbookConnections WSDL operation. |
| GetWorkbookConnectionsSoapOut | The response WSDL message for the GetWorkbookConnections WSDL operation. |

3.1.4.17.1.1 GetWorkbookConnectionsSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/GetWorkbookConnections
```

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

3.1.4.17.1.2 GetWorkbookConnectionsSoapOut

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

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

3.1.4.17.2 Elements

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

| Element | Description |
|---------------------------------------|---|
| GetWorkbookConnections | The input data for the GetWorkbookConnections WSDL operation. |
| GetWorkbookConnectionsResponse | The result data for the GetWorkbookConnections WSDL operation. |

3.1.4.17.2.1 GetWorkbookConnections

The **GetWorkbookConnections** element specifies the input data for the **GetWorkbookConnections** WSDL operation.

```
<xs:element name="GetWorkbookConnections" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

sessionId: A session (2) identifier. MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

3.1.4.17.2.2 GetWorkbookConnectionsResponse

The **GetWorkbookConnectionsResponse** element specifies the result data for the **GetWorkbookConnections** WSDL operation.

```
<xs:element name="GetWorkbookConnectionsResponse"  
  xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" maxOccurs="1" name="GetWorkbookConnectionsResult"  
        type="tns:ArrayOfString"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

```
<xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
</xs:sequence>
</xs:complexType>
</xs:element>
```

GetWorkbookConnectionsResult: An array of data connection (2) names.

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.17.3 Complex Types

None.

3.1.4.17.4 Simple Types

None.

3.1.4.17.5 Attributes

None.

3.1.4.17.6 Groups

None.

3.1.4.17.7 Attribute Groups

None.

3.1.4.18 NewWorkbook

This operation is used to create a new workbook session.

The following is the WSDL port type specification of the **NewWorkbook** WSDL operation.

```
<wsdl:operation name="NewWorkbook" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:NewWorkbookSoapIn"/>
  <wsdl:output message="tns:NewWorkbookSoapOut"/>
</wsdl:operation>
```

If the language represented by **uiCultureName** is unsupported on the protocol server, the protocol server MAY [<8>](#) fall back to some other supported language. **uiCultureName** MAY [<9>](#) be empty, in which case the determination of which language to default to is protocol-server-implementation specific.

If the language represented by **dataCultureName** is unsupported on the protocol server, the protocol server MAY [<10>](#) fall back to some other supported language. **dataCultureName** MAY [<11>](#) be empty, in which case the determination of which language to default to is protocol-server-implementation specific.

If the operation succeeds, the protocol server MUST return the session identifier (2) of the newly created session (2) in the response message, to be used in any future operations in the newly created session.

If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

3.1.4.18.1 Messages

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

| Message | Description |
|---------------------------|--|
| NewWorkbookSoapIn | The request WSDL message for the NewWorkbook WSDL operation. |
| NewWorkbookSoapOut | The response WSDL message for the NewWorkbook WSDL operation. |

3.1.4.18.1.1 NewWorkbookSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/NewWorkbook
```

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

3.1.4.18.1.2 NewWorkbookSoapOut

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

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

3.1.4.18.2 Elements

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

| Element | Description |
|----------------------------|--|
| NewWorkbook | The input data for the NewWorkbook WSDL operation. |
| NewWorkbookResponse | The result data for the NewWorkbook WSDL operation. |

3.1.4.18.2.1 NewWorkbook

The **NewWorkbook** element specifies the input data for the **NewWorkbook** WSDL operation.

```
<xs:element name="NewWorkbook" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="path" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="uiCultureName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="dataCultureName" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
```

</xs:element>

path: The **trusted location (2)** of the new workbook session. MUST be present. MUST NOT be empty. MUST NOT be longer than 4095 characters. MUST be a **UNC**, or a **URI** as specified in [\[RFC2396\]](#).

uiCultureName: A language tag, as specified in [\[RFC1766\]](#), Section 2. Specifies the UI culture. MUST NOT be longer than 4095 characters.

dataCultureName: A language tag, as specified in [\[RFC1766\]](#), Section 2. Specifies the data culture. MUST NOT be longer than 4095 characters.

3.1.4.18.2 NewWorkbookResponse

The **NewWorkbookResponse** element specifies the result data for the **NewWorkbook** WSDL operation.

```
<xs:element name="NewWorkbookResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="NewWorkbookResult" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

NewWorkbookResult: The session identifier (2) of the newly created session (2). MUST be present.

status: The status information returned by the protocol server.

3.1.4.18.3 Complex Types

None.

3.1.4.18.4 Simple Types

None.

3.1.4.18.5 Attributes

None.

3.1.4.18.6 Groups

None.

3.1.4.18.7 Attribute Groups

None.

3.1.4.19 OpenWorkbook

This operation is used to create a new view session, giving access to a workbook on the protocol server.

The following is the WSDL port type specification of the **OpenWorkbook** WSDL operation.

```
<wsdl:operation name="OpenWorkbook" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:OpenWorkbookSoapIn"/>
  <wsdl:output message="tns:OpenWorkbookSoapOut"/>
</wsdl:operation>
```

The protocol client sends an **OpenWorkbookSoapIn** request message, and the protocol server responds with an **OpenWorkbookSoapOut** response message, as follows:

1. The workbook file, specified by **workbookPath**, is specified as follows:
 1. It SHOULD be in a format specified by [\[MS-XLSB\]](#), [\[ISO/IEC29500-1:2011\]](#), [\[MS-OFFMACRO2\]](#), or [\[ODF\]](#), but MAY be in any other format.
 2. If it is in the [\[ISO/IEC29500-1:2011\]](#) format:
 1. It MAY [<12>](#) contain any valid SpreadsheetML records in that format.
 2. If a **odcFile** attribute of a **connection** element ([\[ISO/IEC29500-1:2011\]](#), Part 1, Section 18.13.1) is present, it SHOULD be the path of a file in the format specified by [\[MS-ODCFF\]](#).
 3. If a **connection** attribute of **dbPr** element ([\[ISO/IEC29500-1:2011\]](#), Part 1, Section 18.13.3) is present, it SHOULD be in the format specified by [\[MS-ODBCSTR\]](#) or [\[MS-OLEDBSTR\]](#).
 3. If it is in the [MS-XLSB] format:
 1. It MAY [<13>](#) contain any valid records in that format.
 2. If a **brtBeginExternalConnection** record is present and it has the **stConnectionFile** field present, the **stConnectionFile** SHOULD be the path of a file in the format specified by [MS-ODCFF].
 3. If a **BrtBeginECDBProps** record is present, its **stConn** field SHOULD be in the format specified by [MS-ODBCSTR] or [MS-OLEDBSTR].
 4. If a file in [MS-ODCFF] is referenced by the workbook file:
 1. **SourceFile** element of **OfficeDataConnection** ([\[MS-ODCFF\]](#) section 2.7.1.1) MUST NOT be present.
 2. **Parameter** elements of **CT_Connection** ([\[MS-ODCFF\]](#) section 2.7.1.1) MUST be ignored.
2. If the language represented by **uiCultureName** is unsupported on the protocol server, the protocol server MAY [<14>](#) fall back to some other supported language. **uiCultureName** MAY [<15>](#) be empty, in which case the determination of which language to default to is protocol-server-implementation specific.
3. If the language represented by **dataCultureName** is unsupported on the protocol server, the protocol server MAY [<16>](#) fall back to some other supported language. **dataCultureName**

MAY<17> be empty, in which case the determination of which language to default to is protocol-server-implementation specific.

4. If the operation succeeds, the protocol server MUST return the session identifier (2) of the newly created session (2) in the response message, to be used in any future operations in the newly created session (2).
5. If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section 2.2.9.2, but MAY instead contain any subcode specified in section 2.2.9.1 or any other subcode.

| Subcode |
|----------------------|
| FileOpenNotFound |
| FileOpenAccessDenied |
| FileCorrupt |
| GenericFileOpenError |
| WorkbookNotSupported |

3.1.4.19.1 Messages

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

| Message | Description |
|---------------------|---|
| OpenWorkbookSoapIn | The request WSDL message for the OpenWorkbook WSDL operation. |
| OpenWorkbookSoapOut | The response WSDL message for the OpenWorkbook WSDL operation. |

3.1.4.19.1.1 OpenWorkbookSoapIn

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

The SOAP action value is:

`http://schemas.microsoft.com/office/excel/server/webservices/OpenWorkbook`

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

3.1.4.19.1.2 OpenWorkbookSoapOut

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

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

3.1.4.19.2 Elements

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

| Element | Description |
|-----------------------------|---|
| OpenWorkbook | The input data for the OpenWorkbook WSDL operation. |
| OpenWorkbookResponse | The result data for the OpenWorkbook WSDL operation. |

3.1.4.19.2.1 OpenWorkbook

The **OpenWorkbook** element specifies the input data for the **OpenWorkbook** WSDL operation.

```
<xs:element name="OpenWorkbook" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="workbookPath" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="uiCultureName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="dataCultureName" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

workbookPath: The location of the workbook file. MUST be present. MUST NOT be empty. MUST NOT be longer than 4095 characters. MUST be a URI as specified in [\[RFC2396\]](#), or a UNC.

uiCultureName: A language tag, as specified in [\[RFC1766\]](#), Section 2. Specifies the UI culture. MUST NOT be longer than 4095 characters.

dataCultureName: A language tag, as specified in [\[RFC1766\]](#), Section 2. Specifies the data culture. MUST NOT be longer than 4095 characters.

3.1.4.19.2.2 OpenWorkbookResponse

The **OpenWorkbookResponse** element specifies the result data for the **OpenWorkbook** WSDL operation.

```
<xs:element name="OpenWorkbookResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="OpenWorkbookResult" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

OpenWorkbookResult: The session identifier (2) of the newly created session (2). MUST be present.

status: The status information returned by the protocol server.

3.1.4.19.3 Complex Types

None.

3.1.4.19.4 Simple Types

None.

3.1.4.19.5 Attributes

None.

3.1.4.19.6 Groups

None.

3.1.4.19.7 Attribute Groups

None.

3.1.4.20 OpenWorkbookEx

This operation is used to create a new view session, giving access to a workbook on the protocol server.

The following is the WSDL port type specification of the **OpenWorkbookEx** WSDL operation.

```
<wsdl:operation name="OpenWorkbookEx" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:OpenWorkbookExSoapIn"/>
  <wsdl:output message="tns:OpenWorkbookExSoapOut"/>
</wsdl:operation>
```

For details, see the **OpenWorkbook** operation (section [3.1.4.19](#)). When processing the **OpenWorkbookEx** operation the protocol server MUST suppress connected data refresh.

3.1.4.20.1 Messages

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

| Message | Description |
|------------------------------|---|
| OpenWorkbookExSoapIn | The request WSDL message for the OpenWorkbookEx WSDL operation. |
| OpenWorkbookExSoapOut | The response WSDL message for the OpenWorkbookEx WSDL operation. |

3.1.4.20.1.1 OpenWorkbookExSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/OpenWorkbookEx
```

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

3.1.4.20.1.2 OpenWorkbookExSoapOut

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

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

3.1.4.20.2 Elements

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

| Element | Description |
|-------------------------------|---|
| OpenWorkbookEx | The input data for the OpenWorkbookEx WSDL operation. |
| OpenWorkbookExResponse | The result data for the OpenWorkbookEx WSDL operation. |

3.1.4.20.2.1 OpenWorkbookEx

The **OpenWorkbookEx** element specifies the input data for the **OpenWorkbookEx** WSDL operation.

```
<xs:element name="OpenWorkbookEx" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="workbookPath" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="uiCultureName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="dataCultureName" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="exclusive" type="xs:boolean"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

workbookPath: See section [3.1.4.19.2.1](#).

uiCultureName: See section [3.1.4.19.2.1](#).

dataCultureName: See section [3.1.4.19.2.1](#).

exclusive: If true, the data model in the new view session MUST not be shared with any other sessions, and the database where the model will be loaded (as specified in section [3.1.4.6.3.1](#)) MUST NOT be read-only.

3.1.4.20.2.2 OpenWorkbookExResponse

The **OpenWorkbookExResponse** element specifies the result data for the **OpenWorkbookEx** WSDL operation.

```
<xs:element name="OpenWorkbookExResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="OpenWorkbookExResult" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
```

</xs:element>

OpenWorkbookExResult: See section [3.1.4.19.2.2](#).

status: See section [3.1.4.19.2.2](#).

3.1.4.20.3 Complex Types

None.

3.1.4.20.4 Simple Types

None.

3.1.4.20.5 Attributes

None.

3.1.4.20.6 Groups

None.

3.1.4.20.7 Attribute Groups

None.

3.1.4.21 OpenWorkbookForEditing

This operation is used to create a new edit session, giving access to a workbook on the protocol server.

The following is the WSDL port type specification of the **OpenWorkbookForEditing** WSDL operation.

```
<wsdl:operation name="OpenWorkbookForEditing" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:OpenWorkbookForEditingSoapIn"/>
  <wsdl:output message="tns:OpenWorkbookForEditingSoapOut"/>
</wsdl:operation>
```

The protocol client sends an **OpenWorkbookForEditingSoapIn** request message, and the protocol server responds with an **OpenWorkbookForEditingSoapOut** response message, as follows:

1. The workbook file, specified by **workbookPath**, is specified as follows:
 1. It SHOULD be in a format specified by [\[MS-XLSB\]](#), [\[ISO/IEC29500-1:2011\]](#), [\[MS-OFFMACRO2\]](#), or [\[ODF\]](#), but MAY be in any other format.
 2. If it is in the [\[ISO/IEC29500-1:2011\]](#) format:
 1. It MAY [<18>](#) contain any valid SpreadsheetML records in that format.
 2. If an **odcFile** attribute of a **connection** element ([\[ISO/IEC29500-1:2011\]](#), Part 1, Section 18.13.1) is present, it SHOULD be the path of a file in the format specified by [\[MS-ODCFF\]](#).

3. If a **connection** attribute of **dbPr** element ([\[ISO/IEC29500-1:2011\]](#), Part 1, Section 18.13.3) is present, it SHOULD be in the format specified by [\[MS-ODBCSTR\]](#) or [\[MS-OLEDBSTR\]](#).
3. If it is in the [MS-XLSB] format:
 1. It MAY [<19>](#) contain any valid records in that format.
 2. If a **brtBeginExternalConnection** record is present and it has the **stConnectionFile** field present, the **stConnectionFile** SHOULD be the path of a file in the format specified by [MS-ODCFF].
 3. If a **BrtBeginECDBProps** record is present, its **stConn** field SHOULD be in the format specified by [MS-ODBCSTR] or [MS-OLEDBSTR].
4. If a file in [MS-ODCFF] is referenced by the workbook file:
 1. **SourceFile** element of **OfficeDataConnection** ([\[MS-ODCFF\]](#) section 2.7.1.1) MUST NOT be present.
 2. **Parameter** elements of **CT_Connection** ([\[MS-ODCFF\]](#) section 2.2.1) MUST be ignored.
2. If the language represented by **uiCultureName** is unsupported on the protocol server, the protocol server MAY [<20>](#) fall back to some other supported language. **uiCultureName** MAY [<21>](#) be empty, in which case the determination of which language to default to is protocol-server-implementation specific.
3. If the language represented by **dataCultureName** is unsupported on the protocol server, the protocol server MAY [<22>](#) fall back to some other supported language. **dataCultureName** MAY [<23>](#) be empty, in which case the determination of which language to default to is protocol-server-implementation specific.
4. If the operation succeeds, the protocol server MUST return the session identifier (2) of the newly created session (2) in the response message, to be used in any future operations in the newly created session (2).
5. If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|-----------------------------|
| FileOpenNotFound |
| FileOpenAccessDenied |
| FileCorrupt |
| GenericFileOpenError |

3.1.4.21.1 Messages

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

| Message | Description |
|--------------------------------------|---|
| OpenWorkbookForEditingSoapIn | The request WSDL message for the OpenWorkbookForEditing WSDL operation. |
| OpenWorkbookForEditingSoapOut | The response WSDL message for the OpenWorkbookForEditing WSDL operation. |

3.1.4.21.1.1 OpenWorkbookForEditingSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/OpenWorkbookForEditing
```

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

3.1.4.21.1.2 OpenWorkbookForEditingSoapOut

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

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

3.1.4.21.2 Elements

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

| Element | Description |
|---------------------------------------|---|
| OpenWorkbookForEditing | The input data for the OpenWorkbookForEditing WSDL operation. |
| OpenWorkbookForEditingResponse | The result data for the OpenWorkbookForEditing WSDL operation. |

3.1.4.21.2.1 OpenWorkbookForEditing

The **OpenWorkbookForEditing** element specifies the input data for the **OpenWorkbookForEditing** WSDL operation.

```
<xs:element name="OpenWorkbookForEditing" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="workbookPath" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="uiCultureName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="dataCultureName" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

workbookPath: The location of the workbook file. MUST be present. MUST NOT be empty. MUST NOT be longer than 4095 characters. MUST be a URI as specified in [\[RFC2396\]](#), or a UNC.

uiCultureName: A language tag, as specified in [\[RFC1766\]](#), Section 2. Specifies the UI culture. MUST NOT be longer than 4095 characters.

dataCultureName: A language tag, as specified in [\[RFC1766\]](#), Section 2. Specifies the data culture. MUST NOT be longer than 4095 characters.

3.1.4.21.2.2 OpenWorkbookForEditingResponse

The **OpenWorkbookForEditingResponse** element specifies the result data for the **OpenWorkbookForEditing** WSDL operation.

```
<xs:element name="OpenWorkbookForEditingResponse"
  xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="OpenWorkbookForEditingResult"
        type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

OpenWorkbookForEditingResult: The session identifier (2) of the newly created session (2). MUST be present.

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.21.3 Complex Types

None.

3.1.4.21.4 Simple Types

None.

3.1.4.21.5 Attributes

None.

3.1.4.21.6 Groups

None.

3.1.4.21.7 Attribute Groups

None.

3.1.4.22 Refresh

This operation is used to refresh connected data on a workbook.

The following is the WSDL port type specification of the **Refresh** WSDL operation.

```
<wsdl:operation name="Refresh" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:RefreshSoapIn"/>
```

```
<wsdl:output message="tns:RefreshSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **RefreshSoapIn** request message, and the protocol server responds with a **RefreshSoapOut** response message, as follows:

1. If **connectionName** is not present or is empty, the protocol server refreshes all data connections (2) in the workbook if any exist. If **connectionName** is present and is not empty, the protocol server refreshes the specified data connections (2) in the workbook.
2. If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

3.1.4.22.1 Messages

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

| Message | Description |
|-----------------------|--|
| RefreshSoapIn | The request WSDL message for the Refresh WSDL operation. |
| RefreshSoapOut | The response WSDL message for the Refresh WSDL operation. |

3.1.4.22.1.1 RefreshSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/Refresh
```

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

3.1.4.22.1.2 RefreshSoapOut

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

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

3.1.4.22.2 Elements

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

| Element | Description |
|------------------------|--|
| Refresh | The input data for the Refresh WSDL operation. |
| RefreshResponse | The result data for the Refresh WSDL operation. |

3.1.4.22.2.1 Refresh

The **Refresh** element specifies the input data for the **Refresh** WSDL operation.

```
<xs:element name="Refresh" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="connectionName" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

connectionName: A name of a data connection (2) in the workbook. MUST NOT be longer than 255 characters.

3.1.4.22.2.2 RefreshResponse

The **RefreshResponse** element specifies the result data for the **Refresh** WSDL operation.

```
<xs:element name="RefreshResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

status: An array of **Status** elements as specified in section [2.2.4.2](#).

3.1.4.22.3 Complex Types

None.

3.1.4.22.4 Simple Types

None.

3.1.4.22.5 Attributes

None.

3.1.4.22.6 Groups

None.

3.1.4.22.7 Attribute Groups

None.

3.1.4.23 RefreshEx

This operation is used to refresh connected data on a workbook.

The following is the WSDL port type specification of the **RefreshEx** WSDL operation.

```
<wsdl:operation name="RefreshEx" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:RefreshExSoapIn"/>
  <wsdl:output message="tns:RefreshExSoapOut"/>
</wsdl:operation>
```

For details, see the **Refresh** operation (section [3.1.4.22](#)).

3.1.4.23.1 Messages

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

| Message | Description |
|-------------------------|--|
| RefreshExSoapIn | The request WSDL message for the RefreshEx WSDL operation. |
| RefreshExSoapOut | The response WSDL message for the RefreshEx WSDL operation. |

3.1.4.23.1.1 RefreshExSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/RefreshEx
```

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

3.1.4.23.1.2 RefreshExSoapOut

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

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

3.1.4.23.2 Elements

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

| Element | Description |
|--------------------------|--|
| RefreshEx | The input data for the RefreshEx WSDL operation. |
| RefreshExResponse | The result data for the RefreshEx WSDL operation. |

3.1.4.23.2.1 RefreshEx

The **RefreshEx** element specifies the input data for the **RefreshEx** WSDL operation.

```
<xs:element name="RefreshEx" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="connectionName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="targetApplicationId" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

connectionName: A name of a data connection (2) in the workbook. MUST NOT be longer than 255 characters.

targetApplicationId: Single Sign On identifier, as defined in [\[ISO/IEC29500-1:2012\]](#) section 18.13.1, used when refreshing data connections (2).

3.1.4.23.2.2 RefreshExResponse

The **RefreshExResponse** element specifies the result data for the **RefreshEx** WSDL operation.

```
<xs:element name="RefreshExResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

status: An array of **Status** elements as specified in section [2.2.4.2](#).

3.1.4.23.3 Complex Types

None.

3.1.4.23.4 Simple Types

None.

3.1.4.23.5 Attributes

None.

3.1.4.23.6 Groups

None.

3.1.4.23.7 Attribute Groups

None.

3.1.4.24 SaveWorkbook

This operation is used to save a workbook associated with an open session (2) to the same location from which it was opened.

The following is the WSDL port type specification of the **SaveWorkbook** WSDL operation.

```
<wsdl:operation name="SaveWorkbook" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:SaveWorkbookSoapIn"/>
  <wsdl:output message="tns:SaveWorkbookSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **SaveWorkbookSoapIn** request message, and the protocol server responds with a **SaveWorkbookSoapOut** response message.

If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|----------------------------------|
| FileSaveGeneric |
| OperationNotSupportedInViewOnly |
| SaveFailed |
| SavingUnsupportedInViewMode |
| SavingWorkbookAccessDenied |
| SavingWorkbookLockedOrCheckedOut |
| SavingWorkbookNoWorkbookName |

3.1.4.24.1 Messages

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

| Message | Description |
|---------------------|---|
| SaveWorkbookSoapIn | The request WSDL message for the SaveWorkbook WSDL operation. |
| SaveWorkbookSoapOut | The response WSDL message for the SaveWorkbook WSDL operation. |

3.1.4.24.1.1 SaveWorkbookSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/SaveWorkbook
```

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

3.1.4.24.1.2 SaveWorkbookSoapOut

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

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

3.1.4.24.2 Elements

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

| Element | Description |
|-----------------------------|---|
| SaveWorkbook | The input data for the SaveWorkbook WSDL operation. |
| SaveWorkbookResponse | The result data for the SaveWorkbook WSDL operation. |

3.1.4.24.2.1 SaveWorkbook

The **SaveWorkbook** element specifies the input data for the **SaveWorkbook** WSDL operation.

```
<xs:element name="SaveWorkbook" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

3.1.4.24.2.2 SaveWorkbookResponse

The **SaveWorkbookResponse** element specifies the result data for the **SaveWorkbook** WSDL operation.

```
<xs:element name="SaveWorkbookResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">  
  <xs:complexType>  
    <xs:sequence>  
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>  
    </xs:sequence>  
  </xs:complexType>  
</xs:element>
```

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.24.3 Complex Types

None.

3.1.4.24.4 Simple Types

None.

3.1.4.24.5 Attributes

None.

3.1.4.24.6 Groups

None.

3.1.4.24.7 Attribute Groups

None.

3.1.4.25 SaveWorkbookCopy

This operation is used to save a workbook associated with an open session (2) to a specified location, and in a specified format.

The following is the WSDL port type specification of the **SaveWorkbookCopy** WSDL operation.

```
<wsdl:operation name="SaveWorkbookCopy" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:SaveWorkbookCopySoapIn"/>
  <wsdl:output message="tns:SaveWorkbookCopySoapOut"/>
</wsdl:operation>
```

The protocol client sends a **SaveWorkbookSoapIn** request message and the protocol server responds with a **SaveWorkbookSoapOut** response message, as follows:

- If the protocol server cannot save the workbook in the form specified by **workbookType**, the protocol server returns a SOAP fault with subcode **SaveFailed**.
- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|-----------------------------------|
| EcsSnapshotAccessDenied |
| FileSaveGeneric |
| SaveFailed |
| SavingWorkbookAccessDenied |

| |
|--------------------------------------|
| Subcode |
| SavingWorkbookFileNameExists |
| SavingWorkbookInvalidFileName |
| SavingWorkbookInvalidLocation |

3.1.4.25.1 Messages

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

| Message | Description |
|--------------------------------|---|
| SaveWorkbookCopySoapIn | The request WSDL message for the SaveWorkbookCopy WSDL operation. |
| SaveWorkbookCopySoapOut | The response WSDL message for the SaveWorkbookCopy WSDL operation. |

3.1.4.25.1.1 SaveWorkbookCopySoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/SaveWorkbookCopy
```

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

3.1.4.25.1.2 SaveWorkbookCopySoapOut

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

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

3.1.4.25.2 Elements

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

| Element | Description |
|---------------------------------|---|
| SaveWorkbookCopy | The input data for the SaveWorkbookCopy WSDL operation. |
| SaveWorkbookCopyResponse | The result data for the SaveWorkbookCopy WSDL operation. |

3.1.4.25.2.1 SaveWorkbookCopy

The **SaveWorkbookCopy** element specifies the input data for the **SaveWorkbookCopy** WSDL operation.

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

```

<xs:complexType>
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="workbookPath" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="workbookType" type="tns:WorkbookType"/>
    <xs:element minOccurs="1" maxOccurs="1" name="saveOptions" type="tns:SaveOptions"/>
  </xs:sequence>
</xs:complexType>
</xs:element>

```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

workbookPath: The location of the workbook file to which to write.

- MUST be present.

If the specified value is not a Uniform Resource Identifier (URI) as specified in [\[RFC2396\]](#), or a Universal Naming Convention (UNC), the protocol server MUST return a SOAP fault response with subcode **SavingWorkbookInvalidFileName**.

- The protocol server ignores the number sign ("#") and fragment, as specified in [\[RFC3986\]](#).

workbookType: A **WorkbookType** as specified in section [2.2.5.4](#).

saveOptions: A combination of 0 or more values of **SaveOptions** (section [3.1.4.25.4.1](#)). MUST be present.

3.1.4.25.2.2 SaveWorkbookCopyResponse

The **SaveWorkbookCopyResponse** element specifies the result data for the **SaveWorkbookCopy** WSDL operation.

```

<xs:element name="SaveWorkbookCopyResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.25.3 Complex Types

None.

3.1.4.25.4 Simple Types

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

| Simple type | Description |
|-------------|--|
| SaveOptions | An enumeration of flags that can be specified for the SaveWorkbookCopy operation. |

3.1.4.25.4.1 SaveOptions

Namespace: http://schemas.microsoft.com/office/excel/server/webservices

An enumeration of flags that can be specified for the **SaveWorkbookCopy** operation and is defined as follows.

```
<xs:simpleType name="SaveOptions" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:list>
    <xs:simpleType>
      <xs:restriction base="xs:string">
        <xs:enumeration value="None"/>
        <xs:enumeration value="AllowOverwrite"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:list>
</xs:simpleType>
```

The following table specifies the allowable values for the **SaveOptions** simple type.

| Value | Meaning |
|----------------|--|
| None | MUST NOT be used. |
| AllowOverwrite | If this flag is not set, then if the workbookPath specified in the SaveWorkbookCopy operation corresponds to an existing file, the protocol server MUST NOT overwrite the existing file. |

3.1.4.25.5 Attributes

None.

3.1.4.25.6 Groups

None.

3.1.4.25.7 Attribute Groups

None.

3.1.4.26 SetCalculationOptions

This operation is used to set the workbook calculation mode.

The following is the WSDL port type specification of the **SetCalculationOptions** WSDL operation.

```
<wsdl:operation name="SetCalculationOptions" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:SetCalculationOptionsSoapIn"/>
  <wsdl:output message="tns:SetCalculationOptionsSoapOut"/>
</wsdl:operation>
```

</wsdl:operation>

The protocol client sends a **SetCalculationOptionsSoapIn** request message and the protocol server responds with a **SetCalculationOptionsSoapOut** response message as follows:

- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD be one of the subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

3.1.4.26.1 Messages

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

| Message | Description |
|-------------------------------------|--|
| SetCalculationOptionsSoapIn | The request WSDL message for the SetCalculationOptions WSDL operation. |
| SetCalculationOptionsSoapOut | The response WSDL message for the SetCalculationOptions WSDL operation. |

3.1.4.26.1.1 SetCalculationOptionsSoapIn

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

The SOAP action value is:

`http://schemas.microsoft.com/office/excel/server/webservices/SetCalculationOptions`

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

3.1.4.26.1.2 SetCalculationOptionsSoapOut

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

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

3.1.4.26.2 Elements

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

| Element | Description |
|--------------------------------------|--|
| SetCalculationOptions | The input data for the SetCalculationOptions WSDL operation. |
| SetCalculationOptionsResponse | The result data for the SetCalculationOptions WSDL operation. |

3.1.4.26.2.1 SetCalculationOptions

The **SetCalculationOptions** element specifies the input data for the **SetCalculationOptions** WSDL operation.

```

<xs:element name="SetCalculationOptions" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="calcType"
type="tns:WorkbookCalculation"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operation, but MAY instead be obtained in any other way.

calcType: The calculation mode value as specified in **WorkbookCalculation**.

3.1.4.26.2.2 SetCalculationOptionsResponse

The **SetCalculationOptionsResponse** element specifies the result data for the **SetCalculationOptions** WSDL operation.

```

<xs:element name="SetCalculationOptionsResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.26.3 Complex Types

None.

3.1.4.26.4 Simple Types

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

| Simple type | Description |
|----------------------------|---|
| WorkbookCalculation | The WorkbookCalculation type is an enumeration of a set of options that specify how the protocol server calculates the workbook. |

3.1.4.26.4.1 WorkbookCalculation

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

The **WorkbookCalculation** type is an enumeration of a set of options that specify how the protocol server calculates the workbook. This type is defined as follows.

```

<xs:simpleType name="WorkbookCalculation" xmlns:xs="http://www.w3.org/2001/XMLSchema">

```

```

<xs:restriction base="xs:string">
  <xs:enumeration value="Manual"/>
  <xs:enumeration value="Automatic"/>
  <xs:enumeration value="Semiautomatic"/>
</xs:restriction>
</xs:simpleType>

```

The following table specifies the allowable values for the **WorkbookCalculation** simple type.

| Value | Meaning |
|---------------|--|
| Manual | Specifies workbook is in manual calculation mode . |
| Automatic | Specifies workbook is in automatic calculation mode . |
| Semiautomatic | Specifies workbook is in automatic calculation mode, except for data tables (1). |

3.1.4.26.5 Attributes

None.

3.1.4.26.6 Groups

None.

3.1.4.26.7 Attribute Groups

None.

3.1.4.27 SetCell

This operation is used to set a value to a specific worksheet cell in the workbook.

The following is the WSDL port type specification of the **SetCell** WSDL operation.

```

<wsdl:operation name="SetCell" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:SetCellSoapIn"/>
  <wsdl:output message="tns:SetCellSoapOut"/>
</wsdl:operation>

```

1. The row (2) and column (2) fields on the sheets (2) specified by **sheetName** specify the location of the worksheet cell to set.
2. **cellValue** MUST either be empty and have the **xsi:nil** attribute set to **true** or be one of the following types: **boolean**, **dateTime**, **decimal**, **double**, **float**, **int**, **short**, **string**, or **unsigned byte**.
 1. If the value is of type **string**, it MUST NOT be longer than 32,767 characters.
 2. If the protocol server does not support the value it MUST return a SOAP fault.
 3. If the protocol server does not support setting a value in the specified location, it MUST return a SOAP fault.

3. If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section 2.2.9.2, but MAY instead contain any subcode specified in section 2.2.9.1 or any other subcode.

| Subcode |
|----------------------|
| FormulaParseFailed |
| InvalidSheetName |
| CannotEditPivotTable |

3.1.4.27.1 Messages

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

| Message | Description |
|----------------|--|
| SetCellSoapIn | The request WSDL message for the SetCell WSDL operation. |
| SetCellSoapOut | The response WSDL message for the SetCell WSDL operation. |

3.1.4.27.1.1 SetCellSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/SetCell
```

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

3.1.4.27.1.2 SetCellSoapOut

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

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

3.1.4.27.2 Elements

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

| Element | Description |
|-----------------|--|
| SetCell | The input data for the SetCell WSDL operation. |
| SetCellResponse | The result data for the SetCell WSDL operation. |

3.1.4.27.2.1 SetCell

The **SetCell** element specifies the input data for the **SetCell** WSDL operation.

```

<xs:element name="SetCell" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="row" type="xs:int"/>
      <xs:element minOccurs="1" maxOccurs="1" name="column" type="xs:int"/>
      <xs:element minOccurs="0" maxOccurs="1" name="cellValue"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operations, but MAY instead be obtained in any other way.

sheetName: The name of the sheet (2) that contains the worksheet cell to set. See section [2.2.5.3](#). MUST be present. MUST NOT be empty.

row: The row (2) index of the worksheet cell. MUST be greater than or equal to 0 and less than or equal to 1,048,575.

column: The column (2) index of the worksheet cell. MUST be greater than or equal to 0 and less than or equal to 16,383.

cellValue: The value to set in the worksheet cell.

3.1.4.27.2 SetCellResponse

The **SetCellResponse** element specifies the result data for the **SetCell** WSDL operation.

```

<xs:element name="SetCellResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.27.3 Complex Types

None.

3.1.4.27.4 Simple Types

None.

3.1.4.27.5 Attributes

None.

3.1.4.27.6 Groups

None.

3.1.4.27.7 Attribute Groups

None.

3.1.4.28 SetCellA1

This operation is used to set a value to a specific worksheet cell in the workbook.

The following is the WSDL port type specification of the **SetCellA1** WSDL operation.

```
<wsdl:operation name="SetCellA1" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:SetCellA1SoapIn"/>
  <wsdl:output message="tns:SetCellA1SoapOut"/>
</wsdl:operation>
```

The protocol client sends a **SetCellA1SoapIn** request message, and the protocol server responds with a **SetCellA1SoapOut** response message, as follows:

1. The sheet (2) that contains the worksheet cell that will be set is determined in the following manner:
 - If **rangeName** specifies a sheet (2), that sheet (2) MUST be used.
 - Otherwise, the sheet (2) specified in **sheetName** MUST be used, as specified in section [2.2.5.3](#).
 - If neither **sheetName** nor **rangeName** specify a sheet (2), the protocol server MAY [<24>](#) use any sheet (2) in the workbook.
2. If **sheetName** is empty and **rangeName** specifies a sheet (2), as defined in section [2.2.5.2](#), the protocol server MUST use the sheet (2) that is specified by **rangeName** to locate the range.
3. **cellValue** MUST either be empty and have the xsi:nil attribute set to true or be one of the following types: **boolean**, **dateTime**, **decimal**, **double**, **float**, **int**, **short**, **string**, or **unsigned byte**.
 1. If the value is of type **string**, it MUST NOT be longer than 32767 characters.
 2. If the protocol server does not support the value, it MUST return a SOAP fault.
 3. If the protocol server does not support setting a value in the specified location, it MUST return a SOAP fault.
4. If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|---------------------------|
| FormulaParseFailed |
| InvalidSheetName |

| |
|----------------------------------|
| Subcode |
| DimensionAndArrayMismatch |
| SheetRangeMismatch |
| RangeParseError |

3.1.4.28.1 Messages

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

| Message | Description |
|-------------------------|--|
| SetCellA1SoapIn | The request WSDL message for the SetCellA1 WSDL operation. |
| SetCellA1SoapOut | The response WSDL message for the SetCellA1 WSDL operation. |

3.1.4.28.1.1 SetCellA1SoapIn

The request WSDL message for the **SetCellA1** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/SetCellA1
```

The SOAP body contains the **SetCellA1** element.

3.1.4.28.1.2 SetCellA1SoapOut

The response WSDL message for the **SetCellA1** WSDL operation.

The SOAP body contains the **SetCellA1Response** element.

3.1.4.28.2 Elements

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

| Element | Description |
|--------------------------|--|
| SetCellA1 | The input data for the SetCellA1 WSDL operation. |
| SetCellA1Response | The result data for the SetCellA1 WSDL operation. |

3.1.4.28.2.1 SetCellA1

The **SetCellA1** element specifies the input data for the **SetCellA1** WSDL operation.

```
<xs:element name="SetCellA1" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

```

    <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="rangeName" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="cellValue"/>
  </xs:sequence>
</xs:complexType>
</xs:element>

```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operations, but MAY instead be obtained in any other way.

sheetName: The name of the sheet (2) that contains the worksheet cell to set. See section [2.2.5.3](#). MUST be present. MUST NOT be empty.

rangeName: The range of the worksheet cell to set. See section [2.2.5.2](#). MUST be present.

cellValue: The value to set in the worksheet cell.

3.1.4.28.2.2 SetCellA1Response

The **SetCellA1Response** element specifies the result data for the **SetCellA1** WSDL operation.

```

<xs:element name="SetCellA1Response" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.28.3 Complex Types

None.

3.1.4.28.4 Simple Types

None.

3.1.4.28.5 Attributes

None.

3.1.4.28.6 Groups

None.

3.1.4.28.7 Attribute Groups

None.

3.1.4.29 SetParameters

This operation sets values of workbook parameters in a workbook.

The following is the WSDL port type specification of the **SetParameters** WSDL operation.

```
<wsdl:operation name="SetParameters" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:SetParametersSoapIn"/>
  <wsdl:output message="tns:SetParametersSoapOut"/>
</wsdl:operation>
```

The protocol client sends a **SetParametersSoapIn** request message, and the protocol server responds with a **SetParametersSoapOut** response message, as follows:

- For each [ParameterInfo](#) (section [3.1.4.29.3.2](#)) in **parameters**, if the workbook parameter is a PivotTable page field:
 - If the number of elements in the **Values** array of ParameterInfo is equal to or greater than 1, the protocol server uses the values as the PivotTable **manual filter**.
 - If the number of element in the **Values** array of ParameterInfo is 0, the protocol server changes the PivotTable manual filter to the **default filter value**.
- For each ParameterInfo in **parameters**, if the workbook parameter is a **slicer** page filter:
 - If the number of elements in the **Values** array of ParameterInfo is equal to or greater than 1, the protocol server uses the values as the slicer manual filter.
 - If the number of elements in the **Values** array of ParameterInfo is 0, the protocol server changes the slicer manual filter to the default filter value.
- For each ParameterInfo in **parameters**, if the workbook parameter is neither a slicer page filter nor a PivotTable page filter:
 - If the number of elements in the **Values** array of ParameterInfo is not exactly 1, the protocol server ignores all elements in the array.
 - If the number of elements in the **Values** array of ParameterInfo is exactly 1, the protocol server applies that value to the parameter.
- If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|---------------------------------|
| FormulaEditingNotEnabled |
| InvalidSheetName |
| ParameterDoesNotExist |
| SetParameterFailure |

3.1.4.29.1 Messages

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

| Message | Description |
|-----------------------------|--|
| SetParametersSoapIn | The request WSDL message for the SetParameters WSDL operation. |
| SetParametersSoapOut | The response WSDL message for the SetParameters WSDL operation. |

3.1.4.29.1.1 SetParametersSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/SetParameters
```

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

3.1.4.29.1.2 SetParametersSoapOut

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

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

3.1.4.29.2 Elements

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

| Element | Description |
|------------------------------|--|
| SetParameters | The input data for the SetParameters WSDL operation. |
| SetParametersResponse | The result data for the SetParameters WSDL operation. |

3.1.4.29.2.1 SetParameters

The **SetParameters** element specifies the input data for the **SetParameters** WSDL operation.

```
<xs:element name="SetParameters" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="parameters"
type="tns:ArrayOfParameterInfo"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

sessionId: A session identifier (2). MUST be present. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operations, but MAY instead be obtained in any other way.

parameters: An array of **ParameterInfo** elements (section [3.1.4.29.3.2](#)) to be set in the workbook, as specified in section [3.1.4.29.3.1](#). MUST be present.

3.1.4.29.2.2 SetParametersResponse

The **SetParametersResponse** element specifies the result data for the **SetParameters** WSDL operation.

```
<xs:element name="SetParametersResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

status: An array of **Status** elements as specified in section [2.2.4.2](#).

3.1.4.29.3 Complex Types

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

| Complex type | Description |
|-----------------------------|---|
| ArrayOfParameterInfo | An array of ParameterInfo elements. |
| ParameterInfo | Represents a value or a list of values to be set into a workbook parameter in the workbook. |

3.1.4.29.3.1 ArrayOfParameterInfo

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

An array of **ParameterInfo** elements defined as follows. The number of element in the **ArrayOfParameterInfo** element MUST be less than or equal to 255.

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

ParameterInfo: A workbook parameter as specified in section [3.1.4.29.3.2](#).

3.1.4.29.3.2 ParameterInfo

Namespace: <http://schemas.microsoft.com/office/excel/server/webservices>

Represents a value or a list of values to be set into a workbook parameter in the workbook and is defined as follows.

```
<xs:complexType name="ParameterInfo" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
    <xs:element minOccurs="0" maxOccurs="1" name="Values" type="tns:ArrayOfString"/>
  </xs:sequence>
</xs:complexType>
```

Name: The name of the workbook parameter. MUST NOT exceed 255 characters.

Values: An array of string values to set in the workbook parameter as specified in section [2.2.4.3](#).

3.1.4.29.4 Simple Types

None.

3.1.4.29.5 Attributes

None.

3.1.4.29.6 Groups

None.

3.1.4.29.7 Attribute Groups

None.

3.1.4.30 SetRange

This operation is used to set values to a range of worksheet cells on a workbook.

The following is the WSDL port type specification of the **SetRange** WSDL operation.

```
<wsdl:operation name="SetRange" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:SetRangeSoapIn"/>
  <wsdl:output message="tns:SetRangeSoapOut"/>
</wsdl:operation>
```

1. The **rangeCoordinates** on the sheet (2) specified by **sheetName**; specify the location of the worksheet cells to set.
2. If the user does not have the Open Item permission on the workbook, **rangeValues** SHOULD [<25>](#) contain only one item.
3. Each element inside the **rangeValues** element MUST be of type **ArrayOfAnyType** and be a row (2) of worksheet cells. The number of rows (2) in **rangeValues** MUST be equal to the number of rows (2) specified in the **rangeCoordinates** element.
4. Each element inside each of the **ArrayOfAnyType** type nested inside **rangeValues** MUST be a single worksheet cell. The number of elements inside each such nested **ArrayOfAnyType** MUST be equal to the number of columns (2) specified in the **rangeCoordinates** element.

5. All worksheet cells in the **rangeValues** element MUST either be empty and have the xsi:nil attribute set to true or one be of the following types: **boolean**, **dateTime**, **decimal**, **double**, **float**, **int**, **short**, **string**, or **unsigned byte**.
 1. If the value is of type **string**, it MUST NOT be longer than 32,767 characters.
 2. If the protocol server does not support the value, it MUST return a SOAP fault.
 3. If the protocol server does not support setting a value in the specified location, it MUST return a SOAP fault.
6. If there was an application error on the protocol server during the operation, the protocol server MUST return a SOAP fault response, which SHOULD contain one of the following subcodes or any other subcodes specified in section [2.2.9.2](#), but MAY instead contain any subcode specified in section [2.2.9.1](#) or any other subcode.

| Subcode |
|----------------------------------|
| FormulaParseFailed |
| InvalidSheetName |
| DimensionAndArrayMismatch |
| SheetRangeMismatch |
| RangeParseError |

3.1.4.30.1 Messages

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

| Message | Description |
|------------------------|---|
| SetRangeSoapIn | The request WSDL message for the SetRange WSDL operation. |
| SetRangeSoapOut | The response WSDL message for the SetRange WSDL operation. |

3.1.4.30.1.1 SetRangeSoapIn

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

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/SetRange
```

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

3.1.4.30.1.2 SetRangeSoapOut

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

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

3.1.4.30.2 Elements

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

| Element | Description |
|-------------------------|---|
| SetRange | The input data for the SetRange WSDL operation. |
| SetRangeResponse | The result data for the SetRange WSDL operation. |

3.1.4.30.2.1 SetRange

The **SetRange** element specifies the input data for the **SetRange** WSDL operation.

```
<xs:element name="SetRange" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="rangeCoordinates"
type="tns:RangeCoordinates"/>
      <xs:element minOccurs="0" maxOccurs="1" name="rangeValues" type="tns:ArrayOfAnyType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operations, but MAY instead be obtained in any other way.

sheetName: The name of the worksheet that contains the range to set. See section [2.2.5.3](#). MUST be present. MUST NOT be empty.

rangeCoordinates: The range to set. See section [2.2.5.2](#). MUST be present. MUST NOT be empty.

rangeValues: The values to set in the range specified by **rangeCoordinates**. MUST be present.

3.1.4.30.2.2 SetRangeResponse

The **SetRangeResponse** element specifies the result data for the **SetRange** WSDL operation.

```
<xs:element name="SetRangeResponse" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.30.3 Complex Types

None.

3.1.4.30.4 Simple Types

None.

3.1.4.30.5 Attributes

None.

3.1.4.30.6 Groups

None.

3.1.4.30.7 Attribute Groups

None.

3.1.4.31 SetRangeA1

This operation is used to set values to a range of worksheet cells on a workbook.

The following is the WSDL port type specification of the **SetRangeA1** WSDL operation.

```
<wsdl:operation name="SetRangeA1" xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:input message="tns:SetRangeA1SoapIn"/>
  <wsdl:output message="tns:SetRangeA1SoapOut"/>
</wsdl:operation>
```

1. The sheet (2) that contains the range that will be set is determined in the following manner:
 - If **rangeName** specifies a sheet (2), that sheet (2) MUST be used.
 - Otherwise, the sheet (2) specified in **sheetName** MUST be used, as specified in section [2.2.5.3](#).
 - If neither **sheetName** nor **rangeName** specify a sheet (2), the protocol server MAY [26](#) use any sheet (2) in the workbook.
2. If **sheetName** is empty and **rangeName** specifies a sheet (2), as defined in section [2.2.5.2](#), the sheet (2) that is specified by **rangeName** MUST be used to locate the range.
3. Each element inside the **rangeValues** element MUST be of type **ArrayOfAnyType** and be a row (2) of worksheet cells. The number of rows (2) in **rangeValues** MUST be equal to the number of rows (2) specified in the **rangeName** element.
4. Each element inside each of the **ArrayOfAnyType** type nested inside **rangeValues** MUST be a single worksheet cell. The number of elements inside each such nested **ArrayOfAnyType** MUST be equal to the number of columns (2) specified in the **rangeName** element.
5. All worksheet cells in the **rangeValues** element MUST either be empty and have the xsi:nil attribute set to true or be one of the following types: **boolean**, **dateTime**, **decimal**, **double**, **float**, **int**, **short**, **string**, or **unsigned byte**.

- If the value is of type **string**, it MUST NOT be longer than 32767 characters.
 - If the protocol server does not support the value, it MUST return a SOAP fault.
 - If the protocol server does not support setting a value in the specified location, it MUST return a SOAP fault.
6. If the user does not have the Open Item permission on the workbook, **rangeValues** SHOULD [<27>](#) contain only one worksheet cell.

3.1.4.31.1 Messages

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

| Message | Description |
|--------------------------|---|
| SetRangeA1SoapIn | The request WSDL message for the SetRangeA1 WSDL operation. |
| SetRangeA1SoapOut | The response WSDL message for the SetRangeA1 WSDL operation. |

3.1.4.31.1.1 SetRangeA1SoapIn

The request WSDL message for the **SetRangeA1** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/excel/server/webservices/SetRangeA1
```

The SOAP body contains the **SetRangeA1** element.

3.1.4.31.1.2 SetRangeA1SoapOut

The response WSDL message for the **SetRangeA1** WSDL operation.

The SOAP body contains the **SetRangeA1Response** element.

3.1.4.31.2 Elements

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

| Element | Description |
|---------------------------|---|
| SetRangeA1 | The input data for the SetRangeA1 WSDL operation. |
| SetRangeA1Response | The result data for the SetRangeA1 WSDL operation. |

3.1.4.31.2.1 SetRangeA1

The **SetRangeA1** element specifies the input data for the **SetRangeA1** WSDL operation.

```
<xs:element name="SetRangeA1" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
```

```

<xs:sequence>
  <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
  <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
  <xs:element minOccurs="0" maxOccurs="1" name="rangeName" type="xs:string"/>
  <xs:element minOccurs="0" maxOccurs="1" name="rangeValues" type="tns:ArrayOfAnyType"/>
</xs:sequence>
</xs:complexType>
</xs:element>

```

sessionId: A session identifier (2). MUST be present in the request message. SHOULD have a value that has been obtained from a previous call to the **OpenWorkbook**, **OpenWorkbookEx**, **OpenWorkbookForEditing**, or **NewWorkbook** operations, but MAY instead be obtained in any other way.

sheetName: The name of the sheet (2) that contains the range to set. See section [2.2.5.3](#). MUST be present.

rangeName: The name of the range to set. See section [2.2.5.2](#). MUST be present.

rangeValues: The values to set in the range specified by **rangeName**. MUST be present.

3.1.4.31.2.2 SetRangeA1Response

The **SetRangeA1Response** element specifies the result data for the **SetRangeA1** WSDL operation.

```

<xs:element name="SetRangeA1Response" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

status: An array of status information as specified in section [2.2.4.2](#).

3.1.4.31.3 Complex Types

None.

3.1.4.31.4 Simple Types

None.

3.1.4.31.5 Attributes

None.

3.1.4.31.6 Groups

None.

3.1.4.31.7 Attribute Groups

None.

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

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

For an example of a system that implements this protocol, see [\[MSDN-WDCAEWS\]](#).

4.1 Getting the Value of a Worksheet Cell

This example shows a simple, and typical, flow of how this protocol is used to enable a protocol client to get calculated results from a workbook. This example assumes a workbook has already been created and is stored in a location that the protocol server can load workbook files from.

In this example, a workbook is loaded on the protocol server, recalculated, a value is fetched, and the workbook is closed. The following UML diagram illustrates this example.

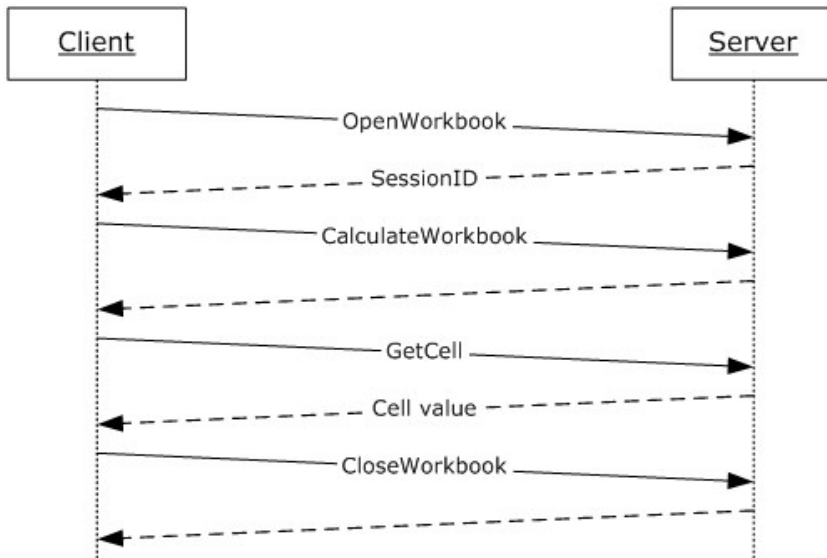


Figure 3: Getting the value of a worksheet cell example

The order of operations, including the relevant **SOAP** requests and responses for each step, is as follows:

- Load the workbook.

Request

```
<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <OpenWorkbook xmlns="http://schemas.microsoft.com/office/excel/server/webservices">
      <workbookPath>http://server/site/documents/book2.xlsx</workbookPath>
      <uiCultureName />
      <dataCultureName />
    </OpenWorkbook>
  </soap:Body>
```

```
</soap:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <OpenWorkbookResponse
xmlns="http://schemas.microsoft.com/office/excel/server/webservices">
      <OpenWorkbookResult>36.c22a23f3-976c-4933-aa85-
74c6f06c51b5164.1.V23.5791QXImNo/JutQkV4kbZDd90.5.en-US5.en-US73.+0480#0000-11-00-
01T02:00:00:0000#+0000#0000-03-00-02T02:00:00:0000#-006036.4d387825-9c48-4d48-98fe-
7894661941631.N</OpenWorkbookResult>
    </OpenWorkbookResponse>
  </soap:Body>
</soap:Envelope>
```

- Recalculate the workbook.

Request

```
<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <CalculateWorkbook xmlns="http://schemas.microsoft.com/office/excel/server/webservices">
      <sessionId>36.c22a23f3-976c-4933-aa85-
74c6f06c51b5164.1.V23.5791QXImNo/JutQkV4kbZDd90.5.en-US5.en-US73.+0480#0000-11-00-
01T02:00:00:0000#+0000#0000-03-00-02T02:00:00:0000#-006036.4d387825-9c48-4d48-98fe-
7894661941631.N</sessionId>
      <calculateType>CalculateFull</calculateType>
    </CalculateWorkbook>
  </soap:Body>
</soap:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <CalculateWorkbookResponse
xmlns="http://schemas.microsoft.com/office/excel/server/webservices" />
  </soap:Body>
</soap:Envelope>
```

- Fetch the specific value from a worksheet cell within the workbook.

Request

```
<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
```

```

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetCellA1 xmlns="http://schemas.microsoft.com/office/excel/server/webservices">
      <sessionId>36.c22a23f3-976c-4933-aa85-
74c6f06c51b5164.1.V23.5791QXImNo/JutQkV4kbZDd90.5.en-US5.en-US73.+0480#0000-11-00-
01T02:00:00:0000#+0000#0000-03-00-02T02:00:00:0000#-006036.4d387825-9c48-4d48-98fe-
7894661941631.N</sessionId>
      <sheetName>Sheet1</sheetName>
      <rangeName>A1</rangeName>
      <formatted>>false</formatted>
    </GetCellA1>
  </soap:Body>
</soap:Envelope>

```

Response

```

<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetCellA1Response xmlns="http://schemas.microsoft.com/office/excel/server/webservices">
      <GetCellA1Result xsi:type="xsd:double">5</GetCellA1Result>
    </GetCellA1Response>
  </soap:Body>
</soap:Envelope>

```

- Close the workbook session on the protocol server.

Request

```

<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <CloseWorkbook xmlns="http://schemas.microsoft.com/office/excel/server/webservices">
      <sessionId>36.c22a23f3-976c-4933-aa85-
74c6f06c51b5164.1.V23.5791QXImNo/JutQkV4kbZDd90.5.en-US5.en-US73.+0480#0000-11-00-
01T02:00:00:0000#+0000#0000-03-00-02T02:00:00:0000#-006036.4d387825-9c48-4d48-98fe-
7894661941631.N</sessionId>
    </CloseWorkbook>
  </soap:Body>
</soap:Envelope>

```

Response

```

<?xml version="1.0" encoding="utf-8"?><soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <CloseWorkbookResponse
xmlns="http://schemas.microsoft.com/office/excel/server/webservices" />
  </soap:Body>
</soap:Envelope>

```


4.2 Refreshing Data and Retrieving a Workbook

This example illustrates how this protocol is used to enable a more complex series of actions, including refreshing connected data and retrieving the resulting workbook after a calculation. This example assumes a workbook has already been created and is stored in a location that the workbook protocol server can load files from. In this example a workbook is loaded on the protocol server, some values are set into the workbook, the connected data is refreshed, the workbook is recalculated, and finally the desired result of that calculation is retrieved and the workbook is closed. The following UML diagram illustrates this example.

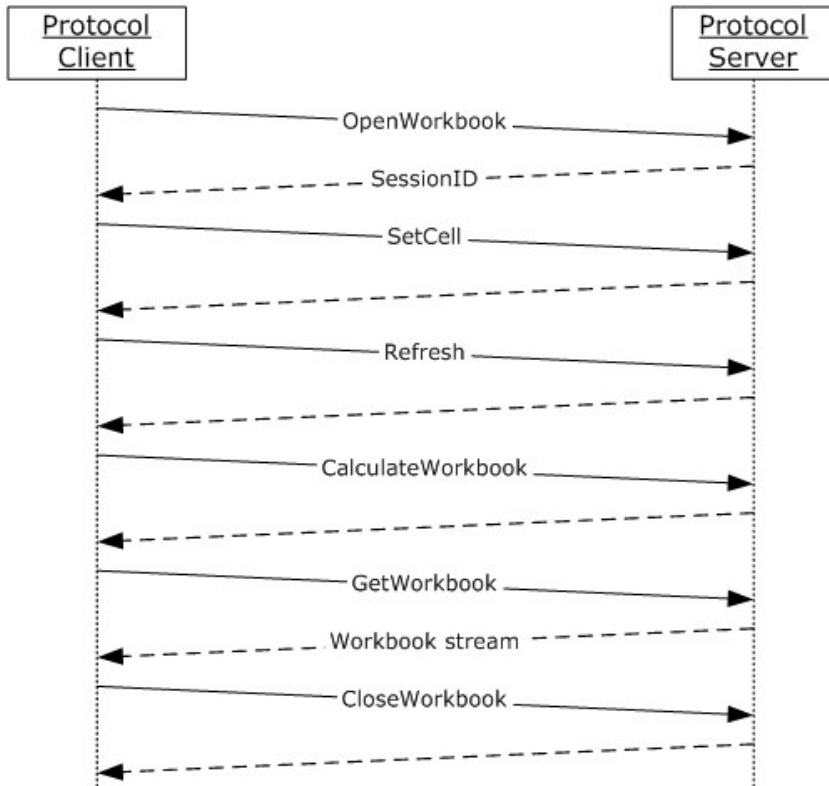


Figure 4: Refresh data and retrieve workbook example

The order of operations, including the relevant SOAP requests and responses for each step, is as follows:

- Load the workbook.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <OpenWorkbook xmlns="http://schemas.microsoft.com/office/excel/server/webservices">
      <workbookPath>http://server/site/documents/book2.xlsx</workbookPath>
      <uiCultureName />
      <dataCultureName />
    </OpenWorkbook>
  </soap:Body>
</soap:Envelope>
```

```
</OpenWorkbook>
</soap:Body>
</soap:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <OpenWorkbookResponse
xmlns="http://schemas.microsoft.com/office/excel/server/webservices">
      <OpenWorkbookResult>36.c22a23f3-976c-4933-aa85-
74c6f06c51b5164.1.V23.494r81aHjSSiziqJhYlU2B590.5.en-US5.en-US73.+0480#0000-11-00-
01T02:00:00:0000#+0000#0000-03-00-02T02:00:00:0000#+006036.4d387825-9c48-4d48-98fe-
7894661941631.N</OpenWorkbookResult>
    </OpenWorkbookResponse>
  </soap:Body>
</soap:Envelope>
```

- Set the value of the worksheet cell.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <SetCellA1 xmlns="http://schemas.microsoft.com/office/excel/server/webservices">
      <sessionId>36.c22a23f3-976c-4933-aa85-
74c6f06c51b5164.1.V23.494r81aHjSSiziqJhYlU2B590.5.en-US5.en-US73.+0480#0000-11-00-
01T02:00:00:0000#+0000#0000-03-00-02T02:00:00:0000#+006036.4d387825-9c48-4d48-98fe-
7894661941631.N</sessionId>
      <sheetName>Sheet1</sheetName>
      <rangeName>A1</rangeName>
      <cellValue xsi:type="xsd:int">5</cellValue>
    </SetCellA1>
  </soap:Body>
</soap:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <SetCellA1Response xmlns="http://schemas.microsoft.com/office/excel/server/webservices"
/>
  </soap:Body>
</soap:Envelope>
```

- Refresh the connected data in the workbook.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <Refresh xmlns="http://schemas.microsoft.com/office/excel/server/webservices">
      <sessionId>36.c22a23f3-976c-4933-aa85-
74c6f06c51b5164.1.V23.494r81aHjSSiziqJhYlU2B590.5.en-US5.en-US73.+0480#0000-11-00-
01T02:00:00:0000#+0000#0000-03-00-02T02:00:00:0000#-006036.4d387825-9c48-4d48-98fe-
7894661941631.N</sessionId>
      <connectionName>Adventure Works</connectionName>
    </Refresh>
  </soap:Body>
</soap:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <RefreshResponse xmlns="http://schemas.microsoft.com/office/excel/server/webservices" />
  </soap:Body>
</soap:Envelope>
```

- Recalculate the workbook.

Request

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <CalculateWorkbook xmlns="http://schemas.microsoft.com/office/excel/server/webservices">
      <sessionId>36.c22a23f3-976c-4933-aa85-
74c6f06c51b5164.1.V23.494r81aHjSSiziqJhYlU2B590.5.en-US5.en-US73.+0480#0000-11-00-
01T02:00:00:0000#+0000#0000-03-00-02T02:00:00:0000#-006036.4d387825-9c48-4d48-98fe-
7894661941631.N</sessionId>
      <calculateType>CalculateFull</calculateType>
    </CalculateWorkbook>
  </soap:Body>
</soap:Envelope>
```

Response

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <CalculateWorkbookResponse
xmlns="http://schemas.microsoft.com/office/excel/server/webservices" />
  </soap:Body>
```


L5+jNy+fhT9+cfz45+MnT44f/2R1ORN3cRzkJ77+7vO/vvke/fn829dPvvyWl3n533789NdfvigWBI41W9Grr579/uLZq
68/+P7pwXi20Bn8uI9GhGJ7pAjDMAj8M0A466c9MX5zVrCTJ0ZOATdBarbKnQE70wwK5JrEhe8+wLoZZHgjFFDZ63dUI
wVlB8K4cwcT3OWROoXhEAT7StHMK9cRwUGxfjvNwBxodFtLs4dkLbHiFaq70kdLbvhcRZ5j7DscIBiYlC+h4fEVLg3QN
KHVz3Mn6KH1DUxLQQkh7tO4kOm7RLI4jLpMhnCLWDzD5910SsYosdcuhKwobArGDxPcIcGG/gscJRkcoejlge8NtYUWFL
7E6En5drSWRDgjjqd0gUhbNuSvA31zQb2EopVh320TjYUUIo6Kkd7GnOclD/IOFWJ48ChAoUvJMC97U44qRTHA56pJf
I+70ORfQxxwvDdc9ylxwfnl6IbhHA2dJswTRd8ZCewGF2m/EYUHQm7CYswoVGPr/WUXbnjb0JqKtSuiRK8S4JC+80Hs
f7BHJ9vvFc1t3Luuv95+vuor181mo7K7BQezV5sLzYsORoIUKeUsa6asLiBw14soRmMejAoJ5njhdJ9FgtCeFjWtwduUB
gMwcJrj6mKuyGOAGOXfG0kkCmqgOJEI7h9M8MF+rW8sDT1T07rOmnLVsPJFZ7fGCHV/WwGQdHp2pMywnMCWvmaFURoKux
1fVUKeh8G2MvVagzW6uYpZlS5libulzoGgx00QQWgoC7AMprcLCoTcOzCWZkoHG3DTgLi4nCRYZIhnhA0hhpv+djVDFBy
nLFHBd7hTESCB+EnoJazlpdq30Ha2cJUt5cdYG5LHrvEqUsg2dRAMontyOL85uTxeio4dVrKzUP+ThpeEN4rIWPUQJRl5
r4YRbA2wNfCzV2p25ms8tn0axnjrmboAKHhnb30YedOpAIqXawDG1qmFtpCrBYW7LrX6kBrBflgm30t1jF6gYkw7+2CsD
RDS0ZDomv8sH0jWjs7GvASvLYEDENB0eoz8biAEP4daqCFwMq4XTCVAR9AS9bNNrmluc002XP6g3cdJfT4Q0lBg1oQ4
7QW6fmRlxvpiisUIHOVvw6ALwTWIH9+nE09+j/gnC8Alzhf3GvAk/18Xpw1A4JTONWBVuTDKlYBks7DDY8LFXJoLk1I/
Y4APmhaAhQBeA8L7j2Gw4tk81+QQ/3f1lKrQ2tj8DCvDmiABAWaoUJBYD50G1NUTLFWSSmJVZkYpMnsxt1yZ2GX3ySfHpd
3a1jRl81AIFcw0ibS6G7mTO9e9TgtjP9DcNV9GndYwZyU2evzThNbWaHDKba+Gp2b4T5dYQJbsfDM9o1R5R/SNGXuuZqk
ExnIdvp4WzLdcwjKz1G1EcX6v1LLFQRtNpYbBK+ft/Eh0n+Al1DhM/utBM2TevwAWiaCLxloZ2A2kNVXLJ9Euu/ZwT7w
YfTtok0mrstCmjFijlnGwC36Amdo9AbZe2VnifUw6p5zbnfSxYsE00XYwdqOLYQaIntyi8LQMhs+NYEX2fJf+OE9x3Co
Hfgzc2YKwMSCB58L7jA8UXTNPOdNby2aqvt/AuAA//8DAFBLAWQAAAYACAAACEAs6QA6YACACBYAAACAAAHs3L3dvcmt
zaGvLDHMc2h1ZXQxLnhtbIyVS2/imBCA7yvtf7B8L014NAERqqgElSNkq+0+zsazEKtJnLUNtP9+XzYJDQ+pb8Aez3z
zsGeYp71VJtMaOKLWKYOGISVQc5mJepfS37/WdWk12rA6Y6W6TaXvoOnt4uX+VGqV10AGIKEWqe0MKaZBYHmBVRMD2QD
NZ7kULXM4FbtAtOoYJkzqspgGIApQCvETT1hpj7DkHkuOKwK31dQGW9RUDKD8etCNLq1VfwzuIqp133zvwGXVIGIrSmHeH
ZSSiS2Ul0qtiOx7dozHjLdpsrfCW4klrmZoc4wad6nXMSDAMLeZwAxs2YmCPKXP0WwVkm30j79fGe+M17jRI+TO3Q7wK4rY6V9+unQ17F7jR
rgBjK8J0ps/bdSvlrFDYpCRGqNjYJGMG3GAJZR1StcTvMJ/zgku0UHQefi4br2t3Y39UCSDnO1L81Mev4HYFQbdOnMus9T
Fb1JfDMjLBV7c79HkZk1eLBJA5HORadbOGbtbC2lPC9NrL6e1KyKXsQ8QmCvy1k811I4AnySa2YYu5kkeCwld6obZ
txNxlgeboXPEa40i8k8OGD2HD+o3ZlGmtcmSytN6ZASzEM787AZd9AVanRnUXfWQ2NqN9Bw6mrXmQ87c4/2Gj7q0WR6h
27v+CrXPZ26aMLutfw9GGS36E/3qRbaZ8+vWm30j79fGe+M17jRI+TO3Q7wK4rY6V9+unQ17F7jR
9G58L1Yp/epFtpnx5f0L2Gp8dJNO2Oe3QcCLEcd+I+P+kAvjYnlbb04+R8Nd6DHwK+Xxq2g+9M7UstSQk5PsVwgmVVvuv
d2sjGSW1Ls4Nt304KHOMAFRUosAdyKU27wfa23Bcw+4ZlJXBcuymd0kYqo5gwg6GEmSpqTtZ0c6n7T1n8BwAA//8DAFBL
AWQAAAYACAAACEALCPW3KgCAA2BgAADQAAAHsL3N0eWxlcy54bWykVN9r2zAQfh/sfxDK61I1lbpY0wXZJMGKFRgzaw
V51w3bE9MNIckk69r/vJDuOqx/a0ZfkdD59d/d9p4tvd1KbZ2Ys1yrB06sJRkzluuCqSvDpp934BiPrqCqo0Tol+Mgsvk
0/f4qtOwr2uGfMIYBQNsF75+oVITbfm0ntla6Zgi+1NpI60JqK2NowWlh/SQoSSTzZiilXuEVYyfW9IJKa3009zrWsqem
ZF9wAaxZGM1/dVUobmko9TcD0fyEHQ6v4CXPjba6dfCAR3RZ8py9rVKGRA9Q0lglciedRblulAO2ehdqv9w4JzPMGqb
3uoCYhh9Gy1GmKQx6a6ncanVJYp3pLF9Qc9UAMTUH+daaIMckAkgwaOoZG3ElgqeGe7DSiq50LbuyDsC/12c5MBGyN1mI
CEzX0Jq9ZLE0I2Fj13kHd+jWASLviASjZzk/Dz71PSt47t0cPw6nTgMSRBLP3r+KLCTgpus+ohk16Po/pscV7dnl+Qh17
wzNhxJTizG/RQ87U5NUKazlovCrZrVwRdo28n2sxnZTnvHjb5T6y2ul4irraFRdvoettqDnrVy66fSLXQ2fXiZ7h5AN8
PLMZe/NLvLFu5nmwPj526H7MaoZOeurSinnb9JWZM5fRCqyUy9hng9gIYl1ddnWnVI+I+eG+AK+TQvJiUXGm0QmJyXqW
F6ZeJNCiLYLN7z+afwMAAP//AwBQSWMEFAAGAAAGAAAHAJCVG1V8JwAARooBACCAAA4bc9waxZvdENhY2hL3Bpdm90Q
2FjaGEZwZpbml0aw9uMS54bWzcXVt3Dayfj9V5z+wH5f1rJkx052LW85kpWk1o69lpOtLW89oGdoiZuLl3Bpdm90Q
+DIMBuoAECGG048YutQV/Q+NBo3MFnf/uyWmafi3pbrqv2emDh70sQobrRVndnM9+eX/1529n2bbJq0W+XFFf+eyu2M7
+9vx//+fZpvy8bi7y+WlXWwqg7IBBRkoq7bns9um2fz15GQLxFW+fbDeFBVQPq3rVd7Az/rmZLupi3yxvS2KZR80Xv4
8MnJKi+rmdTwlzEx/rTp3JeXK7nu1VRNVJXXSxzYcn2t1xswfL8c3GZN/n5DMpVF5/qArJcFH93Pnv5ZV4s+ui/gw6g
FMTgR2K+ei7756cPnjy80Hjb0+/+e7p2el3s+xjPv/tp17rcsU/dkUNOk5n2RyKORSLXxWAj5Amnfh4l3Q6p3MPv+4Ld
S1Zz+v68UFqG1aK7e7zWZdn9e7j//RuXRpLxaf82peLC7vqrkU2T9/NhcVcL3e1fMia+42YHnxpSngK1+CbeuqKuYcJz8
WgvukY78qi+ViC+Q2y7NvlJo2PavyFWj58LrItzuA698PPlzst16VdRza+S/QXG+EVrP25Qyy6rd6mrViNwA7tuyqPN6
fgtAnZ2ezbJl8bkaqx+dPX3y1Nqi81TqcJ4/FoUbOt/c3o1EYXF9RyyQSe6sod66jAvC29u8LhY/NcVK1/+JSM4+Q5FDs
3/wp3y1+eUHfWBLnS/LXNhZPtM/RdmSFF7kVb7otMm/k1VldcU6nTgMSRBLP3r+KLCTgpus+ohk16Po/pscV7dnl+Qh17
pH0/ZVfd1Ae9wSztJJKD5B9Q+/2sbTejlugaHe2Kr989t6/bnssFbnHylJcFPPM8s2qgXC4vS5W5evS7F4D20bAis0K2j
VktTUbYx+2BZsv6Jcl1AltVfA7xOxH05r9dr1t8iWEPUXr6gp+1BxSiMfTFuJqt1xmP0PQFT6riqATQwogAjhy7N5DDuM8
1NsHwvSmhsFA3dyJAKP/ER69oQNZFauPRQONrtXQtlvZCU5Y+Lb1PPhgYIBqklJmGQXAPLo60L4XgxHDMeKa2pzwADae0
K1LnhDvFoOnI6x43UKhuC8WCXgptZ2OasBdklnfOnm4or85/nJ/X9nNbSYGzDh29alm6Th1qWdYr1a7Zoiuyy3MEWSAY
71ASNBMOmfXWQCG/I1p+yq7LeNt1bmJjc51viHzYHCY+L66As6eViN2+nqbjd6EQTCkT4OkOpm/11xgTN15hgOCJXWU
SPxTVoqiX8gUs+wq9aso9I+wlpC9+b2iBe9TzcJjylcBwOu8LsW8RQz+dmTUQCkmECblqWdJ550Y1Ise4yKvt61fLHCD
4OgmMDzPlwbPbbmEYwVvWgy0SQCIBlChggszq8CsDfz+W5j9a19qgkIpnwVALy9hXV67BxtglnsLvHrKHHAMhCde6Lza
sgU9A8wFXu/blfCujanaqV59STD8wqV+FO/yryOvlXfa0Y7TGPROEKIjHqG8UiQIvU226V+sduY9r5C9Gbo10xFNsuAivqXb
arSthCkittH17M2/0esVrR/YUWqbgkWZY3sKr8Ea07ebWb8XCD5KwKD71lu2Xzu12PgxV73rICBY0811i1JmV17EixJ
29sTOJCs4yxblDRPM767WSxhGn8/E4rjarTtVH/Nlu1smVs3V1lcoNnNp7bAET6OuJB3FhCoCGaU7HiCn5HQ4ix0JDiWR
vg9Grd4EhHi5yfaHo3vVfSbVw9sRbMDK7cbHTwHnkQrS7uU7GgTaCmXm6Xovk4YLd6kYdbJuFYGLD4qGF1htIR4QLWqFi
FdrufCSHCq0atcqdWIm3SaCiJdK/QagDolJ0YhaC01ByczA9gazi1Foii2JYelY9C6L1TybUM5HcSe5GYajqd4fs4JLhp

[MS-ESP] — v20140721
Excel Web Services Protocol

Copyright © 2014 Microsoft Corporation.

Release: July 31, 2014

aIRMk1FAvOgA2tO4BCoOnSzE4mYsyCvUPLIEIeWwyoWrguxqz9vtqM4UmdWlP+dAbV6X8Ki+vojBiOkNgktr3wsbhqvJzMY7+Ih3/NNn9pzTr9s85tCDQzFYZ82Waz+OZ/9ue0LaRqTDCaGkJKTrkZ302E76RiSdIFPUURPd8/rH7agC9D4Abgc60WwIQb2x1hadbHj34xOLrcGroyJzQyBA0f2DhIhKVccBhjNnsI4Fbcxw4d6ESMC0uXyFjLMmFpWRXanfeMB+1KcmtbhenHEDz/fJKzcxUHRJAOonFKUAMURVxoA3KTgyXGWI3fxiAIZsRIGNNFDD8QBqB9gr0RlMbiSPNStLs5bo/RMDHS/byKB7VOTG0/F44Dzyk0MlGs7JqERs1VRrdYlEgVkmOMNtPmCQcQlJzkPko/zHr9GLFJjznpv6VAMFqUk4UvVsEE2KdsLanRLC6NGKUmORURVqA3KxqI28niXbIbH3AghCTaiIQ61IdEpQB0HE6EUF/kSDnD15ACXID1QhFnWwA5Wu5fmno6bymSAfl1vU5XrBLQMP8fOAtFLWcnhvaOLOmWFM9o9dXsP9MzGkFRWHvUiwamRMRtOr4nEzNXZeFozjoPykuMJFoGIBkKzJOQKypspoaAcvtIqt31EFdQMM20wim7zZqQbQFFh10UMvIrdtWfzjRCziHb5V/ubD959YhhxstCvxvDDB63RhulMnrJb/my117V7hpr9E+NdeTB44Q6ELcWZe5BOnBZS4JTSAGOWilIbT1Of1TcaLr6+f4QF/wGrY3bq30QFioiPvhorTEY00UmQCfxfBfy54BG1vdNoMzEh1LjciHwZwWJEJsGHdc+c6+wVnpCTSdhoFkhsrtrPF9aLiOTY4adnaJA13Bwx5RcmbjfgFNqscZsaJ+zjbo8WY3pr4vWF0k+g9GRoWSM0B5wm3954mgojAR0UnwxRNz1brBf7EJKIThaO3bJrPEIKmVk0A4JnPNEQk33L6ImfCe/UIwv7Yur6IKDeGFiOkJYSyHAXpnD4FMsS3i06E5ePnasXGAevfAQ1jdiUBFLZMMAXQq6ECJiN/Dag3R35fL249iGQ2jKZHigFgxAt1UnQz6oIjgqSR68PdLwarrvTZMxqMHWn18OgpiOrCkVx5sFF7cXNuz3bMdszdGipI8D2HZe6XRmQXX6bcwykoFMN5sdWE8KEUpCEff946wgnYs02Y681J6wInNS5mDTBcV7bVUGNG3Qy5dELQb7X3G+Bypfp4Iw7RkgwBTDQr1igUwQom03quRs7Z41UKyr80fXGLWIS7pkjWJif/+DVq3GAzKp70GWWI0DaYGUU1S3M6opbBOWPwT4Xqc6tvk6xJjGUtAnLqGxJbJGy86oKLM3+UfwrV6x5Vpd7TtQzRFBbw2Qh7A3BcYD39Sciv6gmiOC3x9v03cSTOPBNBrvkSVWtRwDnHThzOnDhG0ESLk+C5ihaOGXtoIVSp52D/3SiMxkoc2JJ+YaAU6sLhpNr/ChwdAWSTxNDiU4rJQP8iVbJTsuwXXsxmquvZpZsCBVO5duyBDLXMHg7+XKOApVV+8gWNXi4VgguXgem0mXmLf7iG4Hb0zGM01huc4U8t1ewzMBQ+j1Fr3KCM9Mep50KXhDk0WYy8qsg1ikh75CiTMQXe/gCHsmPbuPonqbsXq81yfVeIA5Z9u7MPlegG3ZLMSnH7sUe9RFsxcAArGmObLcyjpl1VYbv6mvsmr8r9tu052ea3+wQd6XxEFaKJQo5oADBewKEt1fMABkOisajtFTVB2Q3FeAl1fObgRORQaensf2317niMRQzazSDAddRwXqta1GBvWlMUKXpEtGFxbg15qAutSMi2yXagkjVummdhFdFMK0Z7F6Gac1cadUrFwjRMIPtLz35/gpSFVXET2DQYV0IHSscYQG9syoWBmi2LoJ/rYryo6PFMR1MFq/jRUxVumdtK8yC4vBvQbaahpJt8fLJxS114iBMPOVAP0IhU1LBEwqihw+dAbZMLChjh7H3UUVW7f98gK3MLqAHFvJ8gN113DYWLDGuU6n8wWt+LGT4GR42N3EL0o1JfqOlmYbhQcgnFgvSiJ7TmmWuvenEdIDQ2pqqjOHeXmfK4DohQMbzSgQU0oSVQFU3CPQw3KxoI2ctN7CxcLruoCncqUKYMIN0JIgo1oiENTSDQWtLHni9aoPH0M7hose3zKJRKLypjzwWt4QrQu4YN1V8v8BvsRIST5EdEQ50dDorGIjelHYB+8Tat3uLwRUgGOUpMRY9qiUZwUHxyFMv5b9mraJ5gCc+972oyDr2aGkYGkoiIRyWnhzBBtb/rEYyVf3QpaaahpJt8fLJxS114iBMPOVAP0IhU1LBEwqihw+dAbZMLChjh7H3ZbMkY/c2IQkXqcouoqc7dI1MjCqvUxXa1w5K1JKEE1UR50eDspMg1z/BCaft2a/ihT4EZGms7qSS+Q0hrW2/niuy54R4YHpRa6Miedmg1w1oJpDcLaePK6HZzLlPkGmtw7NXHSib+1NMOE9479fWPaM+j3sKIRzmlWFP9KEUEHJ4QEHLw8pYNUgZ3OGZKfxHtGfFkGfJU6JPib22MYWIH441H6q4EGVqmgYOWF5U8NHJGALH76uvGw/e57XMPers/cdfPcdPrWsvr6A3A8+WcWxhLtiogkSdK6YH+jj+ALK6c+EryZvVuP1I95soGTR1y/fcnVmkUdr74s1V1Q4QxOqapY/clcSf5M9G4Ty+Dr/jgB5OguZusNqOGek57mBlx4w7jnavSvqtMJM2WsswFY/bLiK1/1fQrZ4Ia6w7wsKV7Kkk9V5qqMvY3wNENhncR0H6J601vJgAvqpOWVqnBAkhvC686iXsmIXrKjDdspBXSNKFI3XRX01Dz7gn2QtkxB8R0RzQoHxt4ji8iNJ7o04mB1aVOCXwwZPCbyt1/AtZ3HC+AKG3Tewp46bq0oLdwGn6zLjsB5QogMX/VKMhiPwS+pKY2AxzLfkopuE5KQaDXFWMALRGJw3azbcX/yagfCYr1kz1wLORCEhAthphgeIhpsIBLg9usWwew82r3CT556vJiIT8fLJxS114iBMPOVAP0IhU1LBEwqihw+dAbZMLChjh7H3j4LLKxeJ0xVcbajmZb4coeGpU1o4KKU0FICUBAw4bpl1YPbd9enj0ascjjlnsOPerDEyKdKfHCQeg49XLBKIEVzmFRyNs1tWn5qETK8zIgg1edpWRuIyZptCgT7D/ensQeZNN94rmdHU8ourjnxIjGNhGrEbv/1elEsrsEfwqKK/XSUSj11lgOGIPNCE1M1R6oU+165JMRCIORsuldZ70fwX3906RycDJurmDZ3K4xULUSCbWFRDJxhPiPROSP29IJMUzSwkm+7kPpzDOWAST4F1GM8TswzF/hU6UT6Pui0eql1yusM0zveY8JHh1N/WTJ61fMwD26ufRmlYx3St9om/IHM91WFI31Kbl1Jvvgqf3E3ED4/Wu2Ldbi+XZdwHx11fISQ0iKJArubucaEiZeamRho3X+qWjuslYjNID8XJB3rm1qCmq21hjoQqQnxK/8LxmUX8PvJIyEnoimJ/Kx+afFXuXVzCWGj11FZNOZyQyLqk8Wly8fwiBwXqBElMglMaPSEJJaZAdgl3TCI524KRWI056wXLRjpm3PWWUDqkM8aZekssqUiAx1tRgqLseYyWnkfIu6S/culXe7HBAfSHTB919hSkMlhgyHcB1q18zUjnc/ejWjMjr250yqer0VcUoArmqIQQYi28sIax/Xu49zZqCQJsfhgBTGO1TOzWz/yzKml1vSE8mUFEQ6XRFfuCBlcRijvazWBTbmw1894TMvSh60ue01Fy9ZozmDSYtCyZ2DRkyyOAJv1LwTwU4LxMiY6Gwsk01nbZned0GYqbQ0rJR0H3jCMLPJTYPN6/yLeL+zaowrCTg9DSOsIRanIdmJsCorHiuUnogV0hCN1YDsNFjBaE23abxMLRNN10JW93XD7jOjFVewYkBUyVz51K2L2LLWmhB6V7rtxLSeWHft0Hk1TI3he/h0I+4ie6sFZV/sWu2pLpe5pCfGzFy01T7jfeBHowpOpP9WtZ5RAUC9XP5S1fAMw8+5+Axzyz0oQ00qaEKObgl9apoH9fKXrc4veT1M3hXbYgknjocT9i0VD0IN9XVI40Iw62awUGrt0KhqywRbiIdgi4w0tUgwo2+TJ7BwvY1Y6E0pJw4cqt0tsNjG3YbZsLFryCDc8aZQnT4Rt81q1OGh3qLXp+2AmFdulHkBMJTKtXt/n1W/WYRKdmISUlu4Ck6qddgfbnE4vFqLtlN5JPfb/bio3IrextUQwnhDSsiOootyKZM1DHYjZG67sqazjnJtE33+AyaU14mUqYcntaYoB0LGrjetqr3AmfQUspcZ9ARB96w8LTYvYa5KzyGJZ5mz16sxpWjv82NqLbW00ciEHyx40jOfXB5sdNflQnAHCZStIRRLMfPv90MtPiJ6+hiQ+4nc9BoFQ07yREZPHJbHcmKxHKMKxTGVfwrPICUhz+I2Q21Y0BYxqXGs8a/Uln5+Bic/LlWHZ1Jt1CgFZcWAPs1W8nFANJ5rE5Lwwe8M41DkGyy4RGIxGaPfwqKuHA40sFOBQBUOfEc5TgDk1OgpMuFXEalMv4T4TsQWtVAbC6M8XcKtM/RyY0S5m2ERrYGM6kzMnEja4HIULwHVQJAgRqSzkG5QW12isC40bqLV1Ui3b2714YrEzs0iDcVjLUSFq9YEgmi09DcmEfpDLfIT2ek8+aY77tfvY00/eH1KMuuOU6Rumer1fPwdsbKi11gD9GVBPLGdmTVEVCxzTULLZBQShhOQZd/Bg5BEmbYl7IzThc1fKMjYNoEEQK/u6W8yHhu5xvN11VCkWhvZjzK2zErPbcU61DMFmvwqFhj32Y1jx3s+Wj31na9yqsgvml+9vs2wPdL2LXCUKz9LLDSxD5K/HlqsfoWky+7krLnftYkPKjh3dJOnW+hub60hyE3aaC3DvDwQXVydB646VbMr2v8otSOXUCXTAWw1HupoS5jf2zZo46C9E9aSxLoi9u8quC2FdxMgPdFtpn6bbWYjmbBWRsRmmzGZihrgvBwPT1hsKx2KOFxpqrG6xSk3ndFdt9C0BZ2Py1YZfreqPK5YTgkxvSGA0+Pds9B4HT2vLF0LdEE1j3qB+LGrWGJzqYLrJMn5pJQZSMTkhZfamsZghgp86IYZZ13FwqJsdserda5fVdN7Q47LvBEsFYC3AI6JYjuteOoWkYR/XGZ6FRXxxJawo4bfc7sPabF5n0mUAMouuY3UmzPKopTfryl8EYbwDvrxxrz/GatPtp0atjDtwD68ZMq7GuKppgHBvWsI4A5X+I2mF68TawE194B9eBe0EhQcX0eL4v4EF/8WVUPXLPu1CXIMumScPDIC3K6MWhRfHxh+kws03S5cNW5Srunoy1ZjloK7w6AAe/6Hu5hpBiir249+vRE0udolQCdi9g25ZmmIh6gm7jQ/+Her3b2GstPVAtw5J1yxzHxH1I4/Gh/q64ERd73aEEDH1IDhP36JUvVY92koiyGLUFdOVyV+7BqWJuS80n09ei+OpPjHqXtg/Osi3KYfAVcmTLIAH2ttzIjxyKN8iXhXi5hTiYo9Jq1FnW3Lod57Ntb7+1xABBjdi5DNxFL9cLduTeIuRwBS1q9UEqe6g/0Wwcz70s131E7Nts2Lhf58vzZP/fFj/C93ryes57B3R07exa6+7LIq701g08ogynP8UGrR8yk9v0m0RATbVJAA8q0RLAr4tVsQ1H3GQFC3JTBxrmglqOBPQA1xYsY4HbZid7gbgoIdugKTKtiHDQ11zn02HkgaA4IYvpq1FganMzAbBW5YYEohHDQXVVRd4W9a/5cleIQ84N3Jk+nz3dI9jCM/rWjTWMogQ6kfQs0BMcukziwHMI7QVgcq8kgIKOyDoIR8FSHCMAP1RFRT9ipYn2pMD9syjI+VULN8EwDmxtVmbh/e20rVtWblLQ3H1DV9Ej9AxIU4qrsf3JNKBdWVNM0X2JVAWHB5IWPa7vUmy4k0mk

GhJl8QgjaqngKGoXi2ZYBlMm7Z8N0GBAPq2mX5bFHALMl8PThbDKPA7Sp1YLJFw27yhrUcQRQu+NjV4gR4iPSlALjOPFRG
oFHgskDmXYE459TE5Y9PZLoigLQLxntfaOAJ4YIfuwvx57QYVVRyHkFDWjcegdPImbqA4xQhGsJBUdGSjKh8W0qCCTi+
xmaN60jYQ9dLwNTRaS2fPbmlottCFTY8FnR1TiZzu4hWzNabqbOCIASoiP6TW0FXQLQio1U81OMELLYJFL+sqBhPOZni
15XV0BAqVI6HrPw2XFAv1+262W25WBSVWMAeWp17V0jDjeG1mMEEJWqweDqtNKEf2IYu7FKt3BUyUjJ6QBSFpaHoDUJF
QCDVjeh/nW3aIVDHJQe+zm5mi r43dEpRZH20sUnBkMl8YCG5XE9KR2yXkcoWh6JGKGB8a9iWx4u4AyqGUK4No5Pr+k0s
EIA TWvulQ8N4mrXaEevUXO//QnJChVZNX3ctDFz7hiTXfzrBEjrnSmQv+7UCsYSG4HeCTjlgWfTqjAa6GhXJyJzlVuuIJJ
212i2GYfAM4+GsNKYC7ZQ/MqRRd+IEuecZAd9eWTS0HtEkVMfx3A0c5BBvYUfXgLv9g4CpUk34IiakSDXJRw7Seyp3Lqa
nUhs5WReQgggwCgo+OhQ+U1IPoIF91+3dP32ALPs7/bwmQzZRDs+ndGQj8VBMHJKK5rDTVsDDqdVYYL6pb/Kq/G8uPsAx
hmv+UKxv6nxzK04U679NTAnBRDPCR7UeKzMJjKLz61OKapvKSFPA9josZMDA86cKdHBVReO4GM2R8ayBo+8BN6cOXDfMy
FAx4twrxR+LcneP3Lwe+2rz53EtN4nyGoJCFKHTj9D1qCqsxc5M1gjm4msD0MA080kTPvScRxnIAK3VrjAo47vDtVC0xVG
QOP0WzrLJFEEC0gnsFzRFYVOaYjAMLR9/DXT11qN1zZMBVBA05KsOXKD9cvDXGpvPvdSkHJ+4rnpKansBctcfSt6j2sK
zlpWFctXYQ/ZRN3GRjs4FUARsylePz+L7qR11FASMvehOq+sF1xT3+wNcJpdZP83w9oZWxmUqyLytKpaXzOFHSjrQ1uY
qtKNMThii6mnwX5wFdtY6Cp9nU9GwuUUPhp8jdUcFlzKsYeyVfKUrAOi0ZhKfWN7bGelrnofpIppHEyVthRUQnbJHGqps
WEB7e17YtkrioLRIXan4Bge+Rpe6tnVbhjWbMyJ7A8VsAyzbcvVYPtV5guAHbG9uXg+oxqQ/QP72pyV8rUI67j1sJzrqXC
jLZJa+/NLALBvixZ7Q2tq8Exm63v4UKan5nvYzBbCc8TiQ7lweoV+xFlDiRkms/L9uoFvZKmxWp222myTWQzfXhNKKRf
DRrOck9ktG/mF+VUV7Q6InmZjgqBej9Bhx8OD1X5KxzKPUtIMUZqwbdm6Kq7KYgnXRE9nz599En9uf9nmN4WyXyflLc/d
fidITxPr82Qm9Xuq/boqiszEdtyMJyxBVfKohqVqOCTpT4XmyJ2UyS43+gTXV4JmM1r6TYM3syZNIzLs1LkbMcGgrL9U30
O3WY5A0bzL9FHa3xaJ0s01mMdtnsUaznJPzJ7E7Nyx79NHT0pJcowskKXF2NM1hM0zHnZlG1weyI67202iy
20qQc2T7qHHRlx+oFN6vsvzgpSeSB3YHNx1YYNw+oFNQkEKW4SSEwxqpfHQ3xYjpoFFuNhgobNsqkJ1r3P65uistPP9m6
ZEmXGVVnBczFl+6HmzbpuyuoKZq+gPO2YvbwzngcAqdHGfbyxye3e40pgH0slqTa2PirrVplKE6tGV61gkLXvBVXpj/
fyrJxz5ZhgtrtYbWt5PqVkvAVmavwVWL8laXj28ym5Xj/VKVDQBczsk5eW2yzTa5xTiW4IBj2YwZJ7MaWla5zpp1JrbZu
4Uss0obv99th7V7DU1luty9ynto26EZ/7tzcVcVYLWAA6G2xK/drAc21I9A8SRA0PLMkrZTWkxzeAtAlJxJd
cf32Q224GLxddmm9xiHLA8NmPGyaxmoxZrNss5vd00ZvktP7yHt10vW8GxChHfsoHxJDAxIR/aQrs1YQNTapR90t27E1p
JAZZLAG44nIGYfMaTcaJfPpuHOCU1ioj+2rCMf2MCXX+CjJdtCDLLeFz+LatdOs+ARVzTKvcEc2FQ2Bm17WYapjKThjJW
shzYUD10HpjE9OQoA7X6N6giBN+Pwb6W1t7+u82uZcVMEdt/hzVoMrJtrQtNlPJRfzBjNtqlRpmJxvI6RBLIeiVZVj
tFlyFG2av1RPKK3MTPf7bmy0h21C4zUUXnd2c01oe80D30UAad97VFNJKvt6sbsIB7ZTvKWRdbNcmSvb5bNsW8Ab2/A4+K
ZdMAC+89m331kn13EYpc+foce5YUyx2ws80P1z8btamMhew8Ncy2129a/s7OHDMlyzbi7W60++tYyGwpDQdH+GPwoy/NH
xGf44yPDHkxn+a13dZk+KXB/MIotwNnUyQy/WdaGdum0IGFmbyhp6+hAeu1/QU6SjO/Erstmjrw5RPMcJt/GRHR1Gt/FH
OOSIetd6vVqLgrCxl1sDW312AGTVNAs3f8pXm7++wg5eqFgbITzcd92jge3PcK/2p2q+XhVZ+wY4tran0c5qJG87d3F+r
E22PrfshzV9fNcGhdgaOJHQ7IjPIXskkVpUztdv24OeQod672C9hxx+BV9tWUVtNRQdvdYO/Ox+huzgd1SsdccqAiFs/d
mOZXJoS1oLD2SV6WSmUa0XTmKZ7XCWbV63g9f7D4Sh5XuWorN7YLdytPX5oOLRqB3EC1WuVrDrISQsE1nn80Vt39TeqA3
x+CPloWjgmAgj3PtpOXOnKVdc5TkD2yt5Pm3VPXomn7P1nR62Ca20vRhr5zSe6jDG81YfH0VXe+y99eLvimZXV3CVInux
3cIM2vZvk4NueD/tymzT8k+WYRLlBJ+0bPN64y1MVQ4EouWl1qW+bvbyb+7JT3RyTbMbTHar0xy9DfFqG5kQ61oFDD9M6
cvd90Vhmqbtqy2S9f6iQn41fb5s9825VY/k3AGzydAAvmdr/5g7+04URJ96ofnsHJ1+2mgLNT4uV/e52XcAYTWtVU/EXer4H
aFnNdtDt1A2cUpgl76XaGvcM1DCueZwFZZfL2fPbhd+ET2+KnhDcwH6S6k6mLWnt218fJpf2pEdstTjZvhb+3770AhrD
rTQD7BV6YtSH+CILRjH00zB43FlbTNA+4FqsHYovXB7TFHAD3NzweH9P3d3qNPfFwkPe5K09g/LlNu1YGQa+Zg8DX3GHw
a/aqCuibMLRpuBaOwzqqBBfDkVUEB6anMhwCngpXSPgqxSESUjHmKh+qEpsUVBmiw1ID6v0juWmEB2qL1QOyxuE12IOA
dazT1Yw9n1FzQ3aibYjwEwfc/R2n7KNynjUXxi4XspQ/VqfLh1ulSfmK8efHihVdI3E7VciGQCIuZwVWlj+bpvFAC1G2e
PzDLcPbY9BOSuW7BwWBDUbhYdCdnOfu/hoVejnFSqYarXZF2nuK6A4eWnt+W8LKr5nRlyYvzeZRicrLgD38Xs83ynjK8
inEih1dFNGxH60sUAu69vjLbPu2OglbZ6RQcbTjw/9TiyvYvGbpXhCsDEWThBGFmVvrMx1Fm/8pcz905Q2ahhW/YLHlOm5
RokyH4YmbwiWekqs3UuPr23S3mhqlW351Y11x2SQMvvQ1JzKiz2OaPH6ELWYQyAVTY+5DY/c1M1hAKxv0+OWDWEFHRqkN
+3N4KntDsHZYQ0bP28tjpeVvHh7pLh4D8Rqx7Pn+nxH/UKYJ1YalUSfmK8efHihVdI3E7VciGQCIuZwVWlj+bpvFAC1G2e
hKVDLWotLEWOkp5hJepDFUiCqs3rJwovjDgqYOTCPKMIESH+My1IYU9+sgt5cf+RLF7ZFs6Rtp910t/8ttYaLNRgm85
lpT4OZahXyVilSuPU6I+vGHp00lakEz1ntP118p2oywLWzJawgLGrvy/sCnF/B5aifKw4Px8fNaFbpm23pqtPqWU2I2uV
jLNRaVSlVIO/hGooWiSdDqdzKruZgaGtS8WKFCTPFoAkrxbyWms6oadvZjSp7ttQpJKxHpXGFURuE+oOFRzrFx2HB9ng
hxedOMBIO9Cw5kQglurvI3KKpUk2cnwkAu81RXK3wnXcNpZ8grdOavv53HlKpt7Xw0iYULnoA4y1rLe2M1LQpVs5E9T
ESU49KJ4pUImvQegfPV7i18TW1Eq2UJHSf6N4M+4t5ECvvdv7hcuUFBK1wNb76f1G3Ex6HxmEATrLeT7mc7aE9BD1k
5Kq9Hx1PVQ+OVZYY4oEU/5qcKo1V01AH51rtR36rFJYGX1oHGyt2kMDimbuWtxX+4tTfzTGDuS2OcrWWRQsEVVuk9W4Lfp
mpc6VbpPtn/ZUNWV1FXpIHuCHU/4d9679bdPhFvng+OoTHS3DT5qum1d+Uw0kSGmb3gm2LXUbaU7iWtRoky07CjT45Dsg
myC4xJ8Wh+pv9/xXGekfKchTI7yU2PQjR9G8IEkYspQ/3vNm1QyodBuoLsgguQAXadOax/FII7ZXKUWLDI5cKdpmhvjJRJh
fuB1eQVS7cdV1Bdr1wp7pcuMMZNdSeHZVDMrwlpeW03CmTz1Tjgc1lM1VOQZXkOmuyjF0Bdn1qhyq1Y5cPIGVY7DeMr
kwJ0yOv09ZXRkN2fK5MLd4Aoy/TTI91NHT2Hk6KgyduWqHYPL0SgehNQOZXKUKDI5WgV1c1QhZXJUIVWYVaHB5ahCgyuo
gK4qNHQ5qpByuarQ4HJUIdxnHW5g1M1RQsrkwtTgcmBqcDn8weBy1LXB5cUCrkwNbgcmLbPp0ixpiNDyuSKU01DEFQVA
y3K5KgeyuRALDK56tDgCrLqNCxHRx0aOToqpVyuOjS4HUUoJhWPNWYDyXGSDK7T1vwTMWmBqcyM/LxuLvL5bXFZfcRhZw
mY0Dz/fweAAAD//wMAUESDBBQABgATAAAAQIAKIRf4zEAANQCAAAASAAAAGwvY29ubmVjdGllvbnMueG1sbjJNbnxpBDbIb
v1fofrLmXXRK1agNdHCKRkLlCdUnTA5dhxiXt5mPlmaXh38dL2yRFva09tt/Xj3d8++QdHJGSjUGK4aAUgEFHY0MjxcP6
7sNnASmrYJSLAAU4YRK3k/fvxjgGgDpzWwKeEZU+5zbn6JIEo9epUFsMfDLlPjXmUNqitQSKpP2iNm74qosPxVe2Samb
8aBNWxEQDT6zjWnN1shH173PyM+6CVbzfzqDuPIdNdYK5ygrq2JHGthly+JSNy19gao5c0RHCY6RDgVnjZWRACyIOiO
3U2SMLShWhqzj8uegXkU8vpjwIId4Ts33z/i+xawFbPqQ00Xc7+UJ3E3tR5oCY07MMqX4BSJhwyUKbrsLZNM2W1YFNy2/Yrs
pJ+ZgtiuCV7ZsrCgerUGSVb28n64G16NFYniQymigRtRzSdZ16vFanV7SvklC4uwi3JNHXKP7T3AjH252MiLDZno6A1H
+YjxVM1/wCz6VmW7ta6XGo5gtcN8gmV7vr68OhdVNVvZqYV+1wRVNc1/EkUsvI73/Af9h+uf1/UZ71aUk3F0gmUCCYO5j
1r19+ezUPw1J+vcjCFnzpR12RcXr6D+CdLgKQAA//8DAFBLAWQUAAAYACAAAACEALmnZ2SkBAAABAAAJwAAAHsL3ByaW

</soap:Envelope>

5 Security

5.1 Security Considerations for Implementers

There are security risks associated with exposing session identifiers (2). If the session identifier is exposed, it is possible for an attacker to read information from, or modify data in, a session (2) on the protocol server. An implementer of this protocol is to consider keeping session identifiers protected. There could be some cases where it is desirable to expose the session identifier, but implementers are to use caution in how they expose session identifiers and consider the security risks.

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

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

```
<?xml version="1.0" encoding="UTF-8"?>
<wsdl:definitions xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
xmlns:http="http://schemas.xmlsoap.org/wsdl/http/"
xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:tns="http://schemas.microsoft.com/office/excel/server/webservices"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
targetNamespace="http://schemas.microsoft.com/office/excel/server/webservices"
xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:types>
    <xs:schema attributeFormDefault="unqualified" elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/office/excel/server/webservices">
      <xs:element name="GetApiVersion">
        <xs:complexType/>
      </xs:element>
      <xs:element name="GetApiVersionResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="GetApiVersionResult"
type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:complexType name="ArrayOfStatus">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="unbounded" name="Status" nillable="true"
type="tns:Status"/>
        </xs:sequence>
      </xs:complexType>
      <xs:complexType name="Status">
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
          <xs:element minOccurs="0" maxOccurs="1" name="Severity" type="xs:string"/>
          <xs:element minOccurs="0" maxOccurs="1" name="Message" type="xs:string"/>
        </xs:sequence>
      </xs:complexType>
      <xs:element name="NewWorkbook">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="path" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="uiCultureName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="dataCultureName" type="xs:string"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="NewWorkbookResponse">
        <xs:complexType>
          <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="NewWorkbookResult"
type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
      <xs:element name="OpenWorkbookForEditing">
```

```

    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="workbookPath" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="uiCultureName" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="dataCultureName" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="OpenWorkbookForEditingResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="OpenWorkbookForEditingResult"
type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="OpenWorkbook">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="workbookPath" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="uiCultureName" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="dataCultureName" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="OpenWorkbookResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="OpenWorkbookResult"
type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="CloseWorkbook">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="CloseWorkbookResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="Refresh">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="connectionName" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="RefreshResponse">
    <xs:complexType>

```

```

        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SaveWorkbook">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SaveWorkbookResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SaveWorkbookCopy">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="workbookPath" type="xs:string"/>
            <xs:element minOccurs="1" maxOccurs="1" name="workbookType"
type="tns:WorkbookType"/>
            <xs:element minOccurs="1" maxOccurs="1" name="saveOptions"
type="tns:SaveOptions"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:simpleType name="WorkbookType">
    <xs:restriction base="xs:string">
        <xs:enumeration value="FullWorkbook"/>
        <xs:enumeration value="FullSnapshot"/>
        <xs:enumeration value="PublishedItemsSnapshot"/>
    </xs:restriction>
</xs:simpleType>
<xs:simpleType name="SaveOptions">
    <xs:list>
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:enumeration value="None"/>
                <xs:enumeration value="AllowOverwrite"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:list>
</xs:simpleType>
<xs:element name="SaveWorkbookCopyResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="Calculate">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>

```



```

        <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="rangeCoordinates"
type="tns:RangeCoordinates"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:complexType name="RangeCoordinates">
    <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="Row" type="xs:int"/>
        <xs:element minOccurs="1" maxOccurs="1" name="Column" type="xs:int"/>
        <xs:element minOccurs="1" maxOccurs="1" name="Height" type="xs:int"/>
        <xs:element minOccurs="1" maxOccurs="1" name="Width" type="xs:int"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="CalculateResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="CalculateA1">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="rangeName" type="xs:string"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="CalculateA1Response">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="GetRange">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="rangeCoordinates"
type="tns:RangeCoordinates"/>
            <xs:element minOccurs="1" maxOccurs="1" name="formatted" type="xs:boolean"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="GetRangeResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="GetRangeResult"
type="tns:ArrayOfAnyType"/>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:complexType name="ArrayOfAnyType">
    <xs:sequence>

```

```

        <xs:element minOccurs="0" maxOccurs="unbounded" name="anyType" nillable="true"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="TypeExporter">
    <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="1" name="CellError" type="tns:CellError"/>
    </xs:sequence>
</xs:complexType>
<xs:simpleType name="CellError">
    <xs:restriction base="xs:string">
        <xs:enumeration value="Div0"/>
        <xs:enumeration value="NA"/>
        <xs:enumeration value="Name"/>
        <xs:enumeration value="Null"/>
        <xs:enumeration value="Num"/>
        <xs:enumeration value="Ref"/>
        <xs:enumeration value="Value"/>
    </xs:restriction>
</xs:simpleType>
<xs:element name="GetRangeA1">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="rangeName" type="xs:string"/>
            <xs:element minOccurs="1" maxOccurs="1" name="formatted" type="xs:boolean"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="GetRangeA1Response">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="GetRangeA1Result"
type="tns:ArrayOfAnyType"/>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="GetCell">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
            <xs:element minOccurs="1" maxOccurs="1" name="row" type="xs:int"/>
            <xs:element minOccurs="1" maxOccurs="1" name="column" type="xs:int"/>
            <xs:element minOccurs="1" maxOccurs="1" name="formatted" type="xs:boolean"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="GetCellResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="GetCellResult"/>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="GetCellA1">
    <xs:complexType>

```

```

        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
          <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
          <xs:element minOccurs="0" maxOccurs="1" name="rangeName" type="xs:string"/>
          <xs:element minOccurs="1" maxOccurs="1" name="formatted" type="xs:boolean"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="GetCellA1Response">
      <xs:complexType>
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="GetCellA1Result"/>
          <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="GetPublishedItemNames">
      <xs:complexType>
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:complexType name="WorkbookItem">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
        <xs:element minOccurs="1" maxOccurs="1" name="Type" type="tns:ItemType"/>
      </xs:sequence>
    </xs:complexType>
    <xs:simpleType name="ItemType">
      <xs:restriction base="xs:string">
        <xs:enumeration value="Chart"/>
        <xs:enumeration value="PivotTable"/>
        <xs:enumeration value="Table"/>
        <xs:enumeration value="NamedRange"/>
      </xs:restriction>
    </xs:simpleType>
    <xs:complexType name="ArrayOfWorkbookItem">
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="WorkbookItem" nillable="true"
type="tns:WorkbookItem"/>
      </xs:sequence>
    </xs:complexType>
    <xs:element name="GetPublishedItemNamesResponse">
      <xs:complexType>
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="GetPublishedItemNamesResult"
type="tns:ArrayOfWorkbookItem"/>
          <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>
    <xs:element name="GetSheetNames">
      <xs:complexType>
        <xs:sequence>
          <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
        </xs:sequence>
      </xs:complexType>
    </xs:element>

```

```

<xs:complexType name="SheetInfo">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Type" type="tns:SheetType"/>
    <xs:element minOccurs="1" maxOccurs="1" name="Visibility"
type="tns:SheetVisibility"/>
  </xs:sequence>
</xs:complexType>
<xs:simpleType name="SheetType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Worksheet"/>
    <xs:enumeration value="Chartsheet"/>
  </xs:restriction>
</xs:simpleType>
<xs:simpleType name="SheetVisibility">
  <xs:restriction base="xs:string">
    <xs:enumeration value="Visible"/>
    <xs:enumeration value="Hidden"/>
    <xs:enumeration value="VeryHidden"/>
  </xs:restriction>
</xs:simpleType>
<xs:complexType name="ArrayOfSheetInfo">
  <xs:sequence>
    <xs:element minOccurs="0" maxOccurs="unbounded" name="SheetInfo" nillable="true"
type="tns:SheetInfo"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="GetSheetNamesResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="GetSheetNamesResult"
type="tns:ArrayOfSheetInfo"/>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SetRange">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="rangeCoordinates"
type="tns:RangeCoordinates"/>
      <xs:element minOccurs="0" maxOccurs="1" name="rangeValues"
type="tns:ArrayOfAnyType"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SetRangeResponse">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
<xs:element name="SetRangeA1">
  <xs:complexType>
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>

```

```

        <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="rangeName" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="rangeValues"
type="tns:ArrayOfAnyType"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="SetRangeA1Response">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetCell">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
            <xs:element minOccurs="1" maxOccurs="1" name="row" type="xs:int"/>
            <xs:element minOccurs="1" maxOccurs="1" name="column" type="xs:int"/>
            <xs:element minOccurs="0" maxOccurs="1" name="cellValue"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetCellResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetCellA1">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="sheetName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="rangeName" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="cellValue"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetCellA1Response">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="CalculateWorkbook">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
            <xs:element minOccurs="1" maxOccurs="1" name="calculateType"
type="tns:CalculateType"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:simpleType name="CalculateType">

```

```

    <xs:restriction base="xs:string">
      <xs:enumeration value="Recalculate"/>
      <xs:enumeration value="CalculateFull"/>
    </xs:restriction>
  </xs:simpleType>
  <xs:element name="CalculateWorkbookResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetSessionInformation">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetSessionInformationResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="serverVersion" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="uiCultureNameUsed"
type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="dataCultureNameUsed"
type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="CancelRequest">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="CancelRequestResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetWorkbook">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
        <xs:element minOccurs="1" maxOccurs="1" name="workbookType"
type="tns:WorkbookType"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="GetWorkbookResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="GetWorkbookResult"
type="xs:base64Binary"/>

```

```

        <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="SetParameters">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
            <xs:element minOccurs="0" maxOccurs="1" name="parameters"
type="tns:ArrayOfParameterInfo"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:complexType name="ArrayOfParameterInfo">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="ParameterInfo"
nillable="true" type="tns:ParameterInfo"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ParameterInfo">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="Name" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="Values" type="tns:ArrayOfString"/>
    </xs:sequence>
</xs:complexType>
<xs:complexType name="ArrayOfString">
    <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="unbounded" name="string" nillable="true"
type="xs:string"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="SetParametersResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:element name="SetCalculationOptions">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
            <xs:element minOccurs="1" maxOccurs="1" name="calcType"
type="tns:WorkbookCalculation"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>
<xs:simpleType name="WorkbookCalculation">
    <xs:restriction base="xs:string">
        <xs:enumeration value="Manual"/>
        <xs:enumeration value="Automatic"/>
        <xs:enumeration value="Semiautomatic"/>
    </xs:restriction>
</xs:simpleType>
<xs:element name="SetCalculationOptionsResponse">
    <xs:complexType>
        <xs:sequence>
            <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
        </xs:sequence>
    </xs:complexType>
</xs:element>

```

```

    </xs:complexType>
  </xs:element>
  <xs:element name="GetChartImageUrl">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="size" type="tns:Size"/>
        <xs:element minOccurs="0" maxOccurs="1" name="chartName" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="Size">
    <xs:sequence>
      <xs:element minOccurs="1" maxOccurs="1" name="Width" type="xs:int"/>
      <xs:element minOccurs="1" maxOccurs="1" name="Height" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="GetChartImageUrlResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="GetChartImageUrlResult"
type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="EnsureWorkbookModel">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:complexType name="WorkbookModelInfo">
    <xs:sequence>
      <xs:element minOccurs="0" maxOccurs="1" name="Server" type="xs:string"/>
      <xs:element minOccurs="0" maxOccurs="1" name="Database" type="xs:string"/>
      <xs:element minOccurs="1" maxOccurs="1" name="SessionInteractivityTimeout"
type="xs:int"/>
      <xs:element minOccurs="1" maxOccurs="1" name="Version" type="xs:int"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="EnsureWorkbookModelResponse">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="EnsureWorkbookModelResult"
type="tns:WorkbookModelInfo"/>
        <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>
  <xs:element name="OpenWorkbookEx">
    <xs:complexType>
      <xs:sequence>
        <xs:element minOccurs="0" maxOccurs="1" name="workbookPath" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="uiCultureName" type="xs:string"/>
        <xs:element minOccurs="0" maxOccurs="1" name="dataCultureName" type="xs:string"/>
        <xs:element minOccurs="1" maxOccurs="1" name="exclusive" type="xs:boolean"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

```



```

        </xs:complexType>
    </xs:element>
    <xs:element name="OpenWorkbookExResponse">
        <xs:complexType>
            <xs:sequence>
                <xs:element minOccurs="0" maxOccurs="1" name="OpenWorkbookExResult"
type="xs:string"/>
                <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="GetWorkbookConnections">
        <xs:complexType>
            <xs:sequence>
                <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="GetWorkbookConnectionsResponse">
        <xs:complexType>
            <xs:sequence>
                <xs:element minOccurs="0" maxOccurs="1" name="GetWorkbookConnectionsResult"
type="tns:ArrayOfString"/>
                <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="RefreshEx">
        <xs:complexType>
            <xs:sequence>
                <xs:element minOccurs="0" maxOccurs="1" name="sessionId" type="xs:string"/>
                <xs:element minOccurs="0" maxOccurs="1" name="connectionName" type="xs:string"/>
                <xs:element minOccurs="0" maxOccurs="1" name="targetApplicationId"
type="xs:string"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:element name="RefreshExResponse">
        <xs:complexType>
            <xs:sequence>
                <xs:element minOccurs="0" maxOccurs="1" name="status" type="tns:ArrayOfStatus"/>
            </xs:sequence>
        </xs:complexType>
    </xs:element>
    <xs:simpleType name="RangeReference">
        <xs:restriction base="xs:string"/>
    </xs:simpleType>
    <xs:simpleType name="SheetName">
        <xs:restriction base="xs:string"/>
    </xs:simpleType>
</xs:schema>
</wsdl:types>
<wsdl:portType name="ExcelServiceSoap">
    <wsdl:operation name="GetApiVersion">
        <wsdl:input message="tns:GetApiVersionSoapIn"/>
        <wsdl:output message="tns:GetApiVersionSoapOut"/>
    </wsdl:operation>
    <wsdl:operation name="NewWorkbook">
        <wsdl:input message="tns:NewWorkbookSoapIn"/>

```

```

    <wsdl:output message="tns:NewWorkbookSoapOut"/>
</wsdl:operation>
<wsdl:operation name="OpenWorkbookForEditing">
    <wsdl:input message="tns:OpenWorkbookForEditingSoapIn"/>
    <wsdl:output message="tns:OpenWorkbookForEditingSoapOut"/>
</wsdl:operation>
<wsdl:operation name="OpenWorkbook">
    <wsdl:input message="tns:OpenWorkbookSoapIn"/>
    <wsdl:output message="tns:OpenWorkbookSoapOut"/>
</wsdl:operation>
<wsdl:operation name="CloseWorkbook">
    <wsdl:input message="tns:CloseWorkbookSoapIn"/>
    <wsdl:output message="tns:CloseWorkbookSoapOut"/>
</wsdl:operation>
<wsdl:operation name="Refresh">
    <wsdl:input message="tns:RefreshSoapIn"/>
    <wsdl:output message="tns:RefreshSoapOut"/>
</wsdl:operation>
<wsdl:operation name="SaveWorkbook">
    <wsdl:input message="tns:SaveWorkbookSoapIn"/>
    <wsdl:output message="tns:SaveWorkbookSoapOut"/>
</wsdl:operation>
<wsdl:operation name="SaveWorkbookCopy">
    <wsdl:input message="tns:SaveWorkbookCopySoapIn"/>
    <wsdl:output message="tns:SaveWorkbookCopySoapOut"/>
</wsdl:operation>
<wsdl:operation name="Calculate">
    <wsdl:input message="tns:CalculateSoapIn"/>
    <wsdl:output message="tns:CalculateSoapOut"/>
</wsdl:operation>
<wsdl:operation name="CalculateA1">
    <wsdl:input message="tns:CalculateA1SoapIn"/>
    <wsdl:output message="tns:CalculateA1SoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetRange">
    <wsdl:input message="tns:GetRangeSoapIn"/>
    <wsdl:output message="tns:GetRangeSoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetRangeA1">
    <wsdl:input message="tns:GetRangeA1SoapIn"/>
    <wsdl:output message="tns:GetRangeA1SoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetCell">
    <wsdl:input message="tns:GetCellSoapIn"/>
    <wsdl:output message="tns:GetCellSoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetCellA1">
    <wsdl:input message="tns:GetCellA1SoapIn"/>
    <wsdl:output message="tns:GetCellA1SoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetPublishedItemNames">
    <wsdl:input message="tns:GetPublishedItemNamesSoapIn"/>
    <wsdl:output message="tns:GetPublishedItemNamesSoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetSheetNames">
    <wsdl:input message="tns:GetSheetNamesSoapIn"/>
    <wsdl:output message="tns:GetSheetNamesSoapOut"/>
</wsdl:operation>
<wsdl:operation name="SetRange">

```

```

    <wsdl:input message="tns:SetRangeSoapIn"/>
    <wsdl:output message="tns:SetRangeSoapOut"/>
</wsdl:operation>
<wsdl:operation name="SetRangeA1">
    <wsdl:input message="tns:SetRangeA1SoapIn"/>
    <wsdl:output message="tns:SetRangeA1SoapOut"/>
</wsdl:operation>
<wsdl:operation name="SetCell">
    <wsdl:input message="tns:SetCellSoapIn"/>
    <wsdl:output message="tns:SetCellSoapOut"/>
</wsdl:operation>
<wsdl:operation name="SetCellA1">
    <wsdl:input message="tns:SetCellA1SoapIn"/>
    <wsdl:output message="tns:SetCellA1SoapOut"/>
</wsdl:operation>
<wsdl:operation name="CalculateWorkbook">
    <wsdl:input message="tns:CalculateWorkbookSoapIn"/>
    <wsdl:output message="tns:CalculateWorkbookSoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetSessionInformation">
    <wsdl:input message="tns:GetSessionInformationSoapIn"/>
    <wsdl:output message="tns:GetSessionInformationSoapOut"/>
</wsdl:operation>
<wsdl:operation name="CancelRequest">
    <wsdl:input message="tns:CancelRequestSoapIn"/>
    <wsdl:output message="tns:CancelRequestSoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetWorkbook">
    <wsdl:input message="tns:GetWorkbookSoapIn"/>
    <wsdl:output message="tns:GetWorkbookSoapOut"/>
</wsdl:operation>
<wsdl:operation name="SetParameters">
    <wsdl:input message="tns:SetParametersSoapIn"/>
    <wsdl:output message="tns:SetParametersSoapOut"/>
</wsdl:operation>
<wsdl:operation name="SetCalculationOptions">
    <wsdl:input message="tns:SetCalculationOptionsSoapIn"/>
    <wsdl:output message="tns:SetCalculationOptionsSoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetChartImageUrl">
    <wsdl:input message="tns:GetChartImageUrlSoapIn"/>
    <wsdl:output message="tns:GetChartImageUrlSoapOut"/>
</wsdl:operation>
<wsdl:operation name="EnsureWorkbookModel">
    <wsdl:input message="tns:EnsureWorkbookModelSoapIn"/>
    <wsdl:output message="tns:EnsureWorkbookModelSoapOut"/>
</wsdl:operation>
<wsdl:operation name="OpenWorkbookEx">
    <wsdl:input message="tns:OpenWorkbookExSoapIn"/>
    <wsdl:output message="tns:OpenWorkbookExSoapOut"/>
</wsdl:operation>
<wsdl:operation name="GetWorkbookConnections">
    <wsdl:input message="tns:GetWorkbookConnectionsSoapIn"/>
    <wsdl:output message="tns:GetWorkbookConnectionsSoapOut"/>
</wsdl:operation>
<wsdl:operation name="RefreshEx">
    <wsdl:input message="tns:RefreshExSoapIn"/>
    <wsdl:output message="tns:RefreshExSoapOut"/>
</wsdl:operation>

```

```

</wsdl:portType>
<wsdl:binding name="ExcelServiceSoap" type="tns:ExcelServiceSoap">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http"/>
  <wsdl:operation name="GetApiVersion">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetApiVersion"
style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="NewWorkbook">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/NewWorkbook"
style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="OpenWorkbookForEditing">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/OpenWorkbookForEditi
ng" style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="OpenWorkbook">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/OpenWorkbook"
style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="CloseWorkbook">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/CloseWorkbook"
style="document"/>
    <wsdl:input>
      <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
      <soap:body use="literal"/>
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="Refresh">

```

```

    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/Refresh"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SaveWorkbook">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SaveWorkbook"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SaveWorkbookCopy">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SaveWorkbookCopy"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="Calculate">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/Calculate"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CalculateA1">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/CalculateA1"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetRange">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetRange"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>

```

```

        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetRangeA1">
        <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetRangeA1"
style="document"/>
        <wsdl:input>
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetCell">
        <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetCell"
style="document"/>
        <wsdl:input>
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetCellA1">
        <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetCellA1"
style="document"/>
        <wsdl:input>
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetPublishedItemNames">
        <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetPublishedItemName
s" style="document"/>
        <wsdl:input>
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetSheetNames">
        <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetSheetNames"
style="document"/>
        <wsdl:input>
            <soap:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>

```

```

    <wsdl:operation name="SetRange">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetRange"
style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SetRangeA1">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetRangeA1"
style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SetCell">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetCell"
style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SetCellA1">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetCellA1"
style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="CalculateWorkbook">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/CalculateWorkbook"
style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetSessionInformation">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetSessionInformatio
n" style="document"/>
      <wsdl:input>

```

```

        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CancelRequest">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/CancelRequest"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetWorkbook">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetWorkbook"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetParameters">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetParameters"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetCalculationOptions">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetCalculationOptions"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetChartImageUrl">
    <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetChartImageUrl"
style="document"/>
    <wsdl:input>
        <soap:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap:body use="literal"/>
    </wsdl:output>

```



```

    </wsdl:operation>
    <wsdl:operation name="EnsureWorkbookModel">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/EnsureWorkbookModel"
style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="OpenWorkbookEx">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/OpenWorkbookEx"
style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetWorkbookConnections">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetWorkbookConnections"
style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="RefreshEx">
      <soap:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/RefreshEx"
style="document"/>
      <wsdl:input>
        <soap:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
  </wsdl:binding>
  <wsdl:binding name="ExcelServiceSoap12" type="tns:ExcelServiceSoap">
    <soap12:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="GetApiVersion">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetApiVersion"
style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="NewWorkbook">

```

```

    <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/NewWorkbook"
style="document"/>
    <wsdl:input>
        <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="OpenWorkbookForEditing">
    <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/OpenWorkbookForEditi
ng" style="document"/>
    <wsdl:input>
        <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="OpenWorkbook">
    <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/OpenWorkbook"
style="document"/>
    <wsdl:input>
        <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CloseWorkbook">
    <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/CloseWorkbook"
style="document"/>
    <wsdl:input>
        <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="Refresh">
    <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/Refresh"
style="document"/>
    <wsdl:input>
        <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SaveWorkbook">
    <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SaveWorkbook"
style="document"/>
    <wsdl:input>
        <soap12:body use="literal"/>

```

```

        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SaveWorkbookCopy">
        <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SaveWorkbookCopy"
style="document"/>
        <wsdl:input>
            <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="Calculate">
        <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/Calculate"
style="document"/>
        <wsdl:input>
            <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="CalculateA1">
        <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/CalculateA1"
style="document"/>
        <wsdl:input>
            <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetRange">
        <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetRange"
style="document"/>
        <wsdl:input>
            <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetRangeA1">
        <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetRangeA1"
style="document"/>
        <wsdl:input>
            <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>

```

```

    <wsdl:operation name="GetCell">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetCell"
style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetCellA1">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetCellA1"
style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetPublishedItemNames">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetPublishedItemName
s" style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetSheetNames">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetSheetNames"
style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SetRange">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetRange"
style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SetRangeA1">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetRangeA1"
style="document"/>
      <wsdl:input>

```

```

        <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetCell">
    <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetCell"
style="document"/>
    <wsdl:input>
        <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetCellA1">
    <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetCellA1"
style="document"/>
    <wsdl:input>
        <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CalculateWorkbook">
    <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/CalculateWorkbook"
style="document"/>
    <wsdl:input>
        <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetSessionInformation">
    <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetSessionInformation"
style="document"/>
    <wsdl:input>
        <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal"/>
    </wsdl:output>
</wsdl:operation>
<wsdl:operation name="CancelRequest">
    <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/CancelRequest"
style="document"/>
    <wsdl:input>
        <soap12:body use="literal"/>
    </wsdl:input>
    <wsdl:output>
        <soap12:body use="literal"/>
    </wsdl:output>

```

```

    </wsdl:operation>
    <wsdl:operation name="GetWorkbook">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetWorkbook"
style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SetParameters">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetParameters"
style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="SetCalculationOptions">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/SetCalculationOption
s" style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetChartImageUrl">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetChartImageUrl"
style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="EnsureWorkbookModel">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/EnsureWorkbookModel"
style="document"/>
      <wsdl:input>
        <soap12:body use="literal"/>
      </wsdl:input>
      <wsdl:output>
        <soap12:body use="literal"/>
      </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="OpenWorkbookEx">
      <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/OpenWorkbookEx"
style="document"/>

```

```

        <wsdl:input>
            <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="GetWorkbookConnections">
        <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/GetWorkbookConnectio
ns" style="document"/>
        <wsdl:input>
            <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
    <wsdl:operation name="RefreshEx">
        <soap12:operation
soapAction="http://schemas.microsoft.com/office/excel/server/webservices/RefreshEx"
style="document"/>
        <wsdl:input>
            <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
</wsdl:binding>
<wsdl:message name="CalculateA1SoapIn">
    <wsdl:part name="parameters" element="tns:CalculateA1"/>
</wsdl:message>
<wsdl:message name="CalculateA1SoapOut">
    <wsdl:part name="parameters" element="tns:CalculateA1Response"/>
</wsdl:message>
<wsdl:message name="CalculateSoapIn">
    <wsdl:part name="parameters" element="tns:Calculate"/>
</wsdl:message>
<wsdl:message name="CalculateSoapOut">
    <wsdl:part name="parameters" element="tns:CalculateResponse"/>
</wsdl:message>
<wsdl:message name="CalculateWorkbookSoapIn">
    <wsdl:part name="parameters" element="tns:CalculateWorkbook"/>
</wsdl:message>
<wsdl:message name="CalculateWorkbookSoapOut">
    <wsdl:part name="parameters" element="tns:CalculateWorkbookResponse"/>
</wsdl:message>
<wsdl:message name="CancelRequestSoapIn">
    <wsdl:part name="parameters" element="tns:CancelRequest"/>
</wsdl:message>
<wsdl:message name="CancelRequestSoapOut">
    <wsdl:part name="parameters" element="tns:CancelRequestResponse"/>
</wsdl:message>
<wsdl:message name="CloseWorkbookSoapIn">
    <wsdl:part name="parameters" element="tns:CloseWorkbook"/>
</wsdl:message>
<wsdl:message name="CloseWorkbookSoapOut">
    <wsdl:part name="parameters" element="tns:CloseWorkbookResponse"/>

```

```

</wsdl:message>
<wsdl:message name="EnsureWorkbookModelSoapIn">
  <wsdl:part name="parameters" element="tns:EnsureWorkbookModel"/>
</wsdl:message>
<wsdl:message name="EnsureWorkbookModelSoapOut">
  <wsdl:part name="parameters" element="tns:EnsureWorkbookModelResponse"/>
</wsdl:message>
<wsdl:message name="GetApiVersionSoapIn">
  <wsdl:part name="parameters" element="tns:GetApiVersion"/>
</wsdl:message>
<wsdl:message name="GetApiVersionSoapOut">
  <wsdl:part name="parameters" element="tns:GetApiVersionResponse"/>
</wsdl:message>
<wsdl:message name="GetCellA1SoapIn">
  <wsdl:part name="parameters" element="tns:GetCellA1"/>
</wsdl:message>
<wsdl:message name="GetCellA1SoapOut">
  <wsdl:part name="parameters" element="tns:GetCellA1Response"/>
</wsdl:message>
<wsdl:message name="GetCellSoapIn">
  <wsdl:part name="parameters" element="tns:GetCell"/>
</wsdl:message>
<wsdl:message name="GetCellSoapOut">
  <wsdl:part name="parameters" element="tns:GetCellResponse"/>
</wsdl:message>
<wsdl:message name="GetChartImageUrlSoapIn">
  <wsdl:part name="parameters" element="tns:GetChartImageUrl"/>
</wsdl:message>
<wsdl:message name="GetChartImageUrlSoapOut">
  <wsdl:part name="parameters" element="tns:GetChartImageUrlResponse"/>
</wsdl:message>
<wsdl:message name="GetPublishedItemNamesSoapIn">
  <wsdl:part name="parameters" element="tns:GetPublishedItemNames"/>
</wsdl:message>
<wsdl:message name="GetPublishedItemNamesSoapOut">
  <wsdl:part name="parameters" element="tns:GetPublishedItemNamesResponse"/>
</wsdl:message>
<wsdl:message name="GetRangeA1SoapIn">
  <wsdl:part name="parameters" element="tns:GetRangeA1"/>
</wsdl:message>
<wsdl:message name="GetRangeA1SoapOut">
  <wsdl:part name="parameters" element="tns:GetRangeA1Response"/>
</wsdl:message>
<wsdl:message name="GetRangeSoapIn">
  <wsdl:part name="parameters" element="tns:GetRange"/>
</wsdl:message>
<wsdl:message name="GetRangeSoapOut">
  <wsdl:part name="parameters" element="tns:GetRangeResponse"/>
</wsdl:message>
<wsdl:message name="GetSessionInformationSoapIn">
  <wsdl:part name="parameters" element="tns:GetSessionInformation"/>
</wsdl:message>
<wsdl:message name="GetSessionInformationSoapOut">
  <wsdl:part name="parameters" element="tns:GetSessionInformationResponse"/>
</wsdl:message>
<wsdl:message name="GetSheetNamesSoapIn">
  <wsdl:part name="parameters" element="tns:GetSheetNames"/>
</wsdl:message>
<wsdl:message name="GetSheetNamesSoapOut">

```



```

    <wsdl:part name="parameters" element="tns:GetSheetNamesResponse"/>
</wsdl:message>
<wsdl:message name="GetWorkbookConnectionsSoapIn">
    <wsdl:part name="parameters" element="tns:GetWorkbookConnections"/>
</wsdl:message>
<wsdl:message name="GetWorkbookConnectionsSoapOut">
    <wsdl:part name="parameters" element="tns:GetWorkbookConnectionsResponse"/>
</wsdl:message>
<wsdl:message name="GetWorkbookSoapIn">
    <wsdl:part name="parameters" element="tns:GetWorkbook"/>
</wsdl:message>
<wsdl:message name="GetWorkbookSoapOut">
    <wsdl:part name="parameters" element="tns:GetWorkbookResponse"/>
</wsdl:message>
<wsdl:message name="NewWorkbookSoapIn">
    <wsdl:part name="parameters" element="tns:NewWorkbook"/>
</wsdl:message>
<wsdl:message name="NewWorkbookSoapOut">
    <wsdl:part name="parameters" element="tns:NewWorkbookResponse"/>
</wsdl:message>
<wsdl:message name="OpenWorkbookExSoapIn">
    <wsdl:part name="parameters" element="tns:OpenWorkbookEx"/>
</wsdl:message>
<wsdl:message name="OpenWorkbookExSoapOut">
    <wsdl:part name="parameters" element="tns:OpenWorkbookExResponse"/>
</wsdl:message>
<wsdl:message name="OpenWorkbookForEditingSoapIn">
    <wsdl:part name="parameters" element="tns:OpenWorkbookForEditing"/>
</wsdl:message>
<wsdl:message name="OpenWorkbookForEditingSoapOut">
    <wsdl:part name="parameters" element="tns:OpenWorkbookForEditingResponse"/>
</wsdl:message>
<wsdl:message name="OpenWorkbookSoapIn">
    <wsdl:part name="parameters" element="tns:OpenWorkbook"/>
</wsdl:message>
<wsdl:message name="OpenWorkbookSoapOut">
    <wsdl:part name="parameters" element="tns:OpenWorkbookResponse"/>
</wsdl:message>
<wsdl:message name="RefreshExSoapIn">
    <wsdl:part name="parameters" element="tns:RefreshEx"/>
</wsdl:message>
<wsdl:message name="RefreshExSoapOut">
    <wsdl:part name="parameters" element="tns:RefreshExResponse"/>
</wsdl:message>
<wsdl:message name="RefreshSoapIn">
    <wsdl:part name="parameters" element="tns:Refresh"/>
</wsdl:message>
<wsdl:message name="RefreshSoapOut">
    <wsdl:part name="parameters" element="tns:RefreshResponse"/>
</wsdl:message>
<wsdl:message name="SaveWorkbookCopySoapIn">
    <wsdl:part name="parameters" element="tns:SaveWorkbookCopy"/>
</wsdl:message>
<wsdl:message name="SaveWorkbookCopySoapOut">
    <wsdl:part name="parameters" element="tns:SaveWorkbookCopyResponse"/>
</wsdl:message>
<wsdl:message name="SaveWorkbookSoapIn">
    <wsdl:part name="parameters" element="tns:SaveWorkbook"/>
</wsdl:message>

```

```
<wsdl:message name="SaveWorkbookSoapOut">
  <wsdl:part name="parameters" element="tns:SaveWorkbookResponse"/>
</wsdl:message>
<wsdl:message name="SetCalculationOptionsSoapIn">
  <wsdl:part name="parameters" element="tns:SetCalculationOptions"/>
</wsdl:message>
<wsdl:message name="SetCalculationOptionsSoapOut">
  <wsdl:part name="parameters" element="tns:SetCalculationOptionsResponse"/>
</wsdl:message>
<wsdl:message name="SetCellA1SoapIn">
  <wsdl:part name="parameters" element="tns:SetCellA1"/>
</wsdl:message>
<wsdl:message name="SetCellA1SoapOut">
  <wsdl:part name="parameters" element="tns:SetCellA1Response"/>
</wsdl:message>
<wsdl:message name="SetCellSoapIn">
  <wsdl:part name="parameters" element="tns:SetCell"/>
</wsdl:message>
<wsdl:message name="SetCellSoapOut">
  <wsdl:part name="parameters" element="tns:SetCellResponse"/>
</wsdl:message>
<wsdl:message name="SetParametersSoapIn">
  <wsdl:part name="parameters" element="tns:SetParameters"/>
</wsdl:message>
<wsdl:message name="SetParametersSoapOut">
  <wsdl:part name="parameters" element="tns:SetParametersResponse"/>
</wsdl:message>
<wsdl:message name="SetRangeA1SoapIn">
  <wsdl:part name="parameters" element="tns:SetRangeA1"/>
</wsdl:message>
<wsdl:message name="SetRangeA1SoapOut">
  <wsdl:part name="parameters" element="tns:SetRangeA1Response"/>
</wsdl:message>
<wsdl:message name="SetRangeSoapIn">
  <wsdl:part name="parameters" element="tns:SetRange"/>
</wsdl:message>
<wsdl:message name="SetRangeSoapOut">
  <wsdl:part name="parameters" element="tns:SetRangeResponse"/>
</wsdl:message>
</wsdl:definitions>
```

7 Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft Office SharePoint Server 2007
- Microsoft SharePoint Server 2010
- Microsoft SharePoint Server 2013

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

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

[<1> Section 2.2.9.1:](#) Office SharePoint Server 2007 does not support this error code.

[<2> Section 2.2.9.2:](#) Excel Services 2010 and Excel Services 2013 can use any of the values specified in section [2.2.9.1](#).

[<3> Section 3.1.2.1:](#) Excel Calculation Services 2007, Excel Calculation Services 2010 and Excel Calculation Services 2013 time out sessions (2) that are not edit sessions after a period of inactivity. This period of time is configurable.

[<4> Section 3.1.2.2:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 time out requests after a period of inactivity. This period of time is configurable.

[<5> Section 3.1.2.2:](#) In Excel Calculation Services 2007, Excel Calculation Services 2010 and Excel Calculation Services 2013, only some operations time out. Other operations can exceed the allotted time.

[<6> Section 3.1.4.9:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 use the sheet (1) that was the **active sheet** when the workbook was opened.

[<7> Section 3.1.4.13:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 use the sheet (1) that was the active sheet when the workbook was opened.

[<8> Section 3.1.4.18:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 attempt to fall back to a supported language when an unsupported language is requested.

[<9> Section 3.1.4.18:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 use the language set for the site (2) when none is specified in the request.

[<10> Section 3.1.4.18:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 attempt to fall back to a supported language when an unsupported language is requested.

[<11> Section 3.1.4.18:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 use the language set for the site (2) when none is specified in the request.

<12> [Section 3.1.4.19:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 have exceptions as noted in [\[MSDN-UFES\]](#).

<13> [Section 3.1.4.19:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 have exceptions as noted in [\[MSDN-UFES\]](#).

<14> [Section 3.1.4.19:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 attempt to fall back to a supported language when an unsupported language is requested.

<15> [Section 3.1.4.19:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 use the language set for the site (2) when none is specified in the current request.

<16> [Section 3.1.4.19:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 attempt to fall back to a supported language when an unsupported language is requested.

<17> [Section 3.1.4.19:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 use the language set for the site (2) when none is specified in the current request.

<18> [Section 3.1.4.21:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 have exceptions as noted in [\[MSDN-UFES\]](#).

<19> [Section 3.1.4.21:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 have exceptions as noted in [\[MSDN-UFES\]](#).

<20> [Section 3.1.4.21:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 attempt to fall back to a supported language when an unsupported language is requested.

<21> [Section 3.1.4.21:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 use the language set for the site (2) when none is specified in the current request.

<22> [Section 3.1.4.21:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 attempt to fall back to a supported language when an unsupported language is requested.

<23> [Section 3.1.4.21:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 use the language set for the site (2) when none is specified in the current request.

<24> [Section 3.1.4.28:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 use the sheet (1) that was the active sheet when the workbook was opened.

<25> [Section 3.1.4.30:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 set the values sent by the protocol client and do not return an error if **rangeValues** contains values in only one dimension (a single column (2) or a single row (2)), and if the first worksheet cell of **rangeCoordinates** falls on a workbook parameter.

<26> [Section 3.1.4.31:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 use the sheet (1) that was the active sheet when the workbook was opened.

<27> [Section 3.1.4.31:](#) Excel Services 2007, Excel Services 2010 and Excel Services 2013 set the values sent by the protocol client and do not return an error if **rangeValues** contains values in only one dimension (a single column or a single row), and if the first worksheet cell of **rangeCoordinates** falls on a workbook parameter.

8 Change Tracking

This section identifies changes that were made to the [MS-ESP] protocol document between the April 2014 and July 2014 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.
- The removal of a document from the documentation set.

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 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 changes were introduced. Minor editorial and formatting changes may have been made, but the technical content of the document is identical to the last released version.

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.

- 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 dochelp@microsoft.com.

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|--|--|-----------------------|--------------------------------|
| 2.2.9.1 Faults | Updated descriptions for error codes. | N | Content updated. |
| 2.2.9.2 Common List of Subcodes | Clarified the product behavior for Excel Services 2013. | N | Product behavior note updated. |
| 3.1.2.1 Session Timeout | Clarified the product behavior for Excel Calculation Services 2013. | N | Product behavior note updated. |
| 3.1.2.2 Request Timeout | Clarified the product behavior for Excel Services 2013 and Excel Calculation Services 2013. | N | Product behavior note updated. |
| 3.1.4 Message Processing Events and Sequencing Rules | Removed description of groups representing interfaces. | N | Content removed. |
| 3.1.4.9 GetCellA1 | Clarified the product behavior for Excel Services 2013 when neither sheetName nor rangeName specify a sheet. | N | Product behavior note updated. |
| 3.1.4.13 GetRangeA1 | Clarified the product behavior for Excel Services 2013 when neither sheetName nor rangeName specify a sheet. | N | Product behavior note updated. |
| 3.1.4.18 NewWorkbook | Clarified the product behavior for Excel Services 2013. | N | Product behavior note updated. |
| 3.1.4.19 OpenWorkbook | Clarified the product behavior for Excel Services 2013. | N | Product behavior note updated. |
| 3.1.4.21 OpenWorkbookForEditing | Clarified the product behavior for Excel Services 2013. | N | Product behavior note updated. |
| 3.1.4.28 | Clarified the product behavior for sheets | N | Product |

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Change type |
|-------------------------------------|--|-----------------------|--------------------------------|
| SetCellA1 | for Excel Services 2013. | | behavior note updated. |
| 3.1.4.30 SetRange | Clarified the product behavior for rangeValue for Excel Services 2013. | N | Product behavior note updated. |
| 3.1.4.31 SetRangeA1 | Clarified the product behavior for Excel Services 2013. | N | Product behavior note updated. |

9 Index

A

Abstract data model
[server](#) 38
[Applicability](#) 18
[ArrayOfAnyTypecomplex type](#) 20
[ArrayOfStatuscomplex type](#) 20
[ArrayOfStringcomplex type](#) 21
[Attribute groups](#) 25
[Attributes](#) 25

C

[Capability negotiation](#) 18
[CellErrorsimple type](#) 23
[Change tracking](#) 173
Common data structures ([section 2.2.9](#) 25, [section 2.2.9](#) 25)
[faults](#) 25
[Common list of subcodes - fault](#) 35
[Complex types](#) 20
[ArrayOfAnyType](#) 20
[ArrayOfStatus](#) 20
[ArrayOfString](#) 21
[RangeCoordinates](#) 21
[Status](#) 22
[TypeExporter](#) 22

D

Data model - abstract
[server](#) 38

E

Events
[local - server](#) 125
[timer - server](#) 125
Examples
[get the value of a worksheet cell](#) 126
[overview](#) 126
[refresh data and retrieve a workbook example](#) 129

F

Faults
[common list of subcodes](#) 35
[Fields - vendor-extensible](#) 18
[Full WSDL](#) 142

G

[Get the value of a worksheet cell example](#) 126
[Glossary](#) 13
[Groups](#) 25

I

[Implementer - security considerations](#) 141
[Index of security parameters](#) 141
[Informative references](#) 16
Initialization
[server](#) 38
[Introduction](#) 13

L

[List of subcodes – common data structures](#) 25
Local events
[server](#) 125

M

Message processing
[server](#) 38
Messages
[ArrayOfAnyTypecomplex type](#) 20
[ArrayOfStatuscomplex type](#) 20
[ArrayOfStringcomplex type](#) 21
[attribute groups](#) 25
[attributes](#) 25
[CellErrorsimple type](#) 23
common data structures ([section 2.2.9](#) 25, [section 2.2.9](#) 25)
[common list of subcodes - fault](#) 35
[complex types](#) 20
[elements](#) 20
[enumerated](#) 19
[faults](#) 25
[groups](#) 25
[namespaces](#) 19
[RangeCoordinatescomplex type](#) 21
[RangeReferencesimple type](#) 23
[SheetNamesimple type](#) 24
[simple types](#) 22
[Statuscomplex type](#) 22
[syntax](#) 19
[transport](#) 19
[TypeExportercomplex type](#) 22
[WorkbookTypesimple type](#) 24

N

[Namespaces](#) 19
[Normative references](#) 15

O

Operations
[Calculate](#) 40
[CalculateA1](#) 42
[CalculateWorkbook](#) 45
[CancelRequest](#) 48
[CloseWorkbook](#) 50
[EnsureWorkbookModel](#) 52
[GetApiVersion](#) 55

- [GetCell](#) 57
- [GetCellA1](#) 60
- [GetChartImageUrl](#) 63
- [GetPublishedItemNames](#) 65
- [GetRange](#) 69
- [GetRangeA1](#) 72
- [GetSessionInformation](#) 75
- [GetSheetNames](#) 77
- [GetWorkbook](#) 81
- [GetWorkbookConnections](#) 84
- [NewWorkbook](#) 86
- [OpenWorkbook](#) 89
- [OpenWorkbookEx](#) 92
- [OpenWorkbookForEditing](#) 94
- [Refresh](#) 97
- [RefreshEx](#) 100
- [SaveWorkbook](#) 102
- [SaveWorkbookCopy](#) 104
- [SetCalculationOptions](#) 107
- [SetCell](#) 110
- [SetCellA1](#) 113
- [SetParameters](#) 116
- [SetRange](#) 119
- [SetRangeA1](#) 122
- [Overview \(synopsis\)](#) 16
 - [published items returned](#) 17
 - [results returned](#) 17

P

- [Parameters - security index](#) 141
- [Preconditions](#) 18
- [Prerequisites](#) 18
- [Product behavior](#) 171
- Protocol Details
 - [overview](#) 37

R

- [RangeCoordinatescomplex type](#) 21
- [RangeReferencesimple type](#) 23
- [References](#) 15
 - [informative](#) 16
 - [normative](#) 15
- [Refresh data and retrieve a workbook example](#) 129
- [Relationship to other protocols](#) 17
- Request timeout
 - [timers](#) 38
- [Results returned](#) 17

S

- Security
 - [implementer considerations](#) 141
 - [parameter index](#) 141
- Sequencing rules
 - [server](#) 38
- Server
 - [abstract data model](#) 38
 - [Calculate operation](#) 40
 - [CalculateA1 operation](#) 42
 - [CalculateWorkbook operation](#) 45
- [CancelRequest operation](#) 48
- [CloseWorkbook operation](#) 50
- [EnsureWorkbookModel operation](#) 52
- [GetApiVersion operation](#) 55
- [GetCell operation](#) 57
- [GetCellA1 operation](#) 60
- [GetChartImageUrl operation](#) 63
- [GetPublishedItemNames operation](#) 65
- [GetRange operation](#) 69
- [GetRangeA1 operation](#) 72
- [GetSessionInformation operation](#) 75
- [GetSheetNames operation](#) 77
- [GetWorkbook operation](#) 81
- [GetWorkbookConnections operation](#) 84
- [initialization](#) 38
- [local events](#) 125
- [message processing](#) 38
- [NewWorkbook operation](#) 86
- [OpenWorkbook operation](#) 89
- [OpenWorkbookEx operation](#) 92
- [OpenWorkbookForEditing operation](#) 94
- [Refresh operation](#) 97
- [RefreshEx operation](#) 100
- [SaveWorkbook operation](#) 102
- [SaveWorkbookCopy operation](#) 104
- [sequencing rules](#) 38
- [SetCalculationOptions operation](#) 107
- [SetCell operation](#) 110
- [SetCellA1 operation](#) 113
- [SetParameters operation](#) 116
- [SetRange operation](#) 119
- [SetRangeA1 operation](#) 122
- [timer events](#) 125
- Session timeout
 - [timers](#) 38
- [SheetNamesimple type](#) 24
- [Simple types](#) 22
 - [CellError](#) 23
 - [RangeReference](#) 23
 - [SheetName](#) 24
 - [WorkbookType](#) 24
- [Standards assignments](#) 18
- [Statuscomplex type](#) 22
- Syntax
 - [messages - overview](#) 19

T

- Timer events
 - [server](#) 125
- Timers
 - [request timeout](#) 38
 - [session timeout](#) 38
- [Tracking changes](#) 173
- [Transport](#) 19
- [TypeExportercomplex type](#) 22
- Types
 - [complex](#) 20
 - [simple](#) 22

V

[Vendor-extensible fields](#) 18
[Versioning](#) 18

W

[WorkbookTypesimple type](#) 24
[WSDL](#) 142