**[MS-VGSFF]:**

**Visio Graphics Service (.vdw) File Format**

Intellectual Property Rights Notice for Open Specifications Documentation

* **Technical Documentation.** Microsoft publishes Open Specifications documentation (“this documentation”) for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.
* **Copyrights**. This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.
* **No Trade Secrets**. Microsoft does not claim any trade secret rights in this documentation.
* **Patents**. Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft [Open Specifications Promise](https://go.microsoft.com/fwlink/?LinkId=214445) or the [Microsoft Community Promise](https://go.microsoft.com/fwlink/?LinkId=214448). If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting [iplg@microsoft.com](mailto:iplg@microsoft.com).
* **License Programs**. To see all of the protocols in scope under a specific license program and the associated patents, visit the [Patent Map](https://aka.ms/AA9ufj8).
* **Trademarks**. The names of companies and products contained in this documentation might be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit [www.microsoft.com/trademarks](https://www.microsoft.com/trademarks).
* **Fictitious Names**. The example companies, organizations, products, domain names, email addresses, logos, people, places, and events that are depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

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

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

**Support.** For questions and support, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com).

**Revision Summary**

| Date | Revision History | Revision Class | Comments |
| --- | --- | --- | --- |
| 7/13/2009 | 0.1 | Major | Initial Availability |
| 8/28/2009 | 0.2 | Editorial | Revised and edited the technical content |
| 11/6/2009 | 0.3 | Editorial | Revised and edited the technical content |
| 2/19/2010 | 1.0 | Major | Updated and revised the technical content |
| 3/31/2010 | 1.01 | Editorial | Revised and edited the technical content |
| 4/30/2010 | 1.02 | Editorial | Revised and edited the technical content |
| 6/7/2010 | 1.03 | Editorial | Revised and edited the technical content |
| 6/29/2010 | 1.04 | Editorial | Changed language and formatting in the technical content. |
| 7/23/2010 | 1.04 | None | No changes to the meaning, language, or formatting of the technical content. |
| 9/27/2010 | 1.04 | None | No changes to the meaning, language, or formatting of the technical content. |
| 11/15/2010 | 1.04 | None | No changes to the meaning, language, or formatting of the technical content. |
| 12/17/2010 | 1.04 | None | No changes to the meaning, language, or formatting of the technical content. |
| 3/18/2011 | 1.04 | None | No changes to the meaning, language, or formatting of the technical content. |
| 6/10/2011 | 1.04 | None | No changes to the meaning, language, or formatting of the technical content. |
| 1/20/2012 | 2.0 | Major | Significantly changed the technical content. |
| 4/11/2012 | 2.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 7/16/2012 | 2.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 9/12/2012 | 2.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 10/8/2012 | 2.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 2/11/2013 | 2.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 7/30/2013 | 3.0 | Major | Significantly changed the technical content. |
| 11/18/2013 | 3.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 2/10/2014 | 3.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 4/30/2014 | 3.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 7/31/2014 | 3.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 10/30/2014 | 3.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 3/16/2015 | 4.0 | Major | Significantly changed the technical content. |
| 9/4/2015 | 4.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 7/15/2016 | 4.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 9/14/2016 | 4.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 9/19/2017 | 4.1 | Minor | Clarified the meaning of the technical content. |
| 7/24/2018 | 5.0 | Major | Significantly changed the technical content. |
| 10/1/2018 | 6.0 | Major | Significantly changed the technical content. |
| 12/11/2018 | 6.0 | None | No changes to the meaning, language, or formatting of the technical content. |
| 4/22/2021 | 7.0 | Major | Significantly changed the technical content. |
| 8/17/2021 | 8.0 | Major | Significantly changed the technical content. |

Table of Contents

[1 Introduction 12](#_Toc79556196)

[1.1 Glossary 12](#_Toc79556197)

[1.2 References 15](#_Toc79556198)

[1.2.1 Normative References 15](#_Toc79556199)

[1.2.2 Informative References 16](#_Toc79556200)

[1.3 Overview 16](#_Toc79556201)

[1.4 Relationship to Protocols and Other Structures 17](#_Toc79556202)

[1.5 Applicability Statement 17](#_Toc79556203)

[1.6 Versioning and Localization 17](#_Toc79556204)

[1.7 Vendor-Extensible Fields 17](#_Toc79556205)

[2 Structures 18](#_Toc79556206)

[2.1 File Structure Overview 18](#_Toc79556207)

[2.1.1 Compound File 18](#_Toc79556208)

[2.1.2 Streams 19](#_Toc79556209)

[2.1.3 Package 19](#_Toc79556210)

[2.1.4 Parts 19](#_Toc79556211)

[2.1.5 Relationships 19](#_Toc79556212)

[2.1.6 Part Enumeration 19](#_Toc79556213)

[2.1.6.1 ShapeGraphic 20](#_Toc79556214)

[2.1.6.2 Fonts 20](#_Toc79556215)

[2.1.6.3 Images 21](#_Toc79556216)

[2.1.6.4 App 21](#_Toc79556217)

[2.1.6.5 Core 21](#_Toc79556218)

[2.1.6.6 DataBinding 22](#_Toc79556219)

[2.1.6.7 DataConnection 22](#_Toc79556220)

[2.1.6.8 DataGraphicDefinition 22](#_Toc79556221)

[2.1.6.9 DataGraphic 22](#_Toc79556222)

[2.1.6.10 ShapeInfo 23](#_Toc79556223)

[2.1.6.11 ShapeTextBinding 23](#_Toc79556224)

[2.1.6.12 ShapeOutline 23](#_Toc79556225)

[2.1.6.13 Rels 24](#_Toc79556226)

[2.1.6.14 ContentType 24](#_Toc79556227)

[2.2 Conceptual Overview 24](#_Toc79556228)

[2.2.1 Web Drawing 24](#_Toc79556229)

[2.2.2 Drawing Page 24](#_Toc79556230)

[2.2.3 Shape 25](#_Toc79556231)

[2.2.3.1 Shape Identification 25](#_Toc79556232)

[2.2.3.2 Shape Visualization 25](#_Toc79556233)

[2.2.3.3 Shape Selection 26](#_Toc79556234)

[2.2.3.4 Shape Hyperlinks 26](#_Toc79556235)

[2.2.3.5 Shape Data 26](#_Toc79556236)

[2.2.4 Data Connectivity and Refresh 26](#_Toc79556237)

[2.2.4.1 Data Connections 26](#_Toc79556238)

[2.2.4.2 Recordset 27](#_Toc79556239)

[2.2.4.3 Recordset Refresh 27](#_Toc79556240)

[2.2.4.4 Recordset Row Addressing 27](#_Toc79556241)

[2.2.4.4.1 Row Order Method 27](#_Toc79556242)

[2.2.4.4.2 Primary Key Method 27](#_Toc79556243)

[2.2.5 Data Binding to Web Drawing Elements 27](#_Toc79556244)

[2.2.5.1 Data Binding 27](#_Toc79556245)

[2.2.5.2 Diagram Update 28](#_Toc79556246)

[2.2.5.3 Data Graphics 28](#_Toc79556247)

[2.2.6 Recalculating Shape Properties 28](#_Toc79556248)

[2.2.6.1 Color Table 29](#_Toc79556249)

[2.2.6.2 Formulas 29](#_Toc79556250)

[2.2.6.2.1 Formula Expression 29](#_Toc79556251)

[2.2.6.2.2 Formula Evaluation 29](#_Toc79556252)

[2.2.6.2.3 Parse Tokens 29](#_Toc79556253)

[2.2.6.2.3.1 Control Tokens 29](#_Toc79556254)

[2.2.6.2.3.2 Function Tokens 29](#_Toc79556255)

[2.2.6.2.3.3 Operand Tokens 30](#_Toc79556256)

[2.2.6.2.3.3.1 String Values 30](#_Toc79556257)

[2.2.6.2.3.3.2 Numeric Values 30](#_Toc79556258)

[2.2.6.2.3.3.3 Boolean Values 31](#_Toc79556259)

[2.2.6.2.3.3.4 Currency Values 31](#_Toc79556260)

[2.2.6.2.3.3.5 Color Values 31](#_Toc79556261)

[2.2.6.2.3.3.6 Date Values 31](#_Toc79556262)

[2.2.6.2.3.3.7 Error Values 31](#_Toc79556263)

[2.2.6.2.4 Evaluation Stack 31](#_Toc79556264)

[2.2.6.3 Unit Number 32](#_Toc79556265)

[2.3 ShapeGraphic XML Part 32](#_Toc79556266)

[2.3.1 XAML Terminology 32](#_Toc79556267)

[2.3.2 XAML Resources 32](#_Toc79556268)

[2.3.2.1 Fonts 32](#_Toc79556269)

[2.3.2.2 Images 35](#_Toc79556270)

[2.3.3 XAML Shapes 36](#_Toc79556271)

[2.3.4 XAML Recalculated Shapes 36](#_Toc79556272)

[2.3.4.1 Sheet Elements 36](#_Toc79556273)

[2.3.4.1.1 Shape Transform 36](#_Toc79556274)

[2.3.4.2 Formatting Elements 37](#_Toc79556275)

[2.3.4.2.1 Shadow Canvas 37](#_Toc79556276)

[2.3.4.2.2 Fill Attributes 37](#_Toc79556277)

[2.3.4.2.3 Stroke Attributes 37](#_Toc79556278)

[2.3.4.3 Geometry Elements 37](#_Toc79556279)

[2.3.4.3.1 Geometry Path Specifications 38](#_Toc79556280)

[2.3.4.4 Model Elements 38](#_Toc79556281)

[2.3.4.4.1 Model Paths 38](#_Toc79556282)

[2.3.4.4.2 Model Ellipses 38](#_Toc79556283)

[2.3.4.5 Text Elements 39](#_Toc79556284)

[2.3.4.5.1 Glyph Canvas 39](#_Toc79556285)

[2.3.4.5.2 Text Model 39](#_Toc79556286)

[2.3.4.5.3 Text Run 40](#_Toc79556287)

[2.3.4.6 Image Elements 40](#_Toc79556288)

[2.4 XML Parts 40](#_Toc79556289)

[2.4.1 Introduction 41](#_Toc79556290)

[2.4.2 App XML Part 41](#_Toc79556291)

[2.4.3 Core XML Part 41](#_Toc79556292)

[2.4.4 DataBinding XML Part 41](#_Toc79556293)

[2.4.4.1 Global Elements 41](#_Toc79556294)

[2.4.4.1.1 BindingConnections 41](#_Toc79556295)

[2.4.4.2 Complex Types 41](#_Toc79556296)

[2.4.4.2.1 CT\_PrimaryKeyValue 41](#_Toc79556297)

[2.4.4.2.2 CT\_PrimaryKeyValues 41](#_Toc79556298)

[2.4.4.2.3 CT\_Binding 42](#_Toc79556299)

[2.4.4.2.4 CT\_Bindings 42](#_Toc79556300)

[2.4.4.2.5 CT\_BindingConnection 43](#_Toc79556301)

[2.4.4.2.6 CT\_BindingConnections 43](#_Toc79556302)

[2.4.5 DataConnection XML Part 44](#_Toc79556303)

[2.4.5.1 Global Elements 44](#_Toc79556304)

[2.4.5.1.1 Connections 44](#_Toc79556305)

[2.4.5.2 Complex Types 44](#_Toc79556306)

[2.4.5.2.1 CT\_DataConnection 44](#_Toc79556307)

[2.4.5.2.2 CT\_DataConnections 45](#_Toc79556308)

[2.4.5.2.3 CT\_PrimaryKey 45](#_Toc79556309)

[2.4.5.2.4 CT\_PrimaryKeys 46](#_Toc79556310)

[2.4.5.2.5 CT\_DataColumn 46](#_Toc79556311)

[2.4.5.2.6 CT\_DataColumns 48](#_Toc79556312)

[2.4.5.2.7 CT\_DataRecordset 48](#_Toc79556313)

[2.4.5.2.8 CT\_DataRecordsets 50](#_Toc79556314)

[2.4.5.2.9 CT\_Connections 50](#_Toc79556315)

[2.4.6 DataGraphic XML Part 50](#_Toc79556316)

[2.4.6.1 Global Elements 50](#_Toc79556317)

[2.4.6.1.1 DataGraphics 50](#_Toc79556318)

[2.4.6.2 Complex Types 51](#_Toc79556319)

[2.4.6.2.1 CT\_DataGraphicDef 51](#_Toc79556320)

[2.4.6.2.2 CT\_DataGraphicDefinitions 51](#_Toc79556321)

[2.4.6.2.3 CT\_Geometry 51](#_Toc79556322)

[2.4.6.2.4 CT\_Sheet 53](#_Toc79556323)

[2.4.6.2.5 CT\_Formulas 56](#_Toc79556324)

[2.4.6.2.6 CT\_FormulaReferences 57](#_Toc79556325)

[2.4.6.2.7 CT\_DataGraphics\_Shape 57](#_Toc79556326)

[2.4.6.2.8 CT\_DataGraphics\_Page 58](#_Toc79556327)

[2.4.6.2.9 CT\_DataGraphics 59](#_Toc79556328)

[2.4.7 DataGraphicDefinition XML Part 60](#_Toc79556329)

[2.4.7.1 Global Elements 60](#_Toc79556330)

[2.4.7.1.1 DataGraphicDefs 60](#_Toc79556331)

[2.4.7.2 Complex Types 60](#_Toc79556332)

[2.4.7.2.1 CT\_IconSetRule 60](#_Toc79556333)

[2.4.7.2.2 CT\_IconSetDef 60](#_Toc79556334)

[2.4.7.2.3 CT\_DataGraphicDefs 61](#_Toc79556335)

[2.4.8 ShapeInfo XML Part 61](#_Toc79556336)

[2.4.8.1 Global Elements 61](#_Toc79556337)

[2.4.8.1.1 Page 61](#_Toc79556338)

[2.4.8.2 Complex Types 61](#_Toc79556339)

[2.4.8.2.1 CT\_PageInfo 61](#_Toc79556340)

[2.4.8.2.2 CT\_Pages 62](#_Toc79556341)

[2.4.8.2.3 CT\_ShapeData 62](#_Toc79556342)

[2.4.8.2.4 CT\_ShapeDataItems 64](#_Toc79556343)

[2.4.8.2.5 CT\_Hyperlink 64](#_Toc79556344)

[2.4.8.2.6 CT\_Hyperlinks 65](#_Toc79556345)

[2.4.8.2.7 CT\_ShapeInfo 66](#_Toc79556346)

[2.4.8.2.8 CT\_Page 67](#_Toc79556347)

[2.4.9 ShapeOutline XML Part 71](#_Toc79556348)

[2.4.9.1 Global Elements 71](#_Toc79556349)

[2.4.9.1.1 Page 71](#_Toc79556350)

[2.4.9.2 Complex Types 71](#_Toc79556351)

[2.4.9.2.1 CT\_Path 71](#_Toc79556352)

[2.4.9.2.2 CT\_Shape 72](#_Toc79556353)

[2.4.9.2.3 CT\_Shapes 73](#_Toc79556354)

[2.4.9.2.4 CT\_ShapeOutline\_Page 73](#_Toc79556355)

[2.4.10 ShapeTextBinding XML Part 73](#_Toc79556356)

[2.4.10.1 Global Elements 73](#_Toc79556357)

[2.4.10.1.1 Page 73](#_Toc79556358)

[2.4.10.2 Complex Types 74](#_Toc79556359)

[2.4.10.2.1 CT\_TextRun 74](#_Toc79556360)

[2.4.10.2.2 CT\_ShapeText 75](#_Toc79556361)

[2.4.10.2.3 CT\_ShapeTextBinding\_Page 76](#_Toc79556362)

[2.5 Formula Evaluation and Shape Property Recalculation 76](#_Toc79556363)

[2.5.1 Introduction 76](#_Toc79556364)

[2.5.2 Formula ABNF and Full Grammar Definition 76](#_Toc79556365)

[2.5.3 Function Token Table 78](#_Toc79556366)

[2.5.4 Function Token Definitions 82](#_Toc79556367)

[2.5.4.1 Abs 82](#_Toc79556368)

[2.5.4.2 ACos 83](#_Toc79556369)

[2.5.4.3 Add 83](#_Toc79556370)

[2.5.4.4 And 85](#_Toc79556371)

[2.5.4.5 Ang360 85](#_Toc79556372)

[2.5.4.6 ASin 85](#_Toc79556373)

[2.5.4.7 ATan 86](#_Toc79556374)

[2.5.4.8 ATan2 86](#_Toc79556375)

[2.5.4.9 BitAnd 87](#_Toc79556376)

[2.5.4.10 BitNot 87](#_Toc79556377)

[2.5.4.11 BitOr 88](#_Toc79556378)

[2.5.4.12 BitXor 88](#_Toc79556379)

[2.5.4.13 Blend 89](#_Toc79556380)

[2.5.4.14 Bound 89](#_Toc79556381)

[2.5.4.15 Cat 91](#_Toc79556382)

[2.5.4.16 Ceiling 91](#_Toc79556383)

[2.5.4.17 CellIsThemed 92](#_Toc79556384)

[2.5.4.18 Char 92](#_Toc79556385)

[2.5.4.19 Cos 93](#_Toc79556386)

[2.5.4.20 CosH 93](#_Toc79556387)

[2.5.4.21 CY 94](#_Toc79556388)

[2.5.4.22 Date 94](#_Toc79556389)

[2.5.4.23 DateTime 95](#_Toc79556390)

[2.5.4.24 DateValue 96](#_Toc79556391)

[2.5.4.25 Day 96](#_Toc79556392)

[2.5.4.26 DayOfYear 97](#_Toc79556393)

[2.5.4.27 Deg 97](#_Toc79556394)

[2.5.4.28 Div 98](#_Toc79556395)

[2.5.4.29 EEQ 99](#_Toc79556396)

[2.5.4.30 EGE 100](#_Toc79556397)

[2.5.4.31 EGT 100](#_Toc79556398)

[2.5.4.32 ELE 101](#_Toc79556399)

[2.5.4.33 ELT 101](#_Toc79556400)

[2.5.4.34 ENE 102](#_Toc79556401)

[2.5.4.35 FEQ 102](#_Toc79556402)

[2.5.4.36 FGE 103](#_Toc79556403)

[2.5.4.37 FGT 103](#_Toc79556404)

[2.5.4.38 Find 104](#_Toc79556405)

[2.5.4.39 FLE 105](#_Toc79556406)

[2.5.4.40 Floor 105](#_Toc79556407)

[2.5.4.41 FLT 106](#_Toc79556408)

[2.5.4.42 FNE 106](#_Toc79556409)

[2.5.4.43 FormatEx 107](#_Toc79556410)

[2.5.4.44 Hour 109](#_Toc79556411)

[2.5.4.45 HSL 109](#_Toc79556412)

[2.5.4.46 Hue 110](#_Toc79556413)

[2.5.4.47 HueDiff 110](#_Toc79556414)

[2.5.4.48 Index 111](#_Toc79556415)

[2.5.4.49 Int 112](#_Toc79556416)

[2.5.4.50 IntersectX 112](#_Toc79556417)

[2.5.4.51 IntersectY 113](#_Toc79556418)

[2.5.4.52 Intup 114](#_Toc79556419)

[2.5.4.53 IsErr 114](#_Toc79556420)

[2.5.4.54 IsErrNA 115](#_Toc79556421)

[2.5.4.55 IsError 115](#_Toc79556422)

[2.5.4.56 IsErrValue 115](#_Toc79556423)

[2.5.4.57 Left 116](#_Toc79556424)

[2.5.4.58 Len 116](#_Toc79556425)

[2.5.4.59 Ln 117](#_Toc79556426)

[2.5.4.60 Log10 117](#_Toc79556427)

[2.5.4.61 Lookup 118](#_Toc79556428)

[2.5.4.62 Lower 118](#_Toc79556429)

[2.5.4.63 Lum 119](#_Toc79556430)

[2.5.4.64 LumDiff 119](#_Toc79556431)

[2.5.4.65 Magnitude 120](#_Toc79556432)

[2.5.4.66 Max 121](#_Toc79556433)

[2.5.4.67 Mid 121](#_Toc79556434)

[2.5.4.68 Min 122](#_Toc79556435)

[2.5.4.69 Minute 122](#_Toc79556436)

[2.5.4.70 Modulus 123](#_Toc79556437)

[2.5.4.71 Month 123](#_Toc79556438)

[2.5.4.72 MsoShade 124](#_Toc79556439)

[2.5.4.73 MsoTint 124](#_Toc79556440)

[2.5.4.74 Mul 125](#_Toc79556441)

[2.5.4.75 Not 126](#_Toc79556442)

[2.5.4.76 Now 127](#_Toc79556443)

[2.5.4.77 Or 127](#_Toc79556444)

[2.5.4.78 Pct 127](#_Toc79556445)

[2.5.4.79 Pi 128](#_Toc79556446)

[2.5.4.80 Pnt 128](#_Toc79556447)

[2.5.4.81 Pow 129](#_Toc79556448)

[2.5.4.82 Rad 130](#_Toc79556449)

[2.5.4.83 Rand 130](#_Toc79556450)

[2.5.4.84 Replace 130](#_Toc79556451)

[2.5.4.85 RGB 131](#_Toc79556452)

[2.5.4.86 Right 132](#_Toc79556453)

[2.5.4.87 Round 132](#_Toc79556454)

[2.5.4.88 Sat 133](#_Toc79556455)

[2.5.4.89 SatDiff 133](#_Toc79556456)

[2.5.4.90 Second 134](#_Toc79556457)

[2.5.4.91 SetAtRef 134](#_Toc79556458)

[2.5.4.92 SetAtRefEval 135](#_Toc79556459)

[2.5.4.93 SetAtRefExpr 135](#_Toc79556460)

[2.5.4.94 Shade 136](#_Toc79556461)

[2.5.4.95 ShapeText 136](#_Toc79556462)

[2.5.4.96 Sign 137](#_Toc79556463)

[2.5.4.97 Sin 137](#_Toc79556464)

[2.5.4.98 SinH 138](#_Toc79556465)

[2.5.4.99 Sqrt 138](#_Toc79556466)

[2.5.4.100 StrSame 139](#_Toc79556467)

[2.5.4.101 StrSameEx 139](#_Toc79556468)

[2.5.4.102 Sub 140](#_Toc79556469)

[2.5.4.103 Substitute 142](#_Toc79556470)

[2.5.4.104 Sum 143](#_Toc79556471)

[2.5.4.105 Tan 143](#_Toc79556472)

[2.5.4.106 TanH 144](#_Toc79556473)

[2.5.4.107 TextHeight 144](#_Toc79556474)

[2.5.4.108 TextWidth 145](#_Toc79556475)

[2.5.4.109 Time 145](#_Toc79556476)

[2.5.4.110 TimeValue 146](#_Toc79556477)

[2.5.4.111 Tint 147](#_Toc79556478)

[2.5.4.112 Tone 147](#_Toc79556479)

[2.5.4.113 Trim 148](#_Toc79556480)

[2.5.4.114 Trunc 149](#_Toc79556481)

[2.5.4.115 UMinus 150](#_Toc79556482)

[2.5.4.116 UPlus 150](#_Toc79556483)

[2.5.4.117 Upper 150](#_Toc79556484)

[2.5.4.118 WeekDay 151](#_Toc79556485)

[2.5.4.119 Year 151](#_Toc79556486)

[2.5.5 Parse Token Table 152](#_Toc79556487)

[2.5.6 Parse Token Definitions 154](#_Toc79556488)

[2.5.6.1 PtgAcre 154](#_Toc79556489)

[2.5.6.2 PtgAngDD 154](#_Toc79556490)

[2.5.6.3 PtgAngDft 154](#_Toc79556491)

[2.5.6.4 PtgAngDMS 155](#_Toc79556492)

[2.5.6.5 PtgAngRad 155](#_Toc79556493)

[2.5.6.6 PtgBool 155](#_Toc79556494)

[2.5.6.7 PtgColorRGB 155](#_Toc79556495)

[2.5.6.8 PtgCy 156](#_Toc79556496)

[2.5.6.9 PtgDataBinding 156](#_Toc79556497)

[2.5.6.10 PtgDate 156](#_Toc79556498)

[2.5.6.11 PtgEDay 156](#_Toc79556499)

[2.5.6.12 PtgEHour 157](#_Toc79556500)

[2.5.6.13 PtgEMin 157](#_Toc79556501)

[2.5.6.14 PtgErr 157](#_Toc79556502)

[2.5.6.15 PtgESec 157](#_Toc79556503)

[2.5.6.16 PtgEWeek 158](#_Toc79556504)

[2.5.6.17 PtgFunc 158](#_Toc79556505)

[2.5.6.18 PtgFuncVar 158](#_Toc79556506)

[2.5.6.19 PtgHectare 158](#_Toc79556507)

[2.5.6.20 PtgJmp 159](#_Toc79556508)

[2.5.6.21 PtgJmpF 159](#_Toc79556509)

[2.5.6.22 PtgJmpLabel 159](#_Toc79556510)

[2.5.6.23 PtgJmpT 159](#_Toc79556511)

[2.5.6.24 PtgMissArg 159](#_Toc79556512)

[2.5.6.25 PtgNoOp 160](#_Toc79556513)

[2.5.6.26 PtgNum 160](#_Toc79556514)

[2.5.6.27 PtgNumCM 160](#_Toc79556515)

[2.5.6.28 PtgNumDft 160](#_Toc79556516)

[2.5.6.29 PtgNumF 160](#_Toc79556517)

[2.5.6.30 PtgNumFI 161](#_Toc79556518)

[2.5.6.31 PtgNumI 161](#_Toc79556519)

[2.5.6.32 PtgNumKM 161](#_Toc79556520)

[2.5.6.33 PtgNumM 161](#_Toc79556521)

[2.5.6.34 PtgNumMI 161](#_Toc79556522)

[2.5.6.35 PtgNumMM 162](#_Toc79556523)

[2.5.6.36 PtgNumMultiDim 162](#_Toc79556524)

[2.5.6.37 PtgNumNM 162](#_Toc79556525)

[2.5.6.38 PtgNumPct 162](#_Toc79556526)

[2.5.6.39 PtgNumYards 163](#_Toc79556527)

[2.5.6.40 PtgPageDft 163](#_Toc79556528)

[2.5.6.41 PtgPnt 163](#_Toc79556529)

[2.5.6.42 PtgPop 163](#_Toc79556530)

[2.5.6.43 PtgPushTop 164](#_Toc79556531)

[2.5.6.44 PtgRecalcRef 164](#_Toc79556532)

[2.5.6.45 PtgStr1 164](#_Toc79556533)

[2.5.6.46 PtgTDurDft 164](#_Toc79556534)

[2.5.6.47 PtgTypCD 164](#_Toc79556535)

[2.5.6.48 PtgTypCi 165](#_Toc79556536)

[2.5.6.49 PtgTypDft 165](#_Toc79556537)

[2.5.6.50 PtgTypDi 165](#_Toc79556538)

[2.5.6.51 PtgTypPi 165](#_Toc79556539)

[2.5.6.52 PtgTypPP 165](#_Toc79556540)

[2.5.6.53 PtgTypPt 166](#_Toc79556541)

[2.5.6.54 PtgUnsWord 166](#_Toc79556542)

[2.5.7 Custom Input Type Definitions 166](#_Toc79556543)

[2.5.7.1 vBoolean 166](#_Toc79556544)

[2.5.7.2 vColor 167](#_Toc79556545)

[2.5.7.3 vDouble 167](#_Toc79556546)

[2.5.7.4 vDoubleEx 168](#_Toc79556547)

[2.5.7.5 vFloat 168](#_Toc79556548)

[2.5.7.6 vSignedInt 168](#_Toc79556549)

[2.5.7.7 vSignedLong 168](#_Toc79556550)

[2.5.7.8 vString 169](#_Toc79556551)

[2.5.7.9 vUnsignedInt 169](#_Toc79556552)

[2.5.7.10 vUnsignedLong 169](#_Toc79556553)

[2.5.8 Custom Token Groupings 170](#_Toc79556554)

[2.5.8.1 vAngle 170](#_Toc79556555)

[2.5.8.2 vAny 170](#_Toc79556556)

[2.5.8.3 vNum 170](#_Toc79556557)

[2.5.8.4 vNumAny 170](#_Toc79556558)

[2.5.8.5 vUnitType 170](#_Toc79556559)

[2.5.9 Custom Internal Unit Types 170](#_Toc79556560)

[2.5.9.1 angleInternalUnitNumber 171](#_Toc79556561)

[2.5.9.2 durationInternalUnitNumber 171](#_Toc79556562)

[2.5.9.3 lengthInternalUnitNumber 171](#_Toc79556563)

[2.5.9.4 typographicInternalUnitNumber 171](#_Toc79556564)

[2.5.10 Custom Structures 171](#_Toc79556565)

[2.5.10.1 vCalendar 171](#_Toc79556566)

[2.5.10.2 vCurrencyID 172](#_Toc79556567)

[2.5.10.3 vFormatString 175](#_Toc79556568)

[2.5.10.4 vLanguageID 175](#_Toc79556569)

[2.5.10.5 vServerAction 175](#_Toc79556570)

[2.5.10.6 vUnitLabel 176](#_Toc79556571)

[3 Structure Examples 177](#_Toc79556572)

[3.1 Document with a Shape on a Page 177](#_Toc79556573)

[3.1.1 App XML Part 177](#_Toc79556574)

[3.1.2 ShapeInfo XML Part 178](#_Toc79556575)

[3.1.3 ShapeOutline XML Part 179](#_Toc79556576)

[3.1.4 XAML 180](#_Toc79556577)

[3.2 Document with Recalculated Visual Properties 181](#_Toc79556578)

[3.2.1 DataBinding XML Part 182](#_Toc79556579)

[3.2.2 DataConnection XML Part 183](#_Toc79556580)

[3.2.3 DataGraphic XML Part 184](#_Toc79556581)

[3.2.4 ShapeInfo XML Part 187](#_Toc79556582)

[3.2.5 ShapeTextBinding XML Part 189](#_Toc79556583)

[3.2.6 XAML 190](#_Toc79556584)

[3.3 Formula Evaluation 193](#_Toc79556585)

[4 Security 196](#_Toc79556586)

[4.1 Security Considerations for Implementers 196](#_Toc79556587)

[4.2 Index of Security Fields 196](#_Toc79556588)

[5 Appendix A: Full XML Schemas 197](#_Toc79556589)

[5.1 app Schema 197](#_Toc79556590)

[5.2 core Schema 197](#_Toc79556591)

[5.3 DataBinding Schema 197](#_Toc79556592)

[5.4 DataConnection Schema 198](#_Toc79556593)

[5.5 DataGraphicDefinition Schema 199](#_Toc79556594)

[5.6 DataGraphic Schema 199](#_Toc79556595)

[5.7 ShapeInfo Schema 201](#_Toc79556596)

[5.8 ShapeTextBinding Schema 203](#_Toc79556597)

[5.9 ShapeOutline Schema 204](#_Toc79556598)

[6 Appendix B: Product Behavior 205](#_Toc79556599)

[7 Change Tracking 206](#_Toc79556600)

[8 Index 207](#_Toc79556601)

# Introduction

The Visio Graphics Service (.vdw) File Format describes a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25), which is a collection of [drawing pages](#Section_ebba8748c5a54ac7a4191c6a70307640), [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b), fonts, images, [data connections](#Section_28d3d364de644cada48144231aed4d7b), and [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) information that can be rendered as a vector or raster drawing.

Sections 1.7 and 2 of this specification are normative. All other sections and examples in this specification are informative.

## Glossary

This document uses the following terms:

**A1**: A [**reference style**](#gt_71d7e4ad-d6b6-49d9-b247-e2ebba1469e9) in which each column is identified sequentially from left-to-right with a letter or series of letters in alphabetical order. Column headings are ordered A-Z, then AA-AZ, BA-BZ... ZA-ZZ, AAA-AAZ, and so forth. Each row is numbered sequentially from the top down.

**ActiveX Data Objects (ADO)**: A data access interface that connects to, retrieves, manipulates, and updates data in Object Linking and Embedding (OLE) database-compliant data sources.

**add-in**: Supplemental functionality that is provided by an external application or macro to extend the capabilities of an application.

**American National Standards Institute (ANSI) character set**: A character set defined by a code page approved by the American National Standards Institute (ANSI). The term "ANSI" as used to signify Windows code pages is a historical reference and a misnomer that persists in the Windows community. The source of this misnomer stems from the fact that the Windows code page 1252 was originally based on an ANSI draft, which became International Organization for Standardization (ISO) Standard 8859-1 [[ISO/IEC-8859-1]](https://go.microsoft.com/fwlink/?LinkId=90689). In Windows, the ANSI character set can be any of the following code pages: 1252, 1250, 1251, 1253, 1254, 1255, 1256, 1257, 1258, 874, 932, 936, 949, or 950. For example, "ANSI application" is usually a reference to a non-[**Unicode**](#gt_c305d0ab-8b94-461a-bd76-13b40cb8c4d8) or code-page-based application. Therefore, "ANSI character set" is often misused to refer to one of the character sets defined by a Windows code page that can be used as an active system code page; for example, character sets defined by code page 1252 or character sets defined by code page 950. Windows is now based on [**Unicode**](#gt_c305d0ab-8b94-461a-bd76-13b40cb8c4d8), so the use of ANSI character sets is strongly discouraged unless they are used to interoperate with legacy applications or legacy data.

**assembly name**: The name of a collection of one or more files that is versioned and deployed as a unit. See also assembly.

**Boolean**: An operation or expression that can be evaluated only as either true or false.

**bounding rectangle**: A frame that encompasses an object. A bounding rectangle is not rotated and, therefore, always aligns along the x and y axes.

**class name**: The name that is used to refer to a class module that provides an implementation of a behavior.

**color space**: A system that describes color numerically by mapping color components to a multidimensional coordinate system. The number of dimensions is typically two, three, or four. For example, if colors are expressed as a combination of the three components red, green, and blue, a three-dimensional space can describe all possible colors. Grayscale colors can be mapped to a two-dimensional color space. If transparency is considered a component, four dimensions are appropriate. Also referred to as color model.

**connection string**: A series of arguments, delimited by a semicolon, that defines the location of a database and how to connect to it.

**culture name**: A part of a language identification tagging system, as described in [[RFC1766]](https://go.microsoft.com/fwlink/?LinkId=120475). Culture names adhere to the format "<languagecode2>-<country/regioncode2>." If a two-letter language code is not available, a three-letter code that is derived from [[ISO-639]](https://go.microsoft.com/fwlink/?LinkId=100294) is used.

**data provider**: A known data source that is specific to a target type and that provides data to a collector type.

**data source**: A database, web service, disk, file, or other collection of information from which data is queried or submitted. Supported data sources vary based on application and data provider.

**data type**: A property of a field that defines the kind of data that is stored in the field, or defines the kind of data returned by an expression when the expression is evaluated.

**drawing**: A collection of drawing objects, such as shapes, curves, or WordArt, that are viewed together as a single image.

**embedded image**: An image that is stored within a document rather than being linked to a source file that is outside the document.

**Extensible Application Markup Language (XAML)**: A declarative XML-based language that is used to represent a tree of objects. XAML markup is stored in .xaml files or, for workflow schemas, .xoml files.

**field**: An element or attribute in a data source that can contain data.

**floating-point number**: A number that is represented by a mantissa and an exponent according to a given base. The mantissa is typically a value between "0" and "1". To find the value of a floating-point number, the base is raised to the power of the exponent, and the mantissa is multiplied by the result.

**font**: An object that defines the graphic design, or formatting, of a collection of numbers, symbols, and letters. A font specifies the style (such as bold and strikeout), size, family (a typeface such as Times New Roman), and other qualities to describe how the collection is drawn.

**font family**: A set of fonts that all have common stroke width and serif characteristics. For example, Times Roman and Times Roman Italic are members of the same font family.

**globally unique identifier (GUID)**: A term used interchangeably with universally unique identifier (UUID) in Microsoft protocol technical documents (TDs). Interchanging the usage of these terms does not imply or require a specific algorithm or mechanism to generate the value. Specifically, the use of this term does not imply or require that the algorithms described in [[RFC4122]](https://go.microsoft.com/fwlink/?LinkId=90460) or [[C706]](https://go.microsoft.com/fwlink/?LinkId=89824) must be used for generating the [**GUID**](#gt_f49694cc-c350-462d-ab8e-816f0103c6c1). See also universally unique identifier (UUID).

**header row**: A row in a table, typically the first row, that contains labels for columns in the table.

**hue-saturation-luminance (HSL)**: A color model that defines a color by using three dimensions: hue, the color itself; saturation, the purity of the color; and luminance, the amount of light that is either reflected or absorbed by the color. See also color scheme and [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5).

**language code identifier (LCID)**: A 32-bit number that identifies the user interface human language dialect or variation that is supported by an application or a client computer.

**list**: A container within a SharePoint site that stores list items. A list has a customizable schema that is composed of one or more fields.

**Office data connection (ODC) file**: A file that stores information about a connection to a data source, such as an Access database, worksheet, or text file. This file facilitates data source administration.

**OLE compound file**: A form of structured storage, as described in [[MS-CFB]](%5bMS-CFB%5d.pdf#Section_53989ce47b054f8d829bd08d6148375b). A compound file allows independent storages and streams to exist within a single file.

**OLE DB**: A set of interfaces that are based on the Component Object Model (COM) programming model and expose data from a variety of sources. These interfaces support the amount of Database Management System (DBMS) functionality that is appropriate for a data store and they enable a data store to share data.

**Open Database Connectivity (ODBC)**: A standard software API method for accessing data that is stored in a variety of proprietary personal computer, minicomputer, and mainframe databases. It is an implementation of [[ISO/IEC9075-3:2008]](https://go.microsoft.com/fwlink/?LinkId=223900) and provides extensions to that standard.

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

**Portable Network Graphics (PNG)**: A bitmap graphics file format that uses lossless data compression and supports variable transparency of images (alpha channels) and control of image brightness on different computers (gamma correction). PNG-format files have a .png file name extension.

**primary key**: A field or set of fields that uniquely identifies each record in a table. A primary key cannot contain a null value.

**query**: A formalized instruction to a data source to either extract data or perform a specified action. A query can be in the form of a query expression, a method-based query, or a combination of the two. The data source can be in different forms, such as a relational database, XML document, or in-memory object. See also search query.

**range**: An addressable region that is in a workbook. A range typically consists of zero or more cells and represents a single, contiguous rectangle of cells on a single sheet.

**red-green-blue (RGB)**: A color model that describes color information in terms of the red (R), green (G), and blue (B) intensities in a color.

**reference style**: A system that is used in formulas to specify cells or ranges of cells. A reference style specifies a cell in a two-dimensional table by identifying the row and column that contain that cell or range of cells.

**refresh**: A process that retrieves values from a data source and populates a workbook with those values.

**row**: A collection of columns that contains property values that describe a single item in a set of items that match the restriction specified in a query.

**rule**: A condition or action, or a set of conditions or actions, that performs tasks automatically based on events and values.

**sheet**: A part of an Excel workbook. There are four types of sheets: worksheet, macro sheet, dialog sheet, and chart sheet. Multiple sheets are stored together within a workbook.

**storage**: An element of a compound file that is a unit of containment for one or more storages and streams, analogous to directories in a file system, as described in [MS-CFB].

**stream**: An element of a compound file, as described in [MS-CFB]. A stream contains a sequence of bytes that can be read from or written to by an application, and they can exist only in storages.

**Structured Query Language (SQL)**: A database query and programming language that is widely used for accessing, querying, updating, and managing data in relational database systems.

**Unicode**: A character encoding standard developed by the Unicode Consortium that represents almost all of the written languages of the world. The [**Unicode**](#gt_c305d0ab-8b94-461a-bd76-13b40cb8c4d8) standard [[UNICODE5.0.0/2007]](https://go.microsoft.com/fwlink/?LinkId=154659) provides three forms (UTF-8, UTF-16, and UTF-32) and seven schemes (UTF-8, UTF-16, UTF-16 BE, UTF-16 LE, UTF-32, UTF-32 LE, and UTF-32 BE).

**Unicode code point**: Any value in the Unicode codespace, which is a range of integers from "0" to "10FFFF16". Each code point is a unique positive integer that maps to a specific character.

**Uniform Resource Identifier (URI)**: A string that identifies a resource. The URI is an addressing mechanism defined in Internet Engineering Task Force (IETF) Uniform Resource Identifier (URI): Generic Syntax [[RFC3986]](https://go.microsoft.com/fwlink/?LinkId=90453).

**Uniform Resource Locator (URL)**: A string of characters in a standardized format that identifies a document or resource on the World Wide Web. The format is as specified in [[RFC1738]](https://go.microsoft.com/fwlink/?LinkId=90287).

**view**: See form view (Microsoft InfoPath), list view (SharePoint Products and Technologies), or [**View**](#gt_3f793b0b-9509-4df0-89f9-92f07954beb8) (Microsoft Business Connectivity Services).

**whitespace**: A character that can be found between words, including a space (" "), a carriage return in combination with a line feed (newline), and a tab character.

**workbook**: A container for a collection of [**sheets**](#gt_0b356926-f9cb-4dc2-a859-71441d62503d).

**zero-based index**: An index in which the first item has an index of "0" (zero).

**zoom level**: The degree to which a portion of an image, document, or other screen object is made to appear closer or farther away relative to its default appearance. This value is usually expressed as a percentage of the default appearance.

**MAY, SHOULD, MUST, SHOULD NOT, MUST NOT:** These terms (in all caps) are used as defined in [[RFC2119]](https://go.microsoft.com/fwlink/?LinkId=90317). All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

## References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the [Errata](https://go.microsoft.com/fwlink/?linkid=850906).

### 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](mailto:dochelp@microsoft.com). We will assist you in finding the relevant information.

[IEEE754] IEEE, "IEEE Standard for Binary Floating-Point Arithmetic", IEEE 754-1985, October 1985, [http://ieeexplore.ieee.org/servlet/opac?punumber=2355](https://go.microsoft.com/fwlink/?LinkId=89903)

[ISO/IEC-14496-22] International Organization for Standardization, "Information technology -- Coding of audio-visual objects -- Part 22: Open Font Format", 2007, [http://www.iso.org/iso/iso\_catalogue/catalogue\_tc/catalogue\_detail.htm?csnumber=43466](https://go.microsoft.com/fwlink/?LinkId=154475)

[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](https://go.microsoft.com/fwlink/?LinkId=252374)

[ISO/IEC29500-2:2011] ISO/IEC, "Information technology -- Document description and processing languages -- Office Open XML File Formats -- Part 2: Open Packaging Conventions", ISO/IEC 29500-2:2011, 2011, [http://www.iso.org/iso/iso\_catalogue/catalogue\_tc/catalogue\_detail.htm?csnumber=59576](https://go.microsoft.com/fwlink/?LinkId=252375)

[MS-CFB] Microsoft Corporation, "[Compound File Binary File Format](%5bMS-CFB%5d.pdf#Section_53989ce47b054f8d829bd08d6148375b)".

[MS-OAUT] Microsoft Corporation, "[OLE Automation Protocol](%5bMS-OAUT%5d.pdf#Section_bbb05720f72445c78d17f83c3d1a3961)".

[MS-ODBCSTR] Microsoft Corporation, "[ODBC Connection String Structure](%5bMS-ODBCSTR%5d.pdf#Section_13b4e848b36c4b11acced6bf199d5391)".

[MS-SLXV] Microsoft Corporation, "[Silverlight XAML Vocabulary Specification 2008](%5bMS-SLXV%5d.pdf)".

[MS-WPFXV] Microsoft Corporation, "[WPF XAML Vocabulary Specification 2006](%5bMS-WPFXV%5d.pdf)".

[MS-XAML] Microsoft Corporation, "[XAML Object Mapping Specification 2006](%5bMS-XAML%5d.pdf)".

[RFC2083] Boutell, T., et al., "PNG (Portable Network Graphics) Specification Version 1.0", RFC 2083, March 1997, [http://www.ietf.org/rfc/rfc2083.txt](https://go.microsoft.com/fwlink/?LinkId=90313)

[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](https://go.microsoft.com/fwlink/?LinkId=90317)

[RFC4234] Crocker, D., Ed., and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", RFC 4234, October 2005, [http://www.rfc-editor.org/rfc/rfc4234.txt](https://go.microsoft.com/fwlink/?LinkId=90462)

[RFC4627] Crockford, D., "The application/json Media Type for JavaScript Object Notation (JSON)", RFC 4627, July 2006, [http://www.rfc-editor.org/rfc/rfc4627.txt](https://go.microsoft.com/fwlink/?LinkId=140879)

[XMLSCHEMA1] Thompson, H., Beech, D., Maloney, M., and Mendelsohn, N., Eds., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, [http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/](https://go.microsoft.com/fwlink/?LinkId=90608)

[XMLSCHEMA2] Biron, P.V., Ed. and Malhotra, A., Ed., "XML Schema Part 2: Datatypes", W3C Recommendation, May 2001, [http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/](https://go.microsoft.com/fwlink/?LinkId=90610)

### Informative References

[HTML] World Wide Web Consortium, "HTML 4.01 Specification", W3C Recommendation, December 1999, [http://www.w3.org/TR/html4/](https://go.microsoft.com/fwlink/?LinkId=89880)

[MSDN-CompareOptions] Microsoft Corporation, "CompareOptions Enum", [https://docs.microsoft.com/en-us/dotnet/api/system.globalization.compareoptions](https://go.microsoft.com/fwlink/?LinkId=180505)

[MSDN-ENCLOC] Microsoft Corporation, "Encoding and Localization", .NET Framework Developer's Guide, [http://msdn.microsoft.com/en-us/library/h6270d0z.aspx](https://go.microsoft.com/fwlink/?LinkId=153669)

[MSDN-FormattingTypes] Microsoft Corporation, "Formatting Types", .NET Developer's Framework, [http://msdn.microsoft.com/en-us/library/fbxft59x.aspx](https://go.microsoft.com/fwlink/?LinkId=153833)

[MSDN-OLEDB] Microsoft Corporation, "Microsoft OLE DB", [https://docs.microsoft.com/en-us/previous-versions/windows/desktop/ms722784(v=vs.85)](https://go.microsoft.com/fwlink/?LinkId=90054)

[MSDN-ToDouble] Microsoft Corporation, "Convert.ToDouble Method", .NET Framework Class Library, [http://msdn.microsoft.com/en-us/library/system.convert.todouble.aspx](https://go.microsoft.com/fwlink/?LinkId=153666)

## Overview

This document describes data contained in a [compound file](#Section_71416c996b32472eb1b85f46497e2df2). There are several [streams](#Section_1fc77f54c9d74549aca5a7bed511c016) and [**storages**](#gt_63525252-3f66-4459-9b64-5b77f57b2f84) within the [File Structure](#Section_3328b64021d548c9a79702e72c6b0f13), each using a different persistence format for its data. A [**stream**](#gt_f3529cd8-50da-4f36-aa0b-66af455edbb6) named "VisioServerData" contains a ZIP archive that stores all the information needed to describe a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25).

A [ShapeGraphic XML Part](#Section_91b39933d4194cff88431368a42a2258) in the ZIP archive describes the graphical elements displayed in the Web drawing. These graphical elements are expressed in Extensible Application Markup Language (XAML). Graphical elements can be static or dynamic. Dynamic graphical elements, also called [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075), have visual properties that are [bound](#Section_b9b194892da24d77a8a51c99bb866fe1) to data in a [**data source**](#gt_e091613c-6901-4874-b9b2-27273ead1075), and the appearance of these elements changes as data in the data source [refreshes](#Section_07b8ea46bb6a4b2b92853c88ad3f1f86).

A collection of [XML Parts](#Section_e353d699f4bc454eaf7b20193355ee32) in the ZIP archive describes the Web drawing's [data connections](#Section_28d3d364de644cada48144231aed4d7b), information about its [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) and the [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) logic its dynamic graphical elements. These [parts](#Section_66eb42ec5ca8459696291233020c701f) are expressed as XML. One of these parts, the [DataGraphic XML Part](#Section_1fb58579229c4f8b9a7b7b606ee56810), describes recalculated properties that are to be evaluated after the data in the Web drawing has been refreshed from data sources. In particular, a grammar for [formula evaluation](#Section_81f0fbe77a314d34b52f5fa9c9380c33) exists to describe how changes in the data are translated into changes in properties of graphical elements in the Web drawing. See [Formula Evaluation and Shape Property Recalc](#Section_04483d570a174b32be7b63c671d7d821)ulation for a detailed explanation of this grammar.

Additional items in the ZIP archive contain information for the [**fonts**](#gt_f8aa3f46-99d1-49bb-858f-b4bfa546c1c2) and images used in the Web drawing. See [Fonts](#Section_6b505bc0615e455f9af5cbf71b0e401e) and [Images](#Section_4cb2e2621f8d44a6b20f0be3a3fb0179) for more information.

## Relationship to Protocols and Other Structures

This file format specifies the [streams](#Section_1fc77f54c9d74549aca5a7bed511c016) and [**storages**](#gt_63525252-3f66-4459-9b64-5b77f57b2f84) in an [**OLE compound file**](#gt_b39d4d3a-2d16-4bab-b9fd-ba8e1f16e6a4) that are required to render a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). For details about OLE compound files see [[MS-CFB]](%5bMS-CFB%5d.pdf#Section_53989ce47b054f8d829bd08d6148375b).

## Applicability Statement

This document specifies a persistence format for [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25) content, which can include [drawing pages](#Section_ebba8748c5a54ac7a4191c6a70307640), [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b), [**fonts**](#gt_f8aa3f46-99d1-49bb-858f-b4bfa546c1c2), images, [data connections](#Section_28d3d364de644cada48144231aed4d7b), and recalculation information, as described in Section 2.2.1. The persistence format is applicable when the document content is graphical in nature.

This persistence format is applicable for use as a stand-alone document.

This persistence format provides interoperability with applications that create or read documents conforming to this structure.

## Versioning and Localization

This document covers versioning and localization issues in the following areas:

* Document version: App XML Part, section [2.4.2](#Section_2c93e85a56504f499b49db4ecd19f55e)
* Default document language: App XML Part, section 2.4.2
* Local overrides to document language are specified in attributes, properties and function arguments in individual sections of this document.

## Vendor-Extensible Fields

This persistence format can be extended by storing information in [streams](#Section_1fc77f54c9d74549aca5a7bed511c016) and [**storages**](#gt_63525252-3f66-4459-9b64-5b77f57b2f84) that are not specified in Section 2. Implementations are not required to preserve or remove additional [**streams**](#gt_f3529cd8-50da-4f36-aa0b-66af455edbb6) or storages when modifying an existing document.

# Structures

This section specifies the overall structure of a file that conforms to this specification.

## File Structure Overview

A file of the type specified by this specification MUST be a [compound file](#Section_71416c996b32472eb1b85f46497e2df2).

A compound file contains [**storages**](#gt_63525252-3f66-4459-9b64-5b77f57b2f84) and [streams](#Section_1fc77f54c9d74549aca5a7bed511c016). A stream that contains a ZIP [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499) is used to persist the information necessary to represent fully a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). This package contains a collection of [parts](#Section_66eb42ec5ca8459696291233020c701f) that are used to persist data in XML or standard binary formats and specify various aspects of the Web drawing as well as the structure of the package.

The following figure shows a possible implementation of a file specified by this specification.

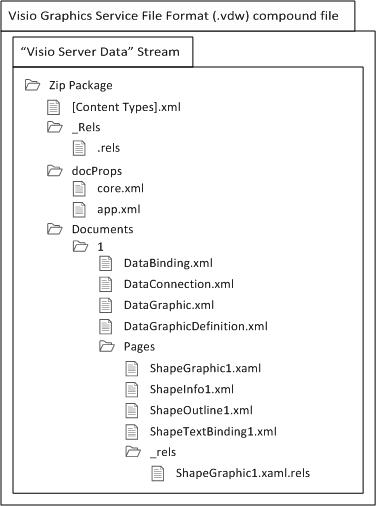


Figure 1: Possible Implementation of a File

### Compound File

A file of the type specified by this document MUST be an [**OLE compound file**](#gt_b39d4d3a-2d16-4bab-b9fd-ba8e1f16e6a4) as specified in [[MS-CFB]](%5bMS-CFB%5d.pdf#Section_53989ce47b054f8d829bd08d6148375b).

### Streams

A file of the type specified by this document consists of [**streams**](#gt_f3529cd8-50da-4f36-aa0b-66af455edbb6) as specified in [[MS-CFB]](%5bMS-CFB%5d.pdf#Section_53989ce47b054f8d829bd08d6148375b).

A file of the type specified by this document MUST contain exactly one stream with the name "VisioServerData". This stream completely specifies the information necessary to represent a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). Other streams that exist in the file are unused and MUST be ignored when rendering aWeb drawing.

### Package

The "VisioServerData" [stream](#Section_1fc77f54c9d74549aca5a7bed511c016) MUST contain exactly one package that is a ZIP archive that conforms to the Open Packaging Conventions as specified in [[ISO/IEC29500-2:2011]](https://go.microsoft.com/fwlink/?LinkId=252375), the further packaging restrictions specified in [[ISO/IEC29500-1:2011]](https://go.microsoft.com/fwlink/?LinkId=252374) Section 9, and this specification.

### Parts

A [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499) is composed of multiple parts as specified in [[ISO/IEC29500-2:2011]](https://go.microsoft.com/fwlink/?LinkId=252375) Section 9.1. Each part has an associated content type that specifies the format it is persisted in. Each part can also be the target or the source of a connection between two parts called a [relationship](#Section_d7daa0e06e22406f96d9655bc5f29e78), as specified in [ISO/IEC29500-2:2011] Section 9.3.

The valid parts, content types, and required and optional relationships between all parts in this package are specified in [Part Enumeration](#Section_24caf49c97804bd0827c61d9cc647ab3).

### Relationships

A relationship specifies a connection between a source and a target resource as specified in [[ISO/IEC29500-1:2011]](https://go.microsoft.com/fwlink/?LinkId=252374) Section 8.3. Relationship identifiers are used in binary and XML [part](#Section_66eb42ec5ca8459696291233020c701f) content to reference unique relationship elements in relationship parts that in turn target other resources. There are several different types of relationships:

* A [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499) relationship is a relationship where the target is a part and the source is the package as a whole.
* A part-to-part relationship is a relationship where the target is a part and the source is a part in the package.
* An explicit relationship is a relationship where a resource is referenced from the contents of a source part by referencing the ID attribute value of a relationship element.
* An implicit relationship is a relationship that is not explicit.
* An internal relationship is a relationship where the target is a part in the package.
* An external relationship is a relationship where the target is an external resource not in the package.

### Part Enumeration

The "VisioServerData" [stream](#Section_1fc77f54c9d74549aca5a7bed511c016) contains the following ZIP [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499) [parts](#Section_66eb42ec5ca8459696291233020c701f) and [relationships](#Section_d7daa0e06e22406f96d9655bc5f29e78):

| Part Name | Relationship target of... |
| --- | --- |
| [Core](#Section_ef5dbca1b87c456cbc1673fafb319612) | package |
| [App](#Section_a9ba94e7522542ce9a4b8552b7d3f6e7) | package |
| [Fonts](#Section_6b505bc0615e455f9af5cbf71b0e401e) | package |
| [Images](#Section_4cb2e2621f8d44a6b20f0be3a3fb0179) | [ShapeGraphic](#Section_91b39933d4194cff88431368a42a2258) |
| [DataBinding](#Section_ddb7bd92bfe64998916e8dfb99140798) | package |
| [DataConnection](#Section_83bf3bed01f44b4fb096b7f19c2aced8) | package |
| [DataGraphic](#Section_1fb58579229c4f8b9a7b7b606ee56810) | package |
| [DataGraphicDefinition](#Section_52fd15daf90d485f9e6123ed67a3b063) | package |
| [ShapeInfo](#Section_7ab8617e1a9f4fcd909c52b0d983de4b) | ShapeGraphic |
| [ShapeOutline](#Section_95cae30c8856445bb58a1de22f34c6b3) | ShapeGraphic |
| [ShapeTextBinding](#Section_f40c633a99df45dcb12a9049fca1fb6b) | ShapeGraphic |
| [Rels](#Section_f996c29c681640c58fd62e2ab94bac40) | package |
| ShapeGraphic | package |
| [Content Type](#Section_6f4e2abfa782498d99d07b262e3f53f5) | package |

#### ShapeGraphic

| Content Type: | application/xaml+xml |
| --- | --- |
| Source Relationship: | http://schemas.openxmlformats.org/officeDocument/2006/relationships/custom-properties |

An instance of this [part](#Section_66eb42ec5ca8459696291233020c701f) specifies a vector description of a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) and all the [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) contained within it.The syntax of the structures contained in this part is specifiedin the [ShapeGraphic XML Part](#Section_d86834c499d248e2b13ab2c8f016eca0).

One ShapeGraphic part MUST exist in the [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499) for each unique value of the **ID** attribute across all the CT\_PageMetaData elements contained in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e). The name of each part of this type MUST be of the form "ShapeGraphic*N*.xaml", where N is the value of the associated **ID** attribute.

Each ShapeGraphic part MUST be the target of a package relationship.

A ShapeGraphic part is permitted to have implicit relationships to the following parts:

* [ShapeTextBinding](#Section_f40c633a99df45dcb12a9049fca1fb6b)
* [ShapeOutline](#Section_95cae30c8856445bb58a1de22f34c6b3)
* [ShapeInfo](#Section_7ab8617e1a9f4fcd909c52b0d983de4b)
* [Images](#Section_4cb2e2621f8d44a6b20f0be3a3fb0179)

#### Fonts

| Content Type: | application/vnd.openxmlformats-officedocument.obfuscatedFont |
| --- | --- |
| Source Relationship: | http://schemas.openxmlformats.org/officeDocument/2006/relationships/font |

An instance of this [part](#Section_66eb42ec5ca8459696291233020c701f) type specifies a [**font**](#gt_f8aa3f46-99d1-49bb-858f-b4bfa546c1c2) resource used in rendering a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). Each part of this type is a binary font file that conforms to the Open Type Font Format as specified in [[ISO/IEC-14496-22]](https://go.microsoft.com/fwlink/?LinkId=154475). The part MUST be modified from the original Open Type Font Format in the following way:

The part name MUST of the form "B03B02B01B00-B11B10-B21B20-B30B31-B32B33B34B35B36B37.odttf" where each "Bxy" is a two digits hexadecimal representation of a byte taken from a [**GUID**](#gt_f49694cc-c350-462d-ab8e-816f0103c6c1).

The part data MUST satisfy the following requirements:

* The first 16 bytes of the part data MUST be the result of a binary XOR operation of the corresponding original Open Type Font Format byte with B37, B36, B35, B34, B33, B32, B31, B30, B20, B21, B10, B11, B00, B01, B02, and B03.
* The next 16 bytes of the part data MUST be the result of a binary XOR operation of the corresponding original Open Type Font Format byte with the same sequence of bytes extracted from the part name.

One Fonts part MUST exist in the [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499) for each unique value of the **FontURI** attribute across all **Glyph** elements contained within all [ShapeGraphic](#Section_91b39933d4194cff88431368a42a2258) parts, as specified in [[MS-SLXV]](%5bMS-SLXV%5d.pdf) section 1.22.31.1.4, that does not contain the string "CommonFonts/". The name of the Fonts part MUST be equal to the value of **FontURI**.

All Fonts parts MUST be the target of a package relationship. A Fonts part MUST NOT have implicit or explicit relationships to any other part specified in this specification.

#### Images

| Content Type: | image/png |
| --- | --- |
| Source Relationship: | http://schemas.openxmlformats.org/officeDocument/2006/relationships/image |

An instance of this [part](#Section_66eb42ec5ca8459696291233020c701f) type specifies an image resource used in rendering a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). Each part of this type is an image file that MUST support the [**Portable Network Graphics (PNG)**](#gt_212c87ce-40ba-4311-be2a-494f1a116604) format specified in [[RFC2083]](https://go.microsoft.com/fwlink/?LinkId=90313).

One Images part MUST exist in the [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499) for each unique value of the following attributes of elements contained within all [ShapeGraphic](#Section_91b39933d4194cff88431368a42a2258) parts:

* All **Source** attributes of the Image elements as specified in [[MS-SLXV]](%5bMS-SLXV%5d.pdf) section 1.22.31.1.5. The name of the Images part MUST be equal to the value of **Source**.
* All **ImageSource** attributes of the ImageBrush elements as specified in [MS-SLXV] section 1.22.2.3.1. The name of the Images part MUST be equal to the value of **ImageSource**.

All Images parts MUST be the target of an implicit relationship from the related ShapeGraphic part. An Images part MUST NOT have implicit or explicit relationships to any other part specified in this specification.

#### App

This [part](#Section_66eb42ec5ca8459696291233020c701f) is specified in [[ISO/IEC29500-1:2011]](https://go.microsoft.com/fwlink/?LinkId=252374) Section 15.2.12.3.The syntax of the structures contained in this part is specified in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e).

#### Core

This [part](#Section_66eb42ec5ca8459696291233020c701f) is specified in [[ISO/IEC29500-1:2011]](https://go.microsoft.com/fwlink/?LinkId=252374) Section 15.2.12.1. The syntax of the structures contained in this part is specified in the [Core XML Part](#Section_f1f650910bf743349e3327acea61c645).

#### DataBinding

| Content Type: | application/xml |
| --- | --- |
| Source Relationship: | http://schemas.openxmlformats.org/officeDocument/2006/relationships/custom-properties |

An instance of this [part](#Section_66eb42ec5ca8459696291233020c701f) type specifies the information that is used to create [data bindings](#Section_b9b194892da24d77a8a51c99bb866fe1) between a [recordsets](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) and [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) in [drawing pages](#Section_ebba8748c5a54ac7a4191c6a70307640) in a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). The syntax of the structures contained in this part is specified in the [DataBinding XML Part](#Section_311514cd8ef444c9ab42562671155333).

A [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499) MUST contain exactly one DataBinding part. The DataBinding part MUST have the name "DataBinding.xml". The DataBinding part MUST be the target of a package relationship.

The DataBinding part MUST NOT have implicit or explicit relationships to any other part specified in this specification.

#### DataConnection

| Content Type: | application/xml |
| --- | --- |
| Source Relationship: | http://schemas.openxmlformats.org/officeDocument/2006/relationships/custom-properties |

An instance of this [part](#Section_66eb42ec5ca8459696291233020c701f) type specifies the [data connection](#Section_28d3d364de644cada48144231aed4d7b) information needed to [**query**](#gt_37fbc661-f744-48fa-9d8e-f34513cab9c2) [**data sources**](#gt_e091613c-6901-4874-b9b2-27273ead1075) and [refresh](#Section_8aae518c4bca49968a84541ff996d01a) the [recordsets](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) referenced by a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). The syntax of the structures contained in this part is specified in the [DataConnection XML Part](#Section_c489eefb7a014941bc37a4d332e66dc8).

A [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499) MUST contain exactly one DataConnection part. The DataConnection part MUST have the name "DataConnection.xml". The DataConnection part MUST be the target of a package relationship.

The DataConnection part MUST NOT have implicit or explicit relationships to any other part specified in this specification.

#### DataGraphicDefinition

| Content Type: | application/xml |
| --- | --- |
| Source Relationship: | http://schemas.openxmlformats.org/officeDocument/2006/relationships/custom-properties |

An instance of this [part](#Section_66eb42ec5ca8459696291233020c701f) type specifies information used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) the [image element](#Section_385a6cb3903347b3acca493e5b363057) [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) in a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). The syntax of the structures contained in this part is specified in the [DataGraphicDefinition XML Part](#Section_43b985472c2f40eaafc98da28ba27fbb).

A [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499) MUST contain exactly one DataGraphicDefinition part. The DataGraphicDefinition part MUST have the name "DataGraphicDefinition.xml". The DataGraphicDefinition part MUST be the target of a package relationship.

The DataGraphicDefinition part MUST NOT have implicit or explicit relationships to any other part specified in this specification.

#### DataGraphic

| Content Type: | application/xml |
| --- | --- |
| Source Relationship: | http://schemas.openxmlformats.org/officeDocument/2006/relationships/custom-properties |

An instance of this [part](#Section_66eb42ec5ca8459696291233020c701f) type specifies information used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) all [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) in a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). The syntax of the structures contained in this part is specified in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087).

A [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499) MUST contain exactly one DataGraphic part. The DataGraphic part MUST have the name "DataGraphic.xml". The DataGraphic part MUST be the target of a package relationship.

The DataGraphic part MUST NOT have implicit or explicit relationships to any other part specified in this specification.

#### ShapeInfo

| Content Type: | application/xml |
| --- | --- |
| Source Relationship: | http://schemas.openxmlformats.org/officeDocument/2006/relationships/custom-properties |

An instance of this [part](#Section_66eb42ec5ca8459696291233020c701f) type specifies information about the [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) on a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) in a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). The syntax of the structures contained in this part is specified in the [ShapeInfo XML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac).

One ShapeInfo part MUST exist for each [ShapeGraphic](#Section_91b39933d4194cff88431368a42a2258) part in the [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499). The ShapeInfo part MUST be the target of an implicit relationship from the related ShapeGraphic part. The name of each ShapeInfo part MUST be of the form "ShapeInfo*N*.xml", where N is the value of the **ID** attribute of the CT\_PageMetaData element contained in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) that also corresponds to the ShapeGraphic part.

The ShapeInfo part MUST NOT have implicit or explicit relationships to any other part specified in this specification.

#### ShapeTextBinding

| Content Type: | application/xml |
| --- | --- |
| Source Relationship: | http://schemas.openxmlformats.org/officeDocument/2006/relationships/custom-properties |

An instance of this [part](#Section_66eb42ec5ca8459696291233020c701f) type specifies information used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) the [text element](#Section_b061c2bde75d48edb51047479b51c6ec) [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) in a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). The syntax of the structures contained in this part is specified in the [ShapeTextBinding XML Part](#Section_2e21d026e11d46ee860a75c6b75bd84a).

One ShapeTextBinding part MUST exist for each [ShapeGraphic](#Section_91b39933d4194cff88431368a42a2258) part in the [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499). The ShapeTextBinding part MUST be the target of an implicit relationship from the related ShapeGraphic part. The name of each ShapeTextBinding part MUST be of the form "ShapeTextBinding*N*.xml", where N is the value of the **ID** attribute of the CT\_PageMetaData element contained in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) that also corresponds to the ShapeGraphic part.

The ShapeTextBinding part MUST NOT have implicit or explicit relationships to any other part specified in this specification.

#### ShapeOutline

| Content Type: | application/xml |
| --- | --- |
| Source Relationship: | http://schemas.openxmlformats.org/officeDocument/2006/relationships/custom-properties |

An instance of this [part](#Section_66eb42ec5ca8459696291233020c701f) type specifies the geometric outline information of the [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) in a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) in a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25). The syntax of the structures contained in this part is specified in the [ShapeOutline XML Part](#Section_e81d9654a5d846cfbba59b380b3a8ade).

One ShapeOutline part MUST exist for each [ShapeGraphic](#Section_91b39933d4194cff88431368a42a2258) part in the [package](#Section_ff4eda2d09694c3c8a4e8afa1e16d499). The ShapeOutline part MUST be the target of an implicit relationship from the related ShapeGraphic part. The name of each ShapeOutline part MUST be of the form "ShapeOutline*N*.xml", where N is the value of the **ID** attribute of the CT\_PageMetaData element contained in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) that also corresponds to the ShapeGraphic part.

The ShapeOutline part MUST NOT have implicit or explicit relationships to any other part specified by this specification.

#### Rels

This [part](#Section_66eb42ec5ca8459696291233020c701f) and its syntax are specified in [[ISO/IEC29500-2:2011]](https://go.microsoft.com/fwlink/?LinkId=252375) Section 9.3.

#### ContentType

This [part](#Section_66eb42ec5ca8459696291233020c701f) and its syntax are specified in [[ISO/IEC29500-2:2011]](https://go.microsoft.com/fwlink/?LinkId=252375) Section 10.1.2.

## Conceptual Overview

This section specifies how higher-level features of the file format are represented by combinations of [parts](#Section_66eb42ec5ca8459696291233020c701f).

### Web Drawing

A Web drawing is a collection of [drawing pages](#Section_ebba8748c5a54ac7a4191c6a70307640), [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b), [**fonts**](#gt_f8aa3f46-99d1-49bb-858f-b4bfa546c1c2), images, [data connections](#Section_28d3d364de644cada48144231aed4d7b), and recalculation information required to render the [**drawing**](#gt_4b4d1a42-c4a2-464d-ad16-d53b5c493fd3) in a web browser.

A Web drawing is specified by the "VisioServerData" [stream](#Section_1fc77f54c9d74549aca5a7bed511c016) within the file format as specified in the [File Structure Overview](#Section_3328b64021d548c9a79702e72c6b0f13).

Properties of the Web drawing such as the name of the application that created it, the list of drawing pages it contains, and its description are specified by the child elements of the CT\_Properties element in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) and by the CT\_coreProperties element in the [Core XML Part](#Section_f1f650910bf743349e3327acea61c645).

For examples of the contents of the "VisioServerData" stream for various Web drawings, see [Structure Examples](#Section_eebab70c52c94ee1b97634154553bcdc).

### Drawing Page

A drawing page is a collection of [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) that are viewed together.

The graphical information necessary to render a drawing page is specified by the [ShapeGraphic XML Part](#Section_d86834c499d248e2b13ab2c8f016eca0). Additional information about shapes in the drawing page and about the properties of the drawing page itself is specified by a [ShapeInfo XML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac).

A drawing page is uniquely identified by **ID** and **Name** attributes in a CT\_PageMetaData element in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e). The following elements in other [parts](#Section_66eb42ec5ca8459696291233020c701f) of the document have attributes that are equal to **ID** or **Name** and specify supplementary information about the drawing page:

* A [CT\_Page](#Section_020589d88a1f420c97805dd19dac1cf7) element in the ShapeInfo XML Part has a **Name** attribute that is equal to the **Name** attribute of CT\_PageMetaData and specifies information about the shapes in the drawing page and the default [**view**](#gt_3f793b0b-9509-4df0-89f9-92f07954beb8) for the drawing page.
* A [CT\_PageInfo](#Section_3eddc14246c1484497128d5790bdb49d) element in the ShapeInfo XML Part has a **Name** attribute that is equal to the **Name** attribute of CT\_PageMetaData.
* A [CT\_DataRecordset](#Section_09771dd2110c41e9ad2b609514a2c79c) element in the [DataConnection XML Part](#Section_c489eefb7a014941bc37a4d332e66dc8) has a **Pages** attribute that specifies a comma-separated list of indices of drawing pages. The index of each drawing page in this list is equal to the **ID** attribute of one of the **PageMetaData** child elements of CT\_PagesMetaData in the ShapeInfo XML Part. A CT\_DataRecordset element specifies a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) linked to shapes in these drawing pages.
* A [CT\_DataGraphics\_Page](#Section_1656fc08de3a4c088c7a5ed89568c0b6) element in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087) has a **PageName** attribute that is equal to the **Name** attribute of CT\_PageMetaData and specifies information used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) the [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) in the drawing page.

A drawing page has display characteristics such as [**zoom level**](#gt_bfb33471-a018-422b-bc63-177c8bc1831f) and offset position of the point on the drawing page that is centered in the view are specified by the attributes of the CT\_Page element in the ShapeInfo XML Part.

A drawing page can contain text, which can depend on various [**fonts**](#gt_f8aa3f46-99d1-49bb-858f-b4bfa546c1c2). Each font used in rendering text in a drawing page is specified by a [Fonts](#Section_6b505bc0615e455f9af5cbf71b0e401e) part and is referenced as specified in [Fonts](#Section_7d86d940fa714c5284fc572f44845f1e).

A drawing page can contain [**embedded images**](#gt_c2dd082b-6519-49a2-99e9-77b671621249). Each image used in a drawing page is specified by an [Images](#Section_4cb2e2621f8d44a6b20f0be3a3fb0179) part and is referenced as specified in [Images](#Section_2183bf65abb049839cc9c5a99409c721).

### Shape

A shape is a collection of geometry, text, images, [hyperlinks](#Section_1fc5829872d4478d9d86b282342940cc), and [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) in a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640).

The set of shapes in a drawing page is specified by the [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0) child elements of the [CT\_Page](#Section_020589d88a1f420c97805dd19dac1cf7) element in the [ShapeInfo XML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac).

The geometry, text, formatting, positioning and image information for a shape is specified by the [ShapeGraphic XML Part](#Section_d86834c499d248e2b13ab2c8f016eca0). Hyperlinks, shape data, and other shape properties are specified by the CT\_ShapeInfo child element of the CT\_Page element in the ShapeInfo XML Part. The following sections specify additional information about shapes.

#### Shape Identification

A [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) is uniquely identified by the **Name** attribute in a [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0) element in the [ShapeInfoXML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac). The following elements in other [parts](#Section_66eb42ec5ca8459696291233020c701f) of the document have attributes that are equal to **Name** and specify supplementary information about this shape:

* A **Canvas** graphical element in the [ShapeGraphic XML Part](#Section_d86834c499d248e2b13ab2c8f016eca0) has a **Name** attribute that is equal to the **Name** attribute of CT\_ShapeInfo and specifies the visual information about the shape.
* A [CT\_DataGraphics\_Shape](#Section_24cbea2f9ad1441a81f41db90fb3dd52) element in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087) has a **ShapeName** attribute that is equal to the **Name** attribute of CT\_ShapeInfo and specifies information used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) the [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) associated with the shape.

#### Shape Visualization

The graphical information necessary to render a [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) is specified by the [ShapeGraphic XML Part](#Section_d86834c499d248e2b13ab2c8f016eca0) using Silverlight [**XAML**](#gt_891018ae-9381-4d29-82d7-20d6c8537472) and MUST conform to the restrictions specified in [XAML Shapes](#Section_9cb62f8d7fc2459180559f852f9b7db9).

Shapes that have no graphical or textual attributes bound to [**data sources**](#gt_e091613c-6901-4874-b9b2-27273ead1075) are called static shapes. The display characteristics of static shapes will never change during the lifetime of the [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640).

Shapes that have graphical attributes bound to data sources are called recalculated shapes. When the data in the data source changes, the graphical attributes of these shapes are recalculated and the visual appearance of the shapes change. Additional restrictions on the XAML that specifies recalculated shapes are specified in [XAML Recalculated Shapes](#Section_0407e80e0a0f4cd7b0d1de23a6a7b3ad).

For more information about binding data to shapes, see [Data Binding to Web Drawing Elements](#Section_30d7029b15b049e5a5ad1cee2e87a2c8).

#### Shape Selection

Interactivity with the web drawing, including shape selection, is enabled with the use of geometric outlines of [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b). The geometric outline of a shape is specified by the **Canvas** element in the [ShapeGraphic XML Part](#Section_d86834c499d248e2b13ab2c8f016eca0) that corresponds to the shape and is also specified by a [CT\_Shape](#Section_6defd531916c4391bbb80bc71d0c0c5c) element in the [ShapeOutline XML Part](#Section_e81d9654a5d846cfbba59b380b3a8ade). The value of the **Name** attribute of the CT\_Shape element MUST be equal to the value of the **Name** attribute of the [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0) element that corresponds to the shape.

Additional geometric information is specified to facilitate the highlighting of a selected shape. The selected shape is visually indicated by highlighting its [**bounding rectangle**](#gt_e349a155-0fb9-4ba1-abd4-e363e21abcd1). The bounding rectangle of a shape is specified by the **Layout** attribute of the corresponding CT\_ShapeInfo element in the [ShapeInfo XML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac).

#### Shape Hyperlinks

A [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) can have one or more hyperlinks associated with it. Hyperlinks can point to destinations outside the [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25) or to [drawing pages](#Section_ebba8748c5a54ac7a4191c6a70307640) and shapes within the Web drawing. The set of hyperlinks associated with a shape is specified by the [CT\_Hyperlinks](#Section_155b1e53317a4af1b82ca10f8e346210) child element of the corresponding [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0) element in the [ShapeInfo XML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac).

The properties of each hyperlink, such as its name, [**Uniform Resource Identifier (URI)**](#gt_e18af8e8-01d7-4f91-8a1e-0fb21b191f95), and description, are specified by the attributes of the [CT\_Hyperlink](#Section_0cd8a3b16777487ea09dc7f31460d3e0) child element of the CT\_Hyperlinks element in the ShapeInfo XML Part.

#### Shape Data

A [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) can have data associated with it which provides information about its meaning. A shape’s data is stored as a set of shape data items. The set of shape data items associated with a shape is specified by the [CT\_ShapeDataItems](#Section_67db1c6345344b4e86ae8bc52c53deec) child element of the corresponding [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0) element in the [ShapeInfo XML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac).

The properties of each shape data item, such as its label, value, and [**data type**](#gt_0d327bc0-a461-4723-8beb-d935fdab4895), are specified by the attributes of the [CT\_ShapeData](#Section_a90cbb53af5347ae9da20afddd00cf2f) child element of the CT\_ShapeDataItems element in the ShapeInfo XML Part. The **BindingID** attribute of this element is an identifier that specifies the [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) of data in a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) that is bound to this shape data item, as specified in the [CT\_Binding](#Section_5ad7ddf78438400abe979591e0cca24a) element of the [DataBinding XML Part](#Section_311514cd8ef444c9ab42562671155333). If the **BindingID** attribute is not present, then the shape data item is not bound to data in a [**data source**](#gt_e091613c-6901-4874-b9b2-27273ead1075).

### Data Connectivity and Refresh

This section describes how [**data sources**](#gt_e091613c-6901-4874-b9b2-27273ead1075) can be referenced, queried and connected to from within a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25).

#### Data Connections

A [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25) can be linked to data from databases and other [**data sources**](#gt_e091613c-6901-4874-b9b2-27273ead1075) which can affect various attributes of the Web drawing including its visual appearance. A relationship to such data sources is called a data connection.

A data connection contains properties that specify how the application connects to and [**queries**](#gt_37fbc661-f744-48fa-9d8e-f34513cab9c2) a data source, including the type of [**data provider**](#gt_33fa4cdc-ae58-4a6c-8111-31377e1d292e) (for example, [**OLE DB**](#gt_333f4fb1-4882-48df-bce6-f9961b408f31) or [**ODBC**](#gt_7883fa02-8dc0-4154-894f-fe3a7bff153e)) required to access a data source, the name of the server on which the data source is hosted, security information to access the data source, and a query to execute on the server.

Data connections are stored either completely inside or partly outside of the document. If the **FileName** attribute of a [CT\_DataConnection](#Section_7e5607b6320948fa9f6dc172500e1dd7) element in the [DataConnection XML Part](#Section_c489eefb7a014941bc37a4d332e66dc8) is not present, a data connection is completely specified by the following pair of elements:

* The containing CT\_DataConnection.
* The [CT\_DataRecordset](#Section_09771dd2110c41e9ad2b609514a2c79c) with a **connectionID** attribute that matches the **ID** attribute of the containing CT\_DataConnection.

If the **FileName** attribute is present, then a data connection is specified by the triplet of:

* The containing CT\_DataConnection.
* The CT\_DataRecordset with a **connectionID** attribute that matches the **ID** attribute of the containing CT\_DataConnection.
* The information contained in the [**Office data connection (ODC) file**](#gt_e6fb952a-b6b6-40af-ad1b-ea6cf97da509) found at the path described by the value of the **FileName** attribute.

A Web drawing can contain zero or more data connections, each uniquely identified by the **ID** attribute of the CT\_DataRecordset element. Data connections can be established for the types of data sources listed in the **ConnectionString** attribute of the CT\_DataConnection element.

#### Recordset

A recordset is the data that is returned from a [**data source**](#gt_e091613c-6901-4874-b9b2-27273ead1075), organized into sets of [**rows**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) and [**fields**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc). The properties of this recordset, such as its identifier, the underlying [data connection](#Section_28d3d364de644cada48144231aed4d7b), the number and type of fields each row contains, the row addressing method, and the set of [drawing pages](#Section_ebba8748c5a54ac7a4191c6a70307640) on which the data is used, are specified in a [CT\_DataRecordset](#Section_09771dd2110c41e9ad2b609514a2c79c) element in a [DataConnection XML Part](#Section_c489eefb7a014941bc37a4d332e66dc8).

#### Recordset Refresh

The operation of replacing the contents of a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) with data queried from an underlying [**data source**](#gt_e091613c-6901-4874-b9b2-27273ead1075), using the associated [data connection](#Section_28d3d364de644cada48144231aed4d7b), is called refreshing the recordset.

#### Recordset Row Addressing

There are two mutually exclusive [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) addressing methods that specify how to reference a particular row across [**refresh**](#gt_d947dfed-8e3a-4451-b978-d69a89ae6060) operations: the [row order method](#Section_85d62e47587a4311aeb79b93ba288da6) and the [primary key method](#Section_aebfb9093e8c4dbdb11c988bbd424388).

##### Row Order Method

In the [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) order method, [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) rows are addressed by their position in the recordset, regardless of their contents. This addressing method is specified by the **RowOrder** attribute of the [CT\_DataRecordset](#Section_09771dd2110c41e9ad2b609514a2c79c) element.

##### Primary Key Method

In the **primary key** method, [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) [**rows**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) are addressed by value of their [**primary key**](#gt_e4d8c530-39c1-4fc6-8ccc-8d51a221158d). This is addressing method is specified by the **PrimaryKey** element of the [CT\_DataRecordset](#Section_09771dd2110c41e9ad2b609514a2c79c) element.

### Data Binding to Web Drawing Elements

This section describes how a [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) item can be bound to a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) cell.

#### Data Binding

The association between a particular cell in a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2), specified by a [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) and a [**field**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc), and a [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) item is called a data binding.

A cell in a recordset can be bound to zero or more shape data items. A cell can be bound to only one shape data item per shape. A shape data item can be bound to only one cell.

The cell bound to a shape data item is specified by the **BindingID** and **Name** attributes of the corresponding [CT\_ShapeData](#Section_a90cbb53af5347ae9da20afddd00cf2f) element in the [ShapeInfo XML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac). The **BindingID** attribute references a [CT\_Binding](#Section_5ad7ddf78438400abe979591e0cca24a) element which specifies the row of the recordset. The **Name** attribute references a [CT\_DataColumn](#Section_f5259dad7ba3465092d7ef1d69cc6a70) element which specifies the field of the recordset.

The [CT\_BindingConnection](#Section_7692f89f4a9a4267b1b5ba319675ca62) element that contains this CT\_Binding element referenced by **BindingID** specifies the recordset that contains the cell bound to the shape data item.

When the data in a cell in a recordset changes, the shape data item that it is bound to is updated in the following manner:

* The **Value** attribute of the CT\_ShapeData is updated with the value of the data in the cell. This could involve a [**data type**](#gt_0d327bc0-a461-4723-8beb-d935fdab4895) conversion from the data type of the cell as specified by the **Type** attribute of its corresponding CT\_DataColumn, to the data type of the shape data item as specified by the **Type** attribute of the bound CT\_ShapeData element.
* The **FormattedValue** attribute of the CT\_ShapeData is updated with a string representation of the value of the data in the cell formatted for display. The formatting of the value into a string is done using the following attributes of the CT\_ShapeData element: **Format, LangID, UnitLabel, CalendarID, CurrencyID, ServerAction, Unit and DisplayUnit.**

#### Diagram Update

Updating the vector description of the [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25) from its current state to an updated state based on change in an underlying [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) is called a diagram update.

[Formula expressions](#Section_1639b4e843e14d409531adf5c2fe8bbc) in these Web drawing elements that are dependent on data bound [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) items use the [PtgDataBinding](#Section_7442cb26af7f46ce9bd348656835396c) token to refer to the shape data items. The [data bindings](#Section_b9b194892da24d77a8a51c99bb866fe1) referred to by these shape data items enable the formula expressions to be recalculated from updated values obtained from a recordset.

#### Data Graphics

A set of [**XAML**](#gt_891018ae-9381-4d29-82d7-20d6c8537472) graphical elements that can be updated by [formula expressions](#Section_1639b4e843e14d409531adf5c2fe8bbc) is called a data graphic. The following table describes the types of XAML elements that can be updated and the elements within the various XML [parts](#Section_d7daa0e06e22406f96d9655bc5f29e78) that determine how the XAML elements are updated.

| XAML Elements | XML Elements |
| --- | --- |
| [Sheet Elements](#Section_94a21e97d2934aee9999f20030a1ee56), [Formatting Elements](#Section_b08c0179146b4b2dbdce35c55c7f7737) and [Geometry Elements](#Section_23c25866532c499aa07c9dc558139a9b) | [CT\_DataGraphics\_Shape](#Section_24cbea2f9ad1441a81f41db90fb3dd52) in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087) |
| [Text Elements](#Section_b061c2bde75d48edb51047479b51c6ec) | [CT\_ShapeText](#Section_80e46fba919845399e9b8c1147fccc25) in the [ShapeTextBinding XML part](#Section_2e21d026e11d46ee860a75c6b75bd84a) |
| [Image Elements](#Section_385a6cb3903347b3acca493e5b363057) | CT\_DataGraphics\_Shape in the DataGraphic XML Part |

XAML graphical elements are made dependent on data bound [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) items when the formula expressions specified within the related XML elements contain [PtgDataBinding](#Section_7442cb26af7f46ce9bd348656835396c) tokens. The [data bindings](#Section_b9b194892da24d77a8a51c99bb866fe1) referred to by these shape data items enable the formula expressions to be recalculated from updated values obtained from a recordset.

### Recalculating Shape Properties

This section describes concepts used when recalculating [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) properties.

#### Color Table

The color table specifies a zero-based array of [**red-green-blue (RGB)**](#gt_2c716d3a-e60b-4e52-bbb0-2fdeb298003b) color values for the document.

The color table is specified by the **ColorTable** attribute of the [CT\_DataGraphics](#Section_81e7a61453b0487eacebb684a078cab8) element in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087) of the document.

Valid indexes into the color table MUST be greater than or equal to 0 and less than the size of the color table.

#### Formulas

The following sections describe the concepts and elements of a formula.

##### Formula Expression

A formula expression is a sequence of values and functions that, when evaluated, produce a new value.

A formula expression contains a sequence of [parse tokens](#Section_5ae5d80a9aaf40948a2d92cf8e499589), each of which is an [operand token](#Section_0706e8d56456401c85fab5f6351a4a6e), a [control token](#Section_41b89226e2a7471fb70c6d01003be607), or a [function token](#Section_e78706d7cd8c424bbc3bfc74904871ca).

Formula expressions are stored as strings using Reverse-Polish notation. Reverse-Polish notation is a logical system for the specification of mathematical formulas in which operands are followed by operators.

##### Formula Evaluation

Evaluation of a formula specified in Reverse-Polish notation is usually based around an [evaluation stack](#Section_f70486783f2d445fae706da7e34fe640). Tokens are added to the evaluation stack as they are encountered in the Reverse-Polish notation and are removed as they are used in the evaluation. When expression evaluation is finished, there will be exactly one [parse token](#Section_5ae5d80a9aaf40948a2d92cf8e499589) left on the evaluation stack. This token is the result of the evaluation.

##### Parse Tokens

All tokens are stored as Parse Things (ptgs). The format of these tokens is TokenValue:TokenType. The syntax of TokenValue is described for each token in the [Parse Token Definitions](#Section_ae5070d120d946c7ba3c1ad07ed812c4) section. TokenType specifies the type of the token and MUST be an entry under the ID column in the Parse Token Table in the [Parse Token Table](#Section_1c7c6787214545db854e2a1e1ca24d69) section.

###### Control Tokens

Control tokens are used at runtime to control the [evaluation stack](#Section_f70486783f2d445fae706da7e34fe640). The control tokens are [PtgJmp](#Section_a2fcbad2d4cd4d0bb4338ec960e51755), [PtgJmpF](#Section_81250f6f47d04c3d93cd60e2d8798784), [PtgJmpT](#Section_f605c7795ea249bca765efaf9ae6e1f0), [PtgJmpLabel](#Section_5f545b4b47f44ed38932659537323ff9), [PtgPop](#Section_a5016a0a9e794b6dbd38231239a97710) and [PtgPushTop](#Section_46e71a47bb0d441cb00756eb6182c94a). The tokens PtgJmp, PtgJmpF, PtgJmpT, and PtgJmpLabel do not push values onto the evaluation stack but can pop values from the evaluation stack. The tokens PtgJmp, PtgJmpF, and PtgJmpT specify a jump location which MUST be present in a succeeding PtgJmpLabel token. When a PtgJmp, PtgJmpF, and PtgJmpT token is encountered, the succeeding tokens up to the matching PtgJmpLabel token in the [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc) are not pushed onto the evaluation stack and are not evaluated. Evaluation resumes at the token succeeding that PtgJmpLabel token.

###### Function Tokens

Function tokens represent variable or fixed argument functions. The TokenValue MUST be the name of the function as specified for each function by the ABNF func-name column in the Function Token Table in the [Function Token Table](#Section_a5e026e746284da18c6f1e6823e72c02) section. The TokenType specifies the type of the token and MUST be either [PtgFunc](#Section_8206928cdaaf400daa071a566fe1c88b) or [PtgFuncVar](#Section_e483d78ecd494ba7aa042f843a09f8df). For variable argument functions, the number of arguments is specified by the ABNF.

###### Operand Tokens

Operand tokens represent values that are used by functions. An operand specifies a single value that is persisted in the file as one of the tokens specified in the token group [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0), or as a [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd), [PtgDataBinding](#Section_7442cb26af7f46ce9bd348656835396c) or [PtgRecalcRef](#Section_1877c57f45234887a93a70db1b9d46e5). An operand token represents, and can be converted into, one of the following types of values:

* A string value
* A numeric value
* A [**Boolean**](#gt_1d79d7a7-ba2c-4b34-931c-7ba8057c87b2) value
* A currency value
* A color value
* A date value
* An error value

These conversions translate many different source operand tokens into tokens representing different classes of inputs, such as strings and numeric values that are required by functions. Functions typically operate on the converted tokens but can also refer to elements of the source token. See the [Custom Input Types](#Section_0f6e4ab0c4c44182b2d1ca077f19afd5) for details on common token conversions used by functions.

String Values

A string value is used to represent textual information and is persisted in the file as a [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69). Other tokens can also represent a string value according to the conversion specified in the custom input type, [vString](#Section_fe954fee44b243f1b639ae8ad580be3b).

Numeric Values

A numeric value represents a number with or without units.

A numeric value is persisted in the file as one of the tokens in the token group [vNum](#Section_ad426b3dbea2407c8687202b9d13adad) (except [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6)) or as a [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b). Other token types can also represent numeric values as specified in the following custom input types: [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d), [vFloat](#Section_1d9b3dd5ba94418ba9853237ae1cf224), [vSignedInt](#Section_6e6498757ff54acfa20b54eb9d8cd472), [vSignedLong](#Section_52e79e59b9eb4e8a8a7e92d0cdbdecf8), [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a), [vUnsignedLong](#Section_1bf83d6915fa4b6dad66865028333cae).

Numeric values with units represent measured numbers. The units that are supported include length, angle, duration and typographic units.

All numeric values, with or without units, are expressed and persisted in the file in internal units as described in [Custom Internal Unit Types](#Section_1ef4d381d8284ad094e4227751b7b6d2). The internal unit depends on the type of measurement the numeric value represents.

If the numeric value represents a length or distance measurement, the value that is persisted in the file is expressed as a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981). The operand tokens that represent length or distance measurements are persisted in the file as one of these tokens: [PtgNumCM](#Section_bc2d88fb83a94d9588803acf4c0cdc27), [PtgNumDft](#Section_de132a3cdab04751844609cf15ec3756), [PtgNumF](#Section_09bdcb1c922444c8b9c3ecdaa03e0c6d), [PtgNumFI](#Section_a3917f0c42e94c71b55c70ea3aed6819), [PtgNumI](#Section_ac87036f348249b58cba49659d7598a8), [PtgNumKM](#Section_6dc5f00e5e7945678ac890f98cd40706), [PtgNumM](#Section_000148cf28984a299ade5cc03f395c6c), [PtgNumMI](#Section_e1f97f6e268c4429868c74e9323a7b9e), [PtgNumMM](#Section_5fc97e8093e041d7bcd6ddd2c66aa589), [PtgNumNM](#Section_bf29f1f677664e9e856811466228244f), [PtgNumYards](#Section_7039749a686144d9bcd9da3946769b16).

If the numeric value represents an angle measurement, the value that is persisted in the file is expressed as an [angleInternalUnitNumber](#Section_51d7e07cd65741efa808ebb426864c91). The operand tokens that represent angles are specified in the [vAngle](#Section_8609b9afc5aa4a0e9c37b1f7929ac9b4) custom token grouping.

If the numeric value represents a duration measurement, the value that is persisted in the file is expressed as a [durationInternalUnitNumber](#Section_79862c6e7c6b4af5b33e935edf99db5b). The operand tokens that represent durations are [PtgEDay](#Section_f6690609c80a49ed9fffe0dbcfb262a6), [PtgEHour](#Section_e31838fa14684d478f193c033df45e4c), [PtgEMin](#Section_f58cc7ec8a1246d89edb4b3e00a08ac6), [PtgESec](#Section_01c1da7178fc485c8ee6b48674eeb35b), [PtgEWeek](#Section_4aae0088510448abaf3f8ec63f95fd5f), and [PtgTDurDft](#Section_1200eb57709c4fcc9d20e2bac20fa493).

If the numeric value represents a length measurement used in typography, the value that is persisted in the file is expressed as a [typographicInternalUnitNumber](#Section_9ba74244fcbe42229c3a78fcbd9db05e). The operand tokens that represent typographic measurements are [PtgTypCD](#Section_bf50076676d84a8d900468c81b728b8e), [PtgTypCi](#Section_c5f1913417a7448287105b162359a1fa), [PtgTypDft](#Section_d66b059e323d45739c91ec57ed148b16), [PtgTypDi](#Section_a56c1d526fe9452b9857c97fd84c3825), [PtgTypPi](#Section_f3c4aabfb3bb4c68b5b9f175f9620048), [PtgTypPP](#Section_997b3916e4734908b63936fd05317e60), and [PtgTypPt](#Section_b30a64caae964c76ac0728982f26bb31).

A numeric value that represents a percentage value is persisted in the file as a [PtgNumPct](#Section_ef65a2675fd84ba3a1448fb7698f6551) token.

A numeric value with units persisted in the file as a [PtgPageDft](#Section_fe0d479f9b534257a3d13618b7314aaa) token indicates that the internal units are determined by the default values of the [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640), as specified by PtgPageDft.

If the numeric value has no units, it represents a number and is persisted in the file as a [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0) or PtgUnsWord.

All numeric values can have a dimension. One-dimensional units are used to represent length, angle, duration and typographic measurements, and are represented as numeric values as described earlier. Two-dimensional units are used to represent area measurements and three-dimensional units are used to represent volume measurements. A numeric value that has a dimension greater than one is called a multidimensional number and is persisted in the file as a [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84).

Boolean Values

A value that represents a [**Boolean**](#gt_1d79d7a7-ba2c-4b34-931c-7ba8057c87b2) value is persisted in the file as a [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508). Other tokens can also represent a Boolean value according to the conversion specified in the custom input type, [vBoolean](#Section_8a0254e2d0bc4714be02f7104ff17c1f).

Currency Values

A value that represents a currency is persisted in the file as a [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3). No other token type can represent a currency value. The only custom input type that preserves both a currency and its associated currency ID is [vDoubleEx](#Section_17ab59629f9e4b75a84af0cc80400b88).

Color Values

A value that represents a color value is persisted in the file as a [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882). Other tokens can also represent a color value according to the conversion specified in the custom input type, [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908).

Date Values

A value that represents a date is persisted in the file as a [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6). Other token types can represent a date according to the conversion specified in the [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) function.

Error Values

An error code that is returned as a result of a formula evaluation is persisted in the file as a [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2). When a function encounters an error value as one of its arguments, it returns the same error value. The exceptions are the functions [IsErr](#Section_daad281ef1d249c092d54c023d4a6679), [IsErrNA](#Section_d1c99e1d3db14b9084eb3bada4f9a4b6), [IsError](#Section_447254752e8446008be8460fda2f67d0), and [IsErrValue](#Section_afc2ec8cc6ab47c79fadb6151a48f7ca) which are specifically designed to detect certain error values.

##### Evaluation Stack

The evaluation stack is used to keep track of intermediate stages and precedence of operations when evaluating a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc). The expression is evaluated from beginning to end, and operands are pushed onto the stack as they are encountered. When [function tokens](#Section_e78706d7cd8c424bbc3bfc74904871ca) are encountered, the required number of operands is popped from the stack and the result of the operation is pushed back onto the stack. [Control tokens](#Section_41b89226e2a7471fb70c6d01003be607) are not pushed onto the stack. Evaluation begins with an empty stack, and when the evaluation is finished, there is exactly one token left on the stack. The remaining token is the result of the evaluation.

#### Unit Number

A Unit Number is a numeric value with a unit of measure. The numeric value is a [Custom Internal Unit Type](#Section_1ef4d381d8284ad094e4227751b7b6d2) or a value specified by [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6). The unit of measure determines how the numeric value is formatted and displayed in the user interface.

## ShapeGraphic XML Part

This part specifies the graphical elements that define the visual contents of a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) in the document before [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) is performed on it. These graphical elements MUST be expressed in Silverlight [**XAML**](#gt_891018ae-9381-4d29-82d7-20d6c8537472) as specified in [[MS-SLXV]](%5bMS-SLXV%5d.pdf) and MUST conform to the constraints defined in this section.

### XAML Terminology

The following terms are used to specify constraints on the [ShapeGraphic](#Section_91b39933d4194cff88431368a42a2258) part in this document:

Element of type *T*, where *T* is **Canvas**, **Ellipse**, **Rectangle**, **Path**, **Image**, **TextBlock**, **Glyphs**, **MatrixTransform**, **SolidColorBrush**, **ImageBrush**, or **Run**: Refers to an XML element that can be processed into an Object Node as described in section 6.6.2 of [[MS-XAML]](%5bMS-XAML%5d.pdf), the XamlType being *T* from the Silverlight Xaml Schema Information Set described in [[MS-SLXV]](%5bMS-SLXV%5d.pdf).

Element of type *T*.*M*, where *T* is **Canvas**, **Ellipse**, **Rectangle**, **Path, Glyphs**, or **Run** and *M* is **RenderTransform**, **Fill, Stroke**, **FontWeight**, or **FontStyle**: Refers to an XML element that can be processed into Member Nodes as described in section 6.6.5 of [MS-XAML], the XamlType being *T* and the XamlMember being *M* from the Silverlight Xaml Schema Information Set described in [MS-SLXV].

*M* property, where *M* is **Source**, **ImageSource**, **Fill**, **Visibility**, **Name**: Refers to either an XML child element that can be processed into a Member Node as described in section 6.6.5 of [MS-XAML], or an XML attribute that can be processed into a Member Node as described in section 6.6.3 of [MS-XAML], the XamlMember being *M* from the Silverlight Xaml Schema Information Set described in [MS-SLXV].

Path primitive : Refers to single-letter commands that are part of the **Data** attribute of an element of type **Path**, according to the parsing of that attribute specified in the **GeometrySyntax** Text Syntax Information Set in [[MS-WPFXV]](%5bMS-WPFXV%5d.pdf).

### XAML Resources

In the [ShapeGraphic](#Section_91b39933d4194cff88431368a42a2258) part, elements of type **Glyphs** refer to external [**font**](#gt_f8aa3f46-99d1-49bb-858f-b4bfa546c1c2) resources, and elements of type **Image** and **ImageBrush** refer to external image resources.

#### Fonts

All elements of type **Glyphs** in the [ShapeGraphic](#Section_91b39933d4194cff88431368a42a2258) part MUST have a **FontUri** attribute defined. If the **FontUri** attribute does not have a prefix of "CommonFonts/", then there MUST be a [Fonts](#Section_6b505bc0615e455f9af5cbf71b0e401e) part whose name is equal to the value of this attribute. When the **FontUri** attribute does have a prefix of "CommonFonts/", the value of this attribute after that prefix, which MUST be a value from the following table, specifies the [**font family**](#gt_bca5490d-d27e-4097-b05d-9efb09083dd2) used to render the glyphs, as well as whether the glyphs use bold or italic formatting.

| Value | Font family | Bold | Italic |
| --- | --- | --- | --- |
| 96124B06-B06C-4D87-9DC8-D4BBF5D7B681.odttf | Arial | False | False |
| C358B472-DE75-4EC3-B0D4-CD16217840DF.odttf | Arial | True | False |
| 4FB139D2-73CE-4404-8857-257D1164B8E2.odttf | Arial | False | True |
| A1A273E7-72DC-40C0-95F0-8C9DD99A69A4.odttf | Arial | True | True |
| 15BE407E-E656-49FB-BD0D-6956801B8457.odttf | Arial Black | False | False |
| 80AAA2AC-A1E3-4D23-A23A-629E77B9D6E3.odttf | Arial Narrow | False | False |
| 5B6F85F0-B9CA-4234-85D5-712569646CF5.odttf | Arial Narrow | True | False |
| 222735B6-9799-46F7-8179-CDFA36EB9772.odttf | Arial Narrow | False | True |
| 0F4CEC39-6EF4-494B-95C4-1C5393B64D7F.odttf | Arial Narrow | True | True |
| E94BD929-2AB4-4DCE-8733-ADA37D797930.odttf | Book Antiqua | False | False |
| FD94B59A-84D1-4A30-9DA0-609D670006A4.odttf | Book Antiqua | True | False |
| 3EAB2CEC-411A-4FBB-B52F-C4DAA4564A22.odttf | Book Antiqua | False | True |
| DCE069D4-DCE4-4DCF-B653-E5C3062890C5.odttf | Book Antiqua | True | True |
| 808543AF-7936-4CA8-B1B4-2A71EABA6571.odttf | Bookman Old Style | False | False |
| 81AD0CBD-900F-4EF6-8905-865617916412.odttf | Bookman Old Style | True | False |
| E3117E24-05C9-4CDA-92E8-7EA58D9B1180.odttf | Bookman Old Style | False | True |
| DF0D362F-ECF6-41CC-B34E-A222E2F48765.odttf | Bookman Old Style | True | True |
| 4C1AABE5-BA71-4F2C-8263-66731F995C4B.odttf | Calibri | False | False |
| 2FCC3378-38EC-463F-BEC2-438FD506EF47.odttf | Calibri | True | False |
| EEA6899E-27A6-4A9A-BA90-C83F21189897.odttf | Calibri | False | True |
| C4D86DCF-C433-4630-B4F8-BC8EFB8A51E9.odttf | Calibri | True | True |
| 6966055E-CCDF-4307-913C-BDFF248FD081.odttf | Cambria | False | False |
| 43223052-5DE8-484B-8EAA-48F50A2CDFBC.odttf | Cambria | True | False |
| 82B04200-E877-4DF3-9841-8091979F5F37.odttf | Cambria | False | True |
| F79EADC2-EC72-4D6D-8440-F3973033C747.odttf | Cambria | True | True |
| A3821346-4A5B-4182-8832-B5B2EF80A41B.odttf | Candara | False | False |
| 09EDCD55-8604-4B5E-9618-352681491E13.odttf | Candara | True | False |
| E73B8D74-2F83-414F-A9B6-1CA67B3BD3E8.odttf | Candara | False | True |
| 597038BC-C5F4-449E-8EFF-4F9460C0AC27.odttf | Candara | True | True |
| 67BF9C01-7EE5-448D-B713-CCBAA6B227D7.odttf | Century | False | False |
| DCFBF993-0CAA-4C71-A58E-1CDD969BD4B8.odttf | Comic Sans MS | False | False |
| C67B43C3-A5C3-41B6-9C47-1D54FDD4FD6D.odttf | Comic Sans MS | True | False |
| 865B4E77-E76B-4460-9ECB-E88E367B9E51.odttf | Consolas | False | False |
| CB370F8E-26AA-45D3-B0C1-E8AE76EE39D7.odttf | Consolas | True | False |
| A86B1F02-B649-4AC1-AED6-F06B39A2427F.odttf | Consolas | False | True |
| BF5AEE82-A2A1-4AA1-A9B7-7DB1732D4AB3.odttf | Consolas | True | True |
| DA5A5D60-F7F2-4520-AEA3-442AFAC8B032.odttf | Corbel | False | False |
| 24D25870-0E64-4226-A752-E0F9E41A580D.odttf | Corbel | True | False |
| EC4CF21B-B4A1-4D31-9CCD-EDB33D819FB8.odttf | Corbel | False | True |
| AAE9787A-E07D-49FB-9CF1-5B6EEC4AF38E.odttf | Corbel | True | True |
| D39AF8ED-AB87-4B59-B8C9-91CCE4C08A9D.odttf | Courier New | False | False |
| 6B2A0BA7-1A7A-4751-9FB8-D06A2E0CDF15.odttf | Courier New | True | False |
| 7B691A34-3EAA-4756-AC6F-615659B2C177.odttf | Courier New | False | True |
| 00E33E22-B4B7-4B8A-A1A7-EA05026E9926.odttf | Courier New | True | True |
| B6243CDB-E343-4FF9-86F2-7FE6B48C19FC.odttf | Franklin Gothic Book | False | False |
| DEF36608-0031-4DF9-81DD-E63A93642DB7.odttf | Franklin Gothic Book | False | True |
| 11B53828-77A2-4F10-BD06-56576BD2B2DD.odttf | Franklin Gothic Demi | False | False |
| 13C1D5A6-C56C-4D8E-AD71-3838DC49FB26.odttf | Franklin Gothic Demi | False | True |
| AA42910A-F8C6-4293-8116-A4D6154E262C.odttf | Franklin Gothic Demi Cond | False | False |
| 61025B6F-F0E9-407F-9524-B3AF18AD2577.odttf | Franklin Gothic Heavy | False | False |
| C436D513-1D30-4CE3-8A2A-E7A3F83DAAF1.odttf | Franklin Gothic Heavy | False | True |
| F369B7AE-EE88-4B34-ADEC-CB15AF8A6B36.odttf | Franklin Gothic Medium | False | False |
| F5138333-541D-4CC9-8E00-D341395C3278.odttf | Franklin Gothic Medium | False | True |
| 61855DFC-5F03-498A-90C5-02D4DFCE359D.odttf | Franklin Gothic Medium Cond | False | False |
| 3E3519FA-71B1-4F7B-8161-7AD77198FB2C.odttf | Garamond | False | False |
| 8A7CF9AD-5C0B-423F-AD30-8CF1F6B29719.odttf | Garamond | True | False |
| F5109707-DE9D-416E-8535-C99238F13E8A.odttf | Garamond | False | True |
| C7E23E6B-DD21-4F21-A7D5-8ABBA7692910.odttf | Georgia | False | False |
| 9F0D1069-4284-46EA-B336-7A8649D6BF55.odttf | Georgia | True | False |
| 9F56D674-6626-4E6B-9223-D8C7AEC2D901.odttf | Georgia | False | True |
| 9AFCD208-D49C-4CAE-9BE5-C256BFC488B0.odttf | Georgia | True | True |
| 2F23B780-62D4-4384-928D-5A5733F6011D.odttf | Segoe UI | False | False |
| 39943FD9-9366-4596-BE1C-2FFA67FFC4D8.odttf | Segoe UI | True | False |
| 297C663E-9645-485B-B560-996CF4D7E544.odttf | Segoe UI | False | True |
| C0D43036-F10C-4CAE-8A01-C26DFEB3D46B.odttf | Segoe UI | True | True |
| 47E24325-C88C-450B-B178-769A8BE88241.odttf | Symbol | False | False |
| 8D1E89DD-FC5F-4230-8652-F172DA76E95E.odttf | Tahoma | False | False |
| C1D99FC0-2154-43D4-9E21-CADA3D102904.odttf | Tahoma | True | False |
| D56DC39A-1D0F-4F8A-9F87-493EBD134E45.odttf | Times New Roman | False | False |
| CE46F0D4-CDA3-41DA-98F7-7AE756189318.odttf | Times New Roman | True | False |
| 12FEEB78-5B96-44C3-9AF6-3237613F26C9.odttf | Times New Roman | False | True |
| EB2B113F-1CC3-462D-88E8-ACE2599DA9B4.odttf | Times New Roman | True | True |
| DF71F07D-0181-4119-B75E-F96878BD1D63.odttf | Trebuchet MS | False | False |
| 4D44B836-2782-4D27-8AE0-DBEBAC456B46.odttf | Trebuchet MS | True | False |
| 1BFDA688-1B13-4263-9F1F-FCFDD57A9F28.odttf | Trebuchet MS | False | True |
| 54EA0ACE-735A-45C0-A667-55462535C685.odttf | Trebuchet MS | True | True |
| 0536AE94-BE0F-4463-A07B-D5FE77A2D4B5.odttf | Verdana | False | False |
| 6E52AB96-E48C-4AB7-9B8D-0E953F032BD4.odttf | Verdana | True | False |
| BA79D24C-3CB7-4AA7-BBAD-7A9040F8EF56.odttf | Verdana | False | True |
| 8F4AA1FC-1A6E-48D9-AAC0-7C5FB1D78982.odttf | Verdana | True | True |
| 22323AAC-91D5-4C5B-92BC-851531B77A51.odttf | Wingdings | False | False |

Every **Glyphs** element whose **FontUri** attribute has a prefix of "CommonFont/" MUST have a **UnicodeString** attribute defined. For each value in the left-hand column of the following table, every **Glyphs** element whose **FontUri** is equal to that value MUST have a **UnicodeString** attribute composed of characters whose [**Unicode code points**](#gt_b178e42f-79a3-4a19-8af7-a3c5e66e7bd9) are in the corresponding cell in the right-hand column.

| FontUri | Unicode code points MUST be |
| --- | --- |
| CommonFont/22323AAC-91D5-4C5B-92BC-851531B77A51.odttf | 32, 74, 76, 108, 110, 113, 118, 158, 167, 178, 216, 223, 224, 232, 233, 234 or 252 |
| CommonFont/47E24325-C88C-450B-B178-769A8BE88241.odttf | 32, 43, 45, 61, 97, 98, 109, 152, 176, 180, 183, 190, 210, 211, 212 or 226 |
| CommonFont/*V*, where *V* is another value | Greater than or equal to 32 and less than 127, or one of the following values: 163, 167, 169, 174, 189, 224, 225, 228, 232, 233, 243, 246, 252, 8211, 8212, 8216, 8217, 8220, 8221, 8226, 8230, 8482, and 8722 |

#### Images

All elements of type **Image** in the [ShapeGraphic](#Section_91b39933d4194cff88431368a42a2258) part MUST have a **Source** property defined, the value of which MUST be the name of an [Images](#Section_4cb2e2621f8d44a6b20f0be3a3fb0179) part.

All elements of type **ImageBrush** in the ShapeGraphic part MUST have an **ImageSource** property defined, the value of which MUST be the name of an Images part.

### XAML Shapes

All graphical elements that correspond to [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) in the [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) are specified as [**XAML**](#gt_891018ae-9381-4d29-82d7-20d6c8537472) **UIElement** objects and MUST conform to the Silverlight XAML syntax as specified in [[MS-SLXV]](%5bMS-SLXV%5d.pdf) section 1.22.31.

For each [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0) element in the [ShapeInfo XML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac), there MUST be an element of type **Canvas** whose **Name** attribute is equal to the **Name** attribute of the CT\_ShapeInfo. That **Canvas** element MUST have a **Tag** attribute defined with a non-empty value. The child elements of that **Canvas** element specify the appearance of the shape that corresponds to the CT\_ShapeInfo.

### XAML Recalculated Shapes

A [**XAML**](#gt_891018ae-9381-4d29-82d7-20d6c8537472) shape that that can be updated by [formula expressions](#Section_1639b4e843e14d409531adf5c2fe8bbc) is called [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075). There are five types of data graphics:

* [Sheet element](#Section_94a21e97d2934aee9999f20030a1ee56) data graphic
* [Formatting element](#Section_b08c0179146b4b2dbdce35c55c7f7737) data graphic
* [Geometry element](#Section_23c25866532c499aa07c9dc558139a9b) data graphic
* [Text element](#Section_b061c2bde75d48edb51047479b51c6ec) data graphic
* [Image element](#Section_385a6cb3903347b3acca493e5b363057) data graphic

The following sections specify the constraints that each type of data graphics MUST conform to so that it can be updated during [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972).

#### Sheet Elements

A **sheet** element [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) is a [**XAML**](#gt_891018ae-9381-4d29-82d7-20d6c8537472) element of type **Canvas** whose geometric positioning information can be updated by [formula expressions](#Section_1639b4e843e14d409531adf5c2fe8bbc). The **Name** attribute of this **Canvas** MUST be equal to the **Name** attribute of a [CT\_Sheet](#Section_50dec7c1701c454c90ca55fbec1e724a) element in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087).

##### Shape Transform

For each [CT\_Sheet](#Section_50dec7c1701c454c90ca55fbec1e724a) element in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087) that includes **Angle**, **FlipX**, **FlipY**, **PinX**, **PinY**, **LocPinX** or **LocPinY** among its defined attributes, the corresponding [sheet element](#Section_94a21e97d2934aee9999f20030a1ee56) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) MUST have a child **Canvas.RenderTransform** element that has a child element of type **MatrixTransform**. This **Canvas.RenderTransform** element is called a Shape Transform and specifies an affine transformation of the geometry elements included in the **sheet** element data graphic.

The measurement units for the **PinX, PinY, LocPinX, LocPinY** attributes are specified as length [internal units](#Section_1ef4d381d8284ad094e4227751b7b6d2). The measurement units for the **Angle** attribute are specified as angle internal units. The **FlipX** and **FlipY** attributes do not have measurement units.

The **MatrixTransform** used to apply these transforms to the **sheet** element data graphic is constructed by applying the transforms in the following order:

1. Translate by (-**LocPinX**, -**LocPinY**)
2. Mirror about the X axis if **FlipX** is equal to 1 and about the Y axis if **FlipY** is equal to 1
3. Rotate around origin by **Angle**
4. Translate by (**PinX**, **PinY**)

#### Formatting Elements

A formatting element [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) is a [sheet element](#Section_94a21e97d2934aee9999f20030a1ee56) data graphic with supplementary constraints that allow the visual formatting properties of the **Canvas** to be updated by [formula expressions](#Section_1639b4e843e14d409531adf5c2fe8bbc). Each [CT\_Sheet](#Section_50dec7c1701c454c90ca55fbec1e724a) element associated with a formatting element data graphic MUST specify the **FillBackground**, **FillBackgroundTrans**, **FillForeground**, **FillForegroundTrans**, **FillPattern**, **LineColor**, **LineColorTrans** or **LineWeight** attributes or MUST have one or more child elements of type [CT\_Geometry](#Section_984113db7f1a430fb5b90f70e3593b50).

A formatting element data graphic MUST contain child elements of type **Path**, **Ellipse**, **Rectangle**, **Canvas**, **TextBlock** or **Canvas.RenderTransform**.

If a formatting element data graphic has a child element of type **Path**, then it MUST NOT have a child element of type **Rectangle**, and it MUST NOT have a child element of type **Ellipse** that is not a [Model Ellipse](#Section_9fcca8d6c94b4989bbfb6dea992d1737).

If a formatting element data graphic has a child element of type **Rectangle**, then it MUST NOT have a child element of type **Path** or **Ellipse**.

##### Shadow Canvas

A child **Canvas** element of a [formatting element](#Section_b08c0179146b4b2dbdce35c55c7f7737) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) with a **Name** value equal to the string "Shdw" is a Shadow Canvas*.* It MUST contain only child elements of types **Path**, **Ellipse**, **Rectangle** or **Canvas.RenderTransform**.

The **Path**, **Rectangle** or **Ellipse** elements that are child elements of a Shadow Canvas specify the appearance and the relative position of the shadow of the formatting element data graphic. A Shadow Canvas is recalculated only according to the formulas in the **LineWeight** attribute and the **Geometry** child elements of a [CT\_Sheet](#Section_50dec7c1701c454c90ca55fbec1e724a).

All **Path**, **Rectangle** or **Ellipse** elements that are part of a formatting element data graphic MUST be either child elements of the formatting element data graphic, or child elements of the corresponding Shadow Canvas.

##### Fill Attributes

For each [CT\_Sheet](#Section_50dec7c1701c454c90ca55fbec1e724a) element in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087) that includes **FillBackground**, **FillBackgroundTrans**, **FillForeground**, **FillForegroundTrans**, or **FillPattern** among its defined attributes, all child **Path**, **Ellipse** or **Rectangle** elements of the corresponding [formatting element](#Section_b08c0179146b4b2dbdce35c55c7f7737) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) MUST NOT have a **Fill** attribute. Any of these elements that have a **Fill** property defined MUST have it defined as a child **Path.Fill**, **Ellipse.Fill** or **Rectangle.Fill** element.

##### Stroke Attributes

For each [CT\_Sheet](#Section_50dec7c1701c454c90ca55fbec1e724a) element in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087) that has a **LineColor** or **LineColorTrans** attribute, all child **Path**, **Ellipse** or **Rectangle** elements of the corresponding [formatting element](#Section_b08c0179146b4b2dbdce35c55c7f7737) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) MUST NOT have a **Stroke** attribute. Any of these elements that have a **Stroke** property defined MUST have it defined as a child **Path.Stroke**, **Ellipse.Stroke** or **Rectangle.Stroke** element that has a child **SolidColorBrush** element.

#### Geometry Elements

A geometry element [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) is a child **Canvas** of a [sheet element](#Section_94a21e97d2934aee9999f20030a1ee56) data graphic whose geometry can be updated by [formula expressions](#Section_1639b4e843e14d409531adf5c2fe8bbc).The **Name** attribute of this child **Canvas** MUST contain the string "\_G*n*\_", where *n* MUST be decimal number and where *n* MUST equal the value of one of the **Index** attributes of a [CT\_Geometry](#Section_984113db7f1a430fb5b90f70e3593b50) element contained within the [CT\_Sheet](#Section_50dec7c1701c454c90ca55fbec1e724a) associated with the sheet element data graphic.

For each CT\_Geometry element in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087), the corresponding sheet element data graphic MUST contain at least one corresponding geometry element data graphic.

##### Geometry Path Specifications

For a [CT\_Geometry](#Section_984113db7f1a430fb5b90f70e3593b50) element in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087) with a **BoundPts** attribute defined, all the corresponding [geometry element](#Section_23c25866532c499aa07c9dc558139a9b) [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) that are either [Model Elements](#Section_1c468e4074744dd09853b7f29ed4c462), or such that there is no Model Element associated to the same CT\_Geometry, MUST be of type **Path** or of type **Ellipse**. When they are of type **Path**, they MUST have a **Data** attribute that specifies the path of the geometry. The value of any of these **Data** attributes MUST contain a number of path primitives that is greater than or equal to the **pointindex1** of the corresponding CT\_Geometry **BoundPts** attribute. These path primitives are indexed in the order of their appearance in the **Data** attribute, the first one having the index 1.

For each CT\_Geometry element in the DataGraphic XML Part whose **BoundPts** attribute has two indices, the corresponding path primitive in each geometry element data graphic MUST be of type **Point** as specified in [[MS-SLXV]](%5bMS-SLXV%5d.pdf) section 1.62 or **LineSegment** as specified in [MS-SLXV] section 1.22.19.3.

For each CT\_Geometry element in the DataGraphic XML Part whose **BoundPts** attribute has three indices, the corresponding path primitive in each geometry element data graphic MUST be of type **ArcSegment** as specified in [MS-SLXV] section 1.22.19.1**.**

#### Model Elements

Model Elements are invisible [**XAML**](#gt_891018ae-9381-4d29-82d7-20d6c8537472) elements that are intended for informational purposes.

##### Model Paths

A **Path** element that is a child of a [sheet element](#Section_94a21e97d2934aee9999f20030a1ee56) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) and whose **Name** attribute contains the string "\_model" is a Model Path.

A Model Path specifies a geometry element data graphic, without any corner rounding applied.

A Model Path is rendered after a corner rounding radius is applied to each corner when [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) is performed.

A Model Path MUST have a **Visibility** property defined as "Collapsed".

A Model Path MUST have a **Tag** attribute with an xsd:double ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.5) value, which specifies the corner rounding radius to be applied to the path before diagram update.

For a [CT\_Sheet](#Section_50dec7c1701c454c90ca55fbec1e724a) element in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087) that has a **Rounding** attribute whose value is different from "0:65" or "0:98", the corresponding sheet element data graphic MUST contain a Model Path that is specific to each child [CT\_Geometry](#Section_984113db7f1a430fb5b90f70e3593b50) element of that CT\_Sheet element.

##### Model Ellipses

An **Ellipse** element that is a child of a [sheet element](#Section_94a21e97d2934aee9999f20030a1ee56) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) and whose **Name** attribute contains the string "\_model" is a Model Ellipse.

A Model Ellipse MUST have a **Visibility** attribute set to "Collapsed".

A Model Ellipse specifies an ellipse that is rendered after the position of its center and its major and minor axis are recalculated according to the formulas in the corresponding [CT\_Geometry](#Section_984113db7f1a430fb5b90f70e3593b50) elements of the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087).

#### Text Elements

A text element [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) is a [**XAML**](#gt_891018ae-9381-4d29-82d7-20d6c8537472) element of type **Canvas** whose text can be updated by [formula expressions](#Section_1639b4e843e14d409531adf5c2fe8bbc). The **Name** attribute of this **Canvas** MUST equal the value of a **Name** attribute of a [CT\_ShapeText](#Section_80e46fba919845399e9b8c1147fccc25) element in a [ShapeTextBinding XML Part](#Section_2e21d026e11d46ee860a75c6b75bd84a).

##### Glyph Canvas

A child **Canvas** element of a [text element](#Section_b061c2bde75d48edb51047479b51c6ec) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) with no Name property defined is the Glyph Canvas corresponding to the text.

Each child of a Glyph Canvas MUST be of type **Glyphs** or **Canvas.RenderTransform**, and specify the appearance, position and color of the text before [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972).

A child **Glyphs** element of a Glyph Canvas MUST have a child **Glyphs.Fill** element. That element MUST have a child **SolidColorBrush** element.

A Glyph Canvas can have transforms applied to it as specified by the **TextAngle**, **TextPinX**, **TextPinY**, **TextLocPinX** or **TextLocPinY** attributes of the corresponding [CT\_Sheet](#Section_50dec7c1701c454c90ca55fbec1e724a) element in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087).

The measurement units for the **TextPinX, TextPinY, TextLocPinX, TextLocPinY** attributes are specified as length [internal units](#Section_1ef4d381d8284ad094e4227751b7b6d2). The measurement units for the **TextAngle** attribute are specified as angle internal units.

The **RenderTransform** matrix is constructed by applying the transforms in the following order:

1. Translate by (-**TextLocPinX**, -**TextLocPinY**)
2. Rotate around origin by **TextAngle**
3. Translate by (**TextPinX**, **TextPinY**)

##### Text Model

A [text element](#Section_b061c2bde75d48edb51047479b51c6ec) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) MUST have a child **TextBlock** element, and that element MUST have exactly one child **Run** element. That **TextBlock** element is the Text Model.

The Text Model specifies the value and typographical attributes of the text before [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) is performed.

A Text Model MUST have a **Visibility** property set to the value "Collapsed".

A Text Model MUST have the following attributes defined: **Width**, **Height** and **Tag**.

The value of the **Tag** attribute of a Text Model specifies additional positioning information for the text within the text element data graphic. It MUST conform to the following syntax:

*Syntax:* horizontal\_alignment;vertical\_alignment;rtl;line\_spacing;margin\_left;margin\_right;margin\_top;margin\_bottom

**horizontal\_alignment** specifies how the text is horizontally located within the [Glyph Canvas](#Section_4588ece693c04e5b8c803e0870294bd2) and MUST be a value from the following table:

| Value | Meaning |
| --- | --- |
| Left | The text lines are flushed to the left of the text block. |
| Center | Each text line is centered in the text block. |
| Right | The text lines are flushed to the right of the text block. |

**vertical\_alignment** specifies how the text is vertically located within the Glyph Canvas and MUST be a value from the following table:

| Value | Meaning |
| --- | --- |
| Top | The text lines are flushed to the top of the text block. |
| Center | The text lines are centered in the text block. |
| Bottom | The text lines are flushed to the bottom of the text block. |

**rtl** is unused and MUST be ignored.

**line\_spacing** is an xsd:double ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.5) that specifies the distance between lines, as a factor of the value of the **FontSize** attribute in the child **Run** element.

**margin\_left** is an xsd:double ([XMLSCHEMA2] section 3.2.5) that specifies the distance in inches between the left edge of the glyph canvas and the left edge of the space occupied by the text glyphs.

**margin\_right** is an xsd:double ([XMLSCHEMA2] section 3.2.5) that specifies the distance in inches between the right edge of the glyph canvas and the right edge of the space occupied by the text glyphs.

**margin\_top** is an xsd:double ([XMLSCHEMA2] section 3.2.5) that specifies the distance in inches between the top edge of the glyph canvas and the top edge of the space occupied by the text glyphs.

**margin\_bottom** is an xsd:double ([XMLSCHEMA2] section 3.2.5) that specifies the distance in inches between the bottom edge of the glyph canvas and the bottom edge of the space occupied by the text glyphs.

##### Text Run

A child **Run** element of a [Text Model](#Section_81df0650e4aa403abaa976490f1f9ab5) is called a Text Run.

The **Name** attribute of each Text Run MUST be equal to the **Name** attribute of the [CT\_TextRun](#Section_9b45f00ef7384cfba633da8bca9b0af4) child of the corresponding [CT\_ShapeText](#Section_80e46fba919845399e9b8c1147fccc25) element.

Each Text Run MUST have the attributes **FontFamily** and **FontSize** defined.

Each Text Run MUST NOT have a **Run.FontWeight** or **Run.FontStyle** child element. If the **FontWeight** or **FontStyle** properties are defined, they MUST be defined as attributes.

#### Image Elements

An image element [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) is a [**XAML**](#gt_891018ae-9381-4d29-82d7-20d6c8537472) element of type **Canvas** that contains one or more **Image** elements whose source image can be updated by [formula expressions](#Section_1639b4e843e14d409531adf5c2fe8bbc). The **Name** attribute of this **Canvas** MUST equal the value of a **DefShapeID** attribute of a [CT\_DataGraphicDef](#Section_52cb1fef11f541c5a70f89aec95917a0) element in the DataGraphic XML part.

Each image element data graphic MUST have at least one child element of type **Image**.

## XML Parts

### Introduction

The following sections specify the structure of the XML [parts](#Section_66eb42ec5ca8459696291233020c701f) that are in the ZIP archive of the "VisioServerData" [stream](#Section_1fc77f54c9d74549aca5a7bed511c016).

### App XML Part

This [part](#Section_66eb42ec5ca8459696291233020c701f) specifies the properties of the document. App Schema, see section [5.1](#Section_ebbad19a77c24cf0974c080cc65c0407).

### Core XML Part

An optional [part](#Section_66eb42ec5ca8459696291233020c701f) that specifies additional properties of the document.

### DataBinding XML Part

This [part](#Section_66eb42ec5ca8459696291233020c701f) specifies the [data binding](#Section_b9b194892da24d77a8a51c99bb866fe1) information that is used to link external data to [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) in [drawing pages](#Section_ebba8748c5a54ac7a4191c6a70307640) of the document.

#### Global Elements

##### BindingConnections

A [CT\_BindingConnections](#Section_4fc00ba8f24243b69cc791796bc6c238) element.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this element.

1. <xsd:element name="BindingConnections" type="CT\_BindingConnections"/>

#### Complex Types

##### CT\_PrimaryKeyValue

*Referenced by:* [CT\_PrimaryKeyValues](#Section_37d1d095aebc4480b1cf30acc6990811)

The **CT\_PrimaryKeyValue** complex type specifies the value of one component of the [**primary key**](#gt_e4d8c530-39c1-4fc6-8ccc-8d51a221158d) in a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2).

*Attributes:*

**Value :** An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the value of a [**field**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc) that is a component of the primary key.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_PrimaryKeyValue">
2. <xsd:attribute name="Value" type="xsd:string"/>
3. </xsd:complexType>

##### CT\_PrimaryKeyValues

*Referenced by:* [CT\_Binding](#Section_5ad7ddf78438400abe979591e0cca24a)

The **CT\_PrimaryKeyValues** complex type specifies the values of all the components of the [**primary key**](#gt_e4d8c530-39c1-4fc6-8ccc-8d51a221158d) of the [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) that is bound to a [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Child Elements:*

**PrimaryKey :** A list of [CT\_PrimaryKeyValue](#Section_72fa7d7e26164b8bbb1b3b72453e0462) elements. The count and the order of the CT\_PrimaryKeyValue elements in this list MUST match the **PrimaryKeys.PrimaryKey** child elements of its containing [CT\_DataRecordset](#Section_09771dd2110c41e9ad2b609514a2c79c), whose **ID** attribute is equal to the **RecordsetID** attributeof this element’s containing [CT\_BindingConnection](#Section_7692f89f4a9a4267b1b5ba319675ca62).

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_PrimaryKeyValues">
2. <xsd:sequence>
3. <xsd:element name="PrimaryKey" minOccurs="1" maxOccurs="unbounded" type="CT\_PrimaryKeyValue"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_Binding

*Referenced by:* [CT\_Bindings](#Section_096f1897249c40c487ba6023f438c74e)

The **CT\_Binding** complex type specifies a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) that is bound to a [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Child Elements:*

**PrimaryKeys :** An optional [CT\_PrimaryKeyValues](#Section_37d1d095aebc4480b1cf30acc6990811) element which specifies the recordset row that is bound to a shape addressed using the [primary key method](#Section_aebfb9093e8c4dbdb11c988bbd424388). If the **Row** attribute exists, this element MUST NOT exist. If the **Row** attribute does not exist, this element MUST exist.

*Attributes:*

**BindingID :** An xsd:integer ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.3.13) attribute that specifies the identifier for this binding. It MUST be unique amongst all **BindingID** attributes in the document.

**Row :**  An optional xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the one-based row number used to address a row using the [row-order method](#Section_85d62e47587a4311aeb79b93ba288da6). If the **PrimaryKeys** element exists, this attribute MUST NOT exist. If the **PrimaryKeys** element does not exist, this attribute MUST exist.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_Binding">
2. <xsd:sequence>
3. <xsd:element name="PrimaryKeys" minOccurs="0" maxOccurs="1" type="CT\_PrimaryKeyValues"/>
4. </xsd:sequence>
5. <xsd:attribute name="BindingID" type="xsd:integer"/>
6. <xsd:attribute name="Row" type="xsd:integer" use="optional"/>
7. </xsd:complexType>

##### CT\_Bindings

*Referenced by:* [CT\_BindingConnection](#Section_7692f89f4a9a4267b1b5ba319675ca62)

The **CT\_Bindings** complex type specifies all the [**rows**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) in a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) that are bound to [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Child Elements:*

**Binding :** A list of [CT\_Binding](#Section_5ad7ddf78438400abe979591e0cca24a) elements.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_Bindings">
2. <xsd:sequence>
3. <xsd:element name="Binding" minOccurs="1" maxOccurs="unbounded" type="CT\_Binding"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_BindingConnection

*Referenced by:* [CT\_BindingConnections](#Section_4fc00ba8f24243b69cc791796bc6c238)

The **CT\_BindingConnection** complex type specifies a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) that has [**rows**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) of data that are bound to [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Child Elements:*

**Bindings :** A [CT\_Bindings](#Section_096f1897249c40c487ba6023f438c74e) element.

*Attributes:*

**RecordsetID :** An xsd:unsignedLong ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.3.21) attribute that specifies the identifier of the recordset. It MUST match an **ID** attribute of a [CT\_DataRecordset](#Section_09771dd2110c41e9ad2b609514a2c79c) element in the [DataConnection XML part](#Section_c489eefb7a014941bc37a4d332e66dc8).

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_BindingConnection">
2. <xsd:sequence>
3. <xsd:element name="Bindings" minOccurs="1" maxOccurs="1" type="CT\_Bindings"/>
4. </xsd:sequence>
5. <xsd:attribute name="RecordsetID" type="xsd:unsignedLong"/>
6. </xsd:complexType>

##### CT\_BindingConnections

*Referenced by:* [BindingConnections](#Section_78da47b061264b4388ca71683d5dd702)

The **CT\_BindingConnections** complex type specifies all the [recordsets](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) in the document that have [**rows**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) of data that are bound to [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Child Elements:*

**BindingConnection :** An optional list of [CT\_BindingConnection](#Section_7692f89f4a9a4267b1b5ba319675ca62) elements. CT\_BindingConnection elements are not present if the shapes are not bound to data from recordsets.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_BindingConnections">
2. <xsd:sequence>
3. <xsd:element name="BindingConnection" minOccurs="0" maxOccurs="unbounded" type="CT\_BindingConnection"/>
4. </xsd:sequence>
5. </xsd:complexType>

### DataConnection XML Part

This [part](#Section_66eb42ec5ca8459696291233020c701f) specifies the [data connection](#Section_28d3d364de644cada48144231aed4d7b) information needed to [**query**](#gt_37fbc661-f744-48fa-9d8e-f34513cab9c2) [**data sources**](#gt_e091613c-6901-4874-b9b2-27273ead1075) and [refresh](#Section_8aae518c4bca49968a84541ff996d01a) the [recordsets](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) referenced by a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25).

#### Global Elements

##### Connections

A [CT\_Connections](#Section_812ac2f1a2e94ace94f7e6fe2837009c) element.

1. <xsd:element name="Connections" type="CT\_Connections"/>

#### Complex Types

##### CT\_DataConnection

*Referenced by:* [CT\_DataConnections](#Section_918bfb0865234923a664b44f66011e43)

The **CT\_DataConnection** complex type specifies a [data connection](#Section_28d3d364de644cada48144231aed4d7b).

*Attributes:*

**ID :** An xsd:unsignedLong ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.3.21) attribute that specifies the identifier of a **CT\_DataConnection**. It MUST be unique amongst all the **DataConnection** child elements of the containing CT\_DataConnections.

**ConnectionString :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the [**connection string**](#gt_03a9d0ca-2f10-4f3d-b910-052714a96f7d) to a [**data source**](#gt_e091613c-6901-4874-b9b2-27273ead1075). It MUST be a valid connection string as specified in [[MS-ODBCSTR]](%5bMS-ODBCSTR%5d.pdf#Section_13b4e848b36c4b11acced6bf199d5391). The type of the data source specified in the connection string MUST be from the following table.

|  |  |
| --- | --- |
| Type of data source | Syntax |
| [**OLE DB**](#gt_333f4fb1-4882-48df-bce6-f9961b408f31) | MUST be a valid connection string as specified in [MS-ODBCSTR] |
| [**ODBC**](#gt_7883fa02-8dc0-4154-894f-fe3a7bff153e) | MUST be a valid connection string as specified in [MS-ODBCSTR] |
| [**List**](#gt_04ce231e-214c-44fd-b7ba-7cc19eee79bf) | MUST be in the following format:  "PROVIDER=WSS;DATABASE=*list URL*;LIST={*list GUID*};*viewparam*"   * *list URL* is the [**URL**](#gt_433a4fb7-ef84-46b0-ab65-905f5e3a80b1) of a list * *list GUID* is the [**GUID**](#gt_f49694cc-c350-462d-ab8e-816f0103c6c1) of the list * *viewparam* MUST be in the format "VIEW={*view GUID*};" if the data source is a [**view**](#gt_3f793b0b-9509-4df0-89f9-92f07954beb8) of a list, where *view GUID* is the GUID of the view; otherwise, *viewparam MUST* be an empty string |
| [**Workbook**](#gt_343c4660-90e1-4d86-b9cc-5007075d9dfe) | MUST be in the following format:  "DataModule=Microsoft.Office.Visio.Server.EcsDataHandler,Microsoft.Office.Visio.Server; Data Source=*workbook URL*;Extended Properties='HDR=*hdrvalue*;';"   * *hdrvalue* equals YES if the first [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) of data is the [**header row**](#gt_42711baf-2679-445d-a994-0eadd91b1a38), and NO otherwise * *workbook URL* is the URL of a workbook |
| **Custom** | MUST be in the following format:  "DataModule=*class name,assembly name*;Add-in key*=value pairs*;"   * *class name* is a [**class name**](#gt_c7d23c85-a7a1-412b-93d5-572ea0918303) * *assembly name* is an [**assembly name**](#gt_2e4c6e9e-9189-4150-a9f6-3b3c366f75cf) * *value pairs* is a semicolon-separated set of key-value pairs specific to the [**add-in**](#gt_a3be101e-9d37-484a-a5e6-b70d559146c6) |

**Filename :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the full path of an [**ODC file**](#gt_e6fb952a-b6b6-40af-ad1b-ea6cf97da509). If the **ConnectionString** attribute exists, this attribute MUST NOT exist. If the **ConnectionString** attribute does not exist, this attribute MUST exist.

1. <xsd:complexType name="CT\_DataConnection">
2. <xsd:attribute name="ID" type="xsd:unsignedLong" use="required"/>
3. <xsd:attribute name="ConnectionString" type="xsd:string" use="optional"/>
4. <xsd:attribute name="Filename" type="xsd:string" use="optional"/>
5. </xsd:complexType>

##### CT\_DataConnections

*Referenced by:* [CT\_Connections](#Section_812ac2f1a2e94ace94f7e6fe2837009c)

The **CT\_DataConnections** complex type specifies all the [data connections](#Section_28d3d364de644cada48144231aed4d7b) of the document.

*Child Elements:*

**DataConnection :** An optional list of [CT\_DataConnection](#Section_7e5607b6320948fa9f6dc172500e1dd7) elements. CT\_DataConnection elements are not present if the document does not have data connections.

1. <xsd:complexType name="CT\_DataConnections">
2. <xsd:sequence>
3. <xsd:element name="DataConnection" minOccurs="0" maxOccurs="unbounded" type="CT\_DataConnection"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_PrimaryKey

*Referenced by:* [CT\_PrimaryKeys](#Section_514c708d371c4dd3907ca4875ec1fb91)

The **CT\_PrimaryKey** complex type specifies a component of a [**primary key**](#gt_e4d8c530-39c1-4fc6-8ccc-8d51a221158d).

*Attributes:*

**Key :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the name of a [**field**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc) that is a component of a primary key. It MUST be a value from the **Name** attribute of a **DataColumn** child element of the [CT\_DataColumns](#Section_2084702f494442eb8bf757fc33425d26) of a [CT\_DataRecordset](#Section_09771dd2110c41e9ad2b609514a2c79c) whose primary key is being specified.

1. <xsd:complexType name="CT\_PrimaryKey">
2. <xsd:attribute name="Key" type="xsd:string" use="required"/>
3. </xsd:complexType>

##### CT\_PrimaryKeys

*Referenced by:* [CT\_DataRecordset](#Section_09771dd2110c41e9ad2b609514a2c79c)

The **CT\_PrimaryKeys** complex type specifies all the components of the [**primary key**](#gt_e4d8c530-39c1-4fc6-8ccc-8d51a221158d) of a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2).

*Child Elements:*

**PrimaryKey :** A list of [CT\_PrimaryKey](#Section_80c4c5601faa4ad48c8d3aab7b8b3188) elements. The number of elements in this list MUST NOT exceed the number of **DataColumn** child elements of the [CT\_DataColumns](#Section_2084702f494442eb8bf757fc33425d26) of a CT\_DataRecordset whose primary key is being specified.

1. <xsd:complexType name="CT\_PrimaryKeys">
2. <xsd:sequence>
3. <xsd:element name="PrimaryKey" minOccurs="1" maxOccurs="unbounded" type="CT\_PrimaryKey"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_DataColumn

*Referenced by:* [CT\_DataColumns](#Section_2084702f494442eb8bf757fc33425d26)

The **CT\_DataColumn** complex type specifies a [**field**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc) in a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2).

*Attributes:*

**Name :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the identifier of the field.

**DisplayName :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the display name of the field.

**Type :**  An xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the [**ADO**](#gt_3477d72a-874b-4a2a-b6a9-1431c128df49) [**data type**](#gt_0d327bc0-a461-4723-8beb-d935fdab4895) of the field. It MUST be a value from the standard ADO data types as listed in following table. For more information about these data types, see [[MSDN-OLEDB]](https://go.microsoft.com/fwlink/?LinkId=90054).

|  |  |
| --- | --- |
| Value | Meaning |
| 0 | Empty |
| 2 | Small Integer |
| 3 | Integer |
| 4 | Single Precision Floating Point |
| 5 | Double Precision Floating Point |
| 6 | Currency |
| 7 | Date |
| 8 | BSTR |
| 9 | IDispatch |
| 10 | Error |
| 11 | [**Boolean**](#gt_1d79d7a7-ba2c-4b34-931c-7ba8057c87b2) |
| 12 | Variant |
| 13 | IUnknown |
| 14 | Decimal |
| 16 | Tiny Integer |
| 17 | Unsigned Tiny Integer |
| 18 | Unsigned Small Integer |
| 19 | Unsigned Integer |
| 20 | Big Integer |
| 21 | Unsigned Big Integer |
| 64 | File Time |
| 72 | [**GUID**](#gt_f49694cc-c350-462d-ab8e-816f0103c6c1) |
| 128 | Binary |
| 129 | Character |
| 130 | Wide Character |
| 131 | Numeric |
| 132 | User Defined Type |
| 133 | Database Date |
| 134 | Database Time |
| 135 | Database Time Stamp |
| 136 | Chapter |
| 138 | Property Variant |
| 139 | Variable Number |
| 200 | Variable Character |
| 201 | Long Variable Character |
| 202 | Wide Variable Character |
| 203 | Long Wide Variable Character |
| 204 | Variable Binary Blob |
| 205 | Long Variable Binary Blob |
| 8192 | Array |

1. <xsd:complexType name="CT\_DataColumn">
2. <xsd:attribute name="Name" type="xsd:string" use="required"/>
3. <xsd:attribute name="DisplayName" type="xsd:string" use="required"/>
4. <xsd:attribute name="Type" type="xsd:integer" use="required"/>
5. </xsd:complexType>

##### CT\_DataColumns

*Referenced by:* [CT\_DataRecordset](#Section_09771dd2110c41e9ad2b609514a2c79c)

The **CT\_DataColumns** complex type specifies the [**fields**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc) of a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2).

*Child Elements:*

**DataColumn :** A list of [CT\_DataColumn](#Section_f5259dad7ba3465092d7ef1d69cc6a70) elements. The **DisplayName** attribute of each **DataColumn** in this listMUST be unique. The **Name** attribute of each **DataColumn** in this listMUST be unique.

1. <xsd:complexType name="CT\_DataColumns">
2. <xsd:sequence>
3. <xsd:element name="DataColumn" minOccurs="1" maxOccurs="unbounded" type="CT\_DataColumn"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_DataRecordset

*Referenced by:* [CT\_DataRecordsets](#Section_7097a58e6c8d428a9c088aadbaf427a1)

The **CT\_DataRecordset** complex type specifies a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2).

*Child Elements:*

**PrimaryKeys :** An optional [CT\_PrimaryKeys](#Section_514c708d371c4dd3907ca4875ec1fb91) element. If the **RowOrder** attribute does not exist or equals 0, this element MUST exist. If the **RowOrder** attribute exists and is equal to 1, this element MUST NOT exist.

**DataColumns :** A [CT\_DataColumns](#Section_2084702f494442eb8bf757fc33425d26) element.

*Attributes:*

**ID :** An xsd:unsignedLong ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.3.21) attribute that specifies the identifier of this recordset. It MUST be unique amongst all the **DataRecordset** child elements of the containing CT\_DataRecordsets.

**ConnectionID :** An xsd:unsignedLong ([XMLSCHEMA2] section 3.3.21) attribute that specifies the identifier of a [CT\_DataConnection](#Section_7e5607b6320948fa9f6dc172500e1dd7) in the document.

**Command :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the [**query**](#gt_37fbc661-f744-48fa-9d8e-f34513cab9c2) string that returns this recordset from a [**data source**](#gt_e091613c-6901-4874-b9b2-27273ead1075). It MUST NOT be empty and MUST use syntax from following table:

|  |  |
| --- | --- |
| Type of data source | Syntax |
| [**OLE DB**](#gt_333f4fb1-4882-48df-bce6-f9961b408f31) | A valid [**SQL**](#gt_37416b0f-ca5b-4fac-96e5-082eb80a3d84) query string as specified in [[MS-ODBCSTR]](%5bMS-ODBCSTR%5d.pdf#Section_13b4e848b36c4b11acced6bf199d5391) |
| [**ODBC**](#gt_7883fa02-8dc0-4154-894f-fe3a7bff153e) | A valid SQL query string as specified in [MS-ODBCSTR] |
| [**List**](#gt_04ce231e-214c-44fd-b7ba-7cc19eee79bf) | A valid SQL query string as specified in [MS-ODBCSTR]. MUST NOT contain a WHERE clause. |
| [**Workbook**](#gt_343c4660-90e1-4d86-b9cc-5007075d9dfe) | The query string MUST be "SheetName=*sheet name*;RangeName=*range name*;", where *sheet name* isthe name of a [**sheet**](#gt_0b356926-f9cb-4dc2-a859-71441d62503d), and *range name* is the name of a [**range**](#gt_c2c93fec-8d3e-45de-8010-c738cc1cea99), given in [**A1**](#gt_950691d0-4b64-4e03-84ae-10d6c872bbdb)[**reference style**](#gt_71d7e4ad-d6b6-49d9-b247-e2ebba1469e9). |
| **Custom** | MUST be same as the **ConnectionString** attribute of the CT\_DataConnection that this recordset uses. |

**Pages :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a comma-separated list of indices of the [drawing pages](#Section_ebba8748c5a54ac7a4191c6a70307640) that contain [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) bound to data in this recordset. The index of each drawing page in this list MUST match the **ID** attribute of one of the **PageMetaData** child elements of the CT\_PagesMetaData in the [ShapeInfo XML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac).

**RowOrder :**  An optional xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies whether this recordset uses the [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) number as the [**primary key**](#gt_e4d8c530-39c1-4fc6-8ccc-8d51a221158d) to bind rows of data in this recordset to shapes in a drawing page. It MUST be 1 or 0. A value of 1 specifies that the row number is used. If the optional **PrimaryKeys** element exists, this attribute MUST NOT exist or MUST be equal to 0. If the **PrimaryKeys** element does not exist, this attribute MUST exist and MUST be equal to 1.

1. <xsd:complexType name="CT\_DataRecordset">
2. <xsd:sequence>
3. <xsd:element name="PrimaryKeys" minOccurs="0" maxOccurs="1" type="CT\_PrimaryKeys"/>
4. <xsd:element name="DataColumns" minOccurs="1" maxOccurs="1" type="CT\_DataColumns"/>
5. </xsd:sequence>
6. <xsd:attribute name="ID" type="xsd:unsignedLong" use="required"/>
7. <xsd:attribute name="ConnectionID" type="xsd:unsignedLong" use="required"/>
8. <xsd:attribute name="Command" type="xsd:string" use="required"/>
9. <xsd:attribute name="Pages" type="xsd:string" use="required"/>
10. <xsd:attribute name="RowOrder" type="xsd:integer" use="optional"/>
11. </xsd:complexType>

##### CT\_DataRecordsets

*Referenced by:* [CT\_Connections](#Section_812ac2f1a2e94ace94f7e6fe2837009c)

The **CT\_DataRecordsets** complex type specifies all the [recordsets](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) of the document.

*Child Elements:*

**DataRecordset :** An optional list of [CT\_DataRecordset](#Section_09771dd2110c41e9ad2b609514a2c79c) elements. CT\_DataRecordset elements are not present if the document does not have recordsets.

1. <xsd:complexType name="CT\_DataRecordsets">
2. <xsd:sequence>
3. <xsd:element name="DataRecordset" minOccurs="0" maxOccurs="unbounded" type="CT\_DataRecordset"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_Connections

*Referenced by:* [Connections](#Section_0b609eff79274eb5abc93b7c5c307d29)

The **CT\_Connections** complex type specifies all the external [**data sources**](#gt_e091613c-6901-4874-b9b2-27273ead1075) in the document.

*Child Elements:*

**DataConnections :** A [CT\_DataConnections](#Section_918bfb0865234923a664b44f66011e43) element.

**DataRecordsets :** A [CT\_DataRecordsets](#Section_7097a58e6c8d428a9c088aadbaf427a1) element.

1. <xsd:complexType name="CT\_Connections">
2. <xsd:sequence>
3. <xsd:element name="DataConnections" minOccurs="1" maxOccurs="1" type="CT\_DataConnections"/>
4. <xsd:element name="DataRecordsets" minOccurs="1" maxOccurs="1" type="CT\_DataRecordsets"/>
5. </xsd:sequence>
6. </xsd:complexType>

### DataGraphic XML Part

This [part](#Section_66eb42ec5ca8459696291233020c701f) specifies information and [formula expressions](#Section_1639b4e843e14d409531adf5c2fe8bbc) used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) the [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) in a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25).

#### Global Elements

##### DataGraphics

A [CT\_DataGraphics](#Section_81e7a61453b0487eacebb684a078cab8) element.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this element.

1. <xsd:element name="DataGraphics" type="CT\_DataGraphics"/>

#### Complex Types

##### CT\_DataGraphicDef

*Referenced by:* [CT\_DataGraphicDefinitions](#Section_abcf02bac3304ecdb03652568f71f0f9)

The **CT\_DataGraphicDef** complex type specifies a reference to information stored in the [DataGraphicDefinition XML Part](#Section_43b985472c2f40eaafc98da28ba27fbb) used to update an [image element](#Section_385a6cb3903347b3acca493e5b363057) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) during [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972).

*Attributes:*

**DefID :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the [CT\_IconSetDef](#Section_fc407930f282426b9d378c8a09a296ad). It MUST be the value of the **DefID** attribute of a CT\_IconSetDef element in the DataGraphicDefinition XML Part. It MUST NOT be an empty string.

**DefShapeID :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the name of the image element data graphic that contains the **Image** elements referencing the [**embedded images**](#gt_c2dd082b-6519-49a2-99e9-77b671621249) corresponding to the CT\_IconSetDef element specified by **DefID**. It MUST NOT be an empty string.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_DataGraphicDef">
2. <xsd:attribute name="DefID" type="xsd:string" use="required"/>
3. <xsd:attribute name="DefShapeID" type="xsd:string" use="required"/>
4. </xsd:complexType>

##### CT\_DataGraphicDefinitions

*Referenced by:* [CT\_DataGraphics\_Shape](#Section_24cbea2f9ad1441a81f41db90fb3dd52)

The **CT\_DataGraphicDefinitions** complex type specifies all the references to information stored in the [DataGraphicDefinition XML Part](#Section_43b985472c2f40eaafc98da28ba27fbb) used to update all the [image element](#Section_385a6cb3903347b3acca493e5b363057) [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) in the [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) during [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972).

*Child Elements:*

**DataGraphicDef :** A list of [CT\_DataGraphicDef](#Section_52cb1fef11f541c5a70f89aec95917a0) elements.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_DataGraphicDefinitions">
2. <xsd:sequence>
3. <xsd:element name="DataGraphicDef" minOccurs="1" maxOccurs="unbounded" type="CT\_DataGraphicDef"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_Geometry

*Referenced by:* [CT\_Sheet](#Section_50dec7c1701c454c90ca55fbec1e724a)

The **CT\_Geometry** complex type specifies how to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) a [geometry element](#Section_23c25866532c499aa07c9dc558139a9b) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075). It MUST exist in a containing CT\_Sheet element that has a corresponding sheet element data graphic in the [ShapeGraphic](#Section_d86834c499d248e2b13ab2c8f016eca0) part of this document.

*Attributes:*

**Index :** An xsd:integer ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.3.13) attribute that specifies the index of the [**XAML**](#gt_891018ae-9381-4d29-82d7-20d6c8537472) Path element in the corresponding [sheet element](#Section_94a21e97d2934aee9999f20030a1ee56) data graphic.

**BoundPts :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the point indices in the XAML Path element that is recalculated. It MUST be ignored when the **Attribute** attribute exists. When the **Attribute** attribute does not exist, It MUST be in the following format:

| Format | Usage |
| --- | --- |
| *pointindex1*,*pointindex2* | For **StartPoint** or **LineSegment** path primitives, as specified in the [geometry path specifications](#Section_78f60135434b407984bdd1062fe933f6) of the associated geometry element data graphic. |
| *pointindex1*,*pointindex2*,*EFlag* | For **ArcSegment** path primitives, as specified in the geometry path specifications of the associated geometry element data graphic. |

Where *pointindex1*, *pointindex2* and *EFlag* are specified in the following table:

| Value | Description |
| --- | --- |
| *pointindex1* | Index of the primitive in the path data, as specified in the geometry path specifications of the associated geometry element data graphic. |
| *pointindex2* | X or Y coordinate. MUST be 0 for the x-coordinate and 1 for the y-coordinate. |
| *EFlag* | A string that specifies that this path data primitive is an **ArcSegment**, as specified in the geometry path specifications of the associated geometry element data graphic. MUST be "E" for Ellipse or "A" for Elliptical Arc. |

**Attribute :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the property of the XAML Path element that the **Formula** attribute is updating. It MUST be ignored when the **BoundPts** attribute exists. When the **BoundPts** attribute does not exist, it MUST be a string from the following table:

| Value | Description |
| --- | --- |
| NoShowGeom | Updates the **Visibility** property of the XAML Path element. If this string exists, MUST NOT show the geometry this element specifies. |
| NoLineGeom | Updates the opacity of the **Stroke** brush of the XAML Path element. If this string exists, the opacity value MUST be set to 0.0. |
| NoFillGeom | Updates the opacity of the **Fill** brush of the XAML Path element. If this string exists, MUST NOT fill the geometry this element specifies. |

**Formula :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc) used to recalculate the XAML Path element. The result token value of the [formula evaluation](#Section_81f0fbe77a314d34b52f5fa9c9380c33) is converted to a string.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_Geometry">
2. <xsd:attribute name="Index" type="xsd:integer" use="required"/>
3. <xsd:attribute name="BoundPts" type="xsd:string" use="optional"/>
4. <xsd:attribute name="Attribute" type="xsd:string" use="required"/>
5. <xsd:attribute name="Formula" type="xsd:string" use="required"/>
6. </xsd:complexType>

##### CT\_Sheet

*Referenced by:* [CT\_Formulas](#Section_7f2ff83ae58b49b2a8f45e273ba8a371)

The **CT\_Sheet** complex type specifies information about the [formula expressions](#Section_1639b4e843e14d409531adf5c2fe8bbc) used to update [sheet element](#Section_94a21e97d2934aee9999f20030a1ee56), [text element](#Section_b061c2bde75d48edb51047479b51c6ec) and [formatting element](#Section_b08c0179146b4b2dbdce35c55c7f7737) [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) during [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972).

When any attribute among the following is specified: **LocPinX**, **LocPinY**, **PinX** or **PinY**, then all of the other attributes of that list MUST be specified, to allow the [shape transform](#Section_c36eb41e53144ecb808acfc9ccd917be) of a sheet element data graphic to be fully specified.

When any attribute among the following is specified: **TextLocPinX**, **TextLocPinY**, **TextPinX** or **TextPinY** then all of the other attributes of that list MUST be specified, to allow the **Render transform** of the [glyph canvas](#Section_4588ece693c04e5b8c803e0870294bd2) to be fully specified.

When any attribute among the following is specified: **FillBackground** or **FillForeground** then all of the other attributes of that list MUST be specified.

*Child Elements:*

**Geometry :** An optional list of [CT\_Geometry](#Section_984113db7f1a430fb5b90f70e3593b50) elements. CT\_Geometry elements exist only if there are [geometry element](#Section_23c25866532c499aa07c9dc558139a9b) data graphics that correspond to the containing CT\_Sheet element.

*Attributes:*

**Name :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the name of the corresponding sheet element. It MUST be a value from the **Name** attribute of a sheet element in the [ShapeGraphic](#Section_d86834c499d248e2b13ab2c8f016eca0) part.

**Angle :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **RenderTransform** property of the corresponding sheet element data graphic. The default value is "0". The canvas is applied as a rotation around the origin, after the (-LocPinX, -LocPinY) translation is applied.

**FlipX :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **RenderTransform** property of the corresponding sheet element data graphic. The default value is "0". The canvas is applied as a flip transform along the X axis after the rotation is applied.

**FlipY :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **RenderTransform** property of the corresponding sheet element data graphic. The default value is "0". The canvas is applied as a flip transform along the Y axis after the rotation is applied.

**LocPinX :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **RenderTransform** property of the corresponding sheet element data graphic. The default value is "0". LocPinX and **LocPinY** translation is applied to the canvas before all other transforms, by (-LocPinX, -LocPinY).

**LocPinY :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **RenderTransfrom** property of the corresponding sheet element data graphic. The default value is "0".

**PinX :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **RenderTransform** property of the corresponding sheet element data graphic. The default value is "0". PinX and **PinY** translation is applied to the canvas after all other transforms, by (PinX, PinY). For more information, see section 2.3.4.1.1.

**PinY :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **RenderTransform** property of the corresponding sheet element data graphic. The default value is "0".

**HideText :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **Visibility** property of the text contained in the corresponding text element data graphic. If this attribute does not exist then the text visibility is not recalculated.

**FillBackground :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the background color which is part of the **Fill** property in the corresponding formatting element data graphic. If this attribute does not exist then the background color is not recalculated.

**FillBackgroundTrans :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the transparency of the background color which is part of the **Fill** property in the corresponding formatting element data graphic. If this attribute does not exist then the transparency of the background color is not recalculated.

**FillForeground :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the foreground color which is part of the **Fill** property in the corresponding formatting element data graphic. If this attribute does not exist then the foreground color is not recalculated.

**FillForegroundTrans :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the transparency of the foreground color which is part of the **Fill** property in the corresponding formatting element data graphic. If this attribute does not exist then the transparency of the foreground color is not recalculated.

**FillPattern :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **FillPattern** property in the corresponding formatting element data graphic. The result of the [formula evaluation](#Section_81f0fbe77a314d34b52f5fa9c9380c33) for this attribute MUST be an integer value as specified in the following table. If this attribute does not exist, the fill pattern defaults to 1.

| Value | Description |
| --- | --- |
| Less than 0 | MUST be ignored and defaults to 1. |
| 0 | No pattern |
| 1 | Solid pattern |
| Greater than 1 and less than 25 | MUST be ignored and defaults to 1. |
| 25 | Gradient – left to right horizontal |
| 26 | Gradient – center out radial |
| 27 | Gradient - right to left horizontal |
| 28 | Gradient - top to bottom vertical |
| 29 | Gradient - center out vertical |
| 30 | Gradient - bottom to top vertical |
| 31 | Gradient - top left rectangle |
| 32 | Gradient - top right rectangle |
| 33 | Gradient - bottom left rectangle |
| 34 | Gradient - bottom right rectangle |
| 35 | Gradient - center out rectangle |
| 36 | Gradient - top left radial |
| 37 | Gradient - top right radial |
| 38 | Gradient - bottom left radial |
| 39 | Gradient - bottom right radial |
| 40 | Gradient - center out radial |
| Greater than 40 | MUST be ignored and defaults to 0. |

**LineColor :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **Stroke** property in the corresponding formatting element data graphic. If this attribute does not exist then the line color is not recalculated.

**LineColorTrans :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **Stroke** property in the corresponding formatting element data graphic. If this attribute does not exist then the transparency of the line color is not recalculated.

**LineWeight :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the StrokeThickness property in the corresponding formatting element data graphic, as specified in section 1.22.31.1.11 of [[MS-SLXV]](%5bMS-SLXV%5d.pdf). If this attribute does not exist then the line weight is not recalculated.

**TextWidth :** An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **Canvas.RenderTransform** property in the corresponding glyph canvas in the text element data graphic. The default value is "0".

**TextHeight :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **Canvas.RenderTransform** property in the corresponding glyph canvas in the text element data graphic. The default value is "0".

**TextAngle :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **Glyphs.RenderTransform** property of the corresponding glyph canvas in the text element data graphic. The default value is "0". The canvas is applied a rotation around the origin, after the (-TextLocPinX, -TextLocPinY) translation is applied.

**TextLocPinX :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **Glyphs.RenderTransform** property in the corresponding glyph canvas in the text element data graphic. The default value is "0". TextLocPinX and TextLocPinY translation is applied to the canvas before all other transforms, by (-TextLocPinX, -TextLocPinY).

**TextLocPinY :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **Glyphs.RenderTransform** property in the corresponding glyph canvas in the text element data graphic. The default value is "0".

**TextPinX :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **Glyphs.RenderTransform** property in the corresponding glyph canvas in the text element data graphic. The default value is "0". TextPinX and **TextPinY** translation is applied to the canvas after all other transforms, by (TextPinX, TextPinY).

**TextPinY :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the **Glyphs.RenderTransform** property in the corresponding glyph canvas in the text element data graphic. The default value is "0".

**Rounding :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the corner rounding radius of the **Data** attribute in the **Path** element in the sheet element data graphic. If this attribute does not exist then the corner rounding is not recalculated.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_Sheet">
2. <xsd:sequence>
3. <xsd:element name="Geometry" minOccurs="0" maxOccurs="unbounded" type="CT\_Geometry"/>
4. </xsd:sequence>
5. <xsd:attribute name="Name" type="xsd:string"/>
6. <xsd:attribute name="Angle" type="xsd:string" use="optional"/>
7. <xsd:attribute name="FlipX" type="xsd:string" use="optional"/>
8. <xsd:attribute name="FlipY" type="xsd:string" use="optional"/>
9. <xsd:attribute name="LocPinX" type="xsd:string" use="optional"/>
10. <xsd:attribute name="LocPinY" type="xsd:string" use="optional"/>
11. <xsd:attribute name="PinX" type="xsd:string" use="optional"/>
12. <xsd:attribute name="PinY" type="xsd:string" use="optional"/>
13. <xsd:attribute name="HideText" type="xsd:string" use="optional"/>
14. <xsd:attribute name="FillBackground" type="xsd:string" use="optional"/>
15. <xsd:attribute name="FillBackgroundTrans" type="xsd:string" use="optional"/>
16. <xsd:attribute name="FillForeground" type="xsd:string" use="optional"/>
17. <xsd:attribute name="FillForegroundTrans" type="xsd:string" use="optional"/>
18. <xsd:attribute name="FillPattern" type="xsd:string" use="optional"/>
19. <xsd:attribute name="LineColor" type="xsd:string" use="optional"/>
20. <xsd:attribute name="LineColorTrans" type="xsd:string" use="optional"/>
21. <xsd:attribute name="LineWeight" type="xsd:string" use="optional"/>
22. <xsd:attribute name="TextWidth" type="xsd:string" use="optional"/>
23. <xsd:attribute name="TextHeight" type="xsd:string" use="optional"/>
24. <xsd:attribute name="TextAngle" type="xsd:string" use="optional"/>
25. <xsd:attribute name="TextLocPinX" type="xsd:string" use="optional"/>
26. <xsd:attribute name="TextLocPinY" type="xsd:string" use="optional"/>
27. <xsd:attribute name="TextPinX" type="xsd:string" use="optional"/>
28. <xsd:attribute name="TextPinY" type="xsd:string" use="optional"/>
29. <xsd:attribute name="Rounding" type="xsd:string" use="optional"/>
30. </xsd:complexType>

##### CT\_Formulas

*Referenced by:* [CT\_DataGraphics\_Shape](#Section_24cbea2f9ad1441a81f41db90fb3dd52)

The **CT\_Formulas** complex type specifies the [CT\_Sheet](#Section_50dec7c1701c454c90ca55fbec1e724a) elements within the containing CT\_DataGraphics\_Shape element that has properties that are recalculated by [formula expressions](#Section_1639b4e843e14d409531adf5c2fe8bbc).

*Child Elements:*

**Sheet :** A list of CT\_Sheet elements.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_Formulas">
2. <xsd:sequence>
3. <xsd:element name="Sheet" minOccurs="1" maxOccurs="unbounded" type="CT\_Sheet"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_FormulaReferences

*Referenced by:* [CT\_DataGraphics\_Shape](#Section_24cbea2f9ad1441a81f41db90fb3dd52)

The **CT\_FormulaReferences** complex type specifies a list of formula references. A formula reference is a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc) that is shared and referenced by other formula expressions within the containing CT\_DataGraphics\_Shape element. Each formula reference is specified as a string attribute. The name of the attribute is a formula identifier. The formula identifier MUST be unique amongst all formula identifiers in the CT\_FormulaReferences element for the containing CT\_DataGraphics\_Shape element. The value of the attribute MUST be a formula expression.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_FormulaReferences">
2. <xsd:anyAttribute namespace="##local" processContents="skip"/>
3. </xsd:complexType>

##### CT\_DataGraphics\_Shape

*Referenced by:* [CT\_DataGraphics\_Page](#Section_1656fc08de3a4c088c7a5ed89568c0b6)

The **CT\_DataGraphics\_Shape** complex type specifies information used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) the [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) associated with a [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Child Elements:*

**DataGraphicDefinitions :** An optional [CT\_DataGraphicDefinitions](#Section_abcf02bac3304ecdb03652568f71f0f9) element. It MUST exist if **Formulas** does not exist.

**Formulas :** An optional [CT\_Formulas](#Section_7f2ff83ae58b49b2a8f45e273ba8a371) element. It MUST exist if **DataGraphicDefinitions** does not exist.

**FormulaReferences :** An optional [CT\_FormulaReferences](#Section_c376a1af36d2469686af16d31ea7afeb) element. It MUST NOT exist if **Formulas** does not exist.

*Attributes:*

**ID :** An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies an identifier for the data graphic. It MUST NOT be an empty string and MUST be unique amongst the **ID** attributes of the CT\_DataGraphics\_Shape elements in this XML part.

**ShapeName :** An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the identifier of the shape that references this data graphic. It MUST match the **Name** attribute of a **Canvas** element in the [ShapeGraphic Part](#Section_d86834c499d248e2b13ab2c8f016eca0).

**DataRefs :** An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the external data that is bound to this data graphic. If this attribute does not exist or if it is an empty string then the external data bound to this data graphic is unknown. If a string for this attribute does exist it MUST have the following format:

1. rowID1,colID1;rowID2,colID2;…;rowIDn,colIDn

| Name | Description |
| --- | --- |
| rowID | An integer that specifies the [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) in the [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) that this shape is bound to. MUST be the value of the **BindingID** attribute of a [CT\_Binding](#Section_5ad7ddf78438400abe979591e0cca24a) element in the [DataBinding XML Part](#Section_311514cd8ef444c9ab42562671155333). |
| colID | A string that specifies the [**field**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc) in the recordset that this shape is bound to. MUST be the value of the **Name** attribute of a [CT\_DataColumn](#Section_f5259dad7ba3465092d7ef1d69cc6a70) element in the [DataConnection XML Part](#Section_c489eefb7a014941bc37a4d332e66dc8). |

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_DataGraphics\_Shape">
2. <xsd:sequence>
3. <xsd:element name="DataGraphicDefinitions" minOccurs="0" maxOccurs="1" type="CT\_DataGraphicDefinitions"/>
4. <xsd:element name="Formulas" minOccurs="0" maxOccurs="1" type="CT\_Formulas"/>
5. <xsd:element name="FormulaReferences" minOccurs="0" maxOccurs="1" type="CT\_FormulaReferences"/>
6. </xsd:sequence>
7. <xsd:attribute name="ID" type="xsd:string" use="required"/>
8. <xsd:attribute name="ShapeName" type="xsd:string" use="required"/>
9. <xsd:attribute name="DataRefs" type="xsd:string" use="optional"/>
10. </xsd:complexType>

##### CT\_DataGraphics\_Page

*Referenced by:* [CT\_DataGraphics](#Section_81e7a61453b0487eacebb684a078cab8)

The **CT\_DataGraphics\_Page** complex type specifies information used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) all the [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) in a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640).

*Child Elements:*

**Shape :** An optional list of [CT\_DataGraphics\_Shape](#Section_24cbea2f9ad1441a81f41db90fb3dd52) elements. CT\_DataGraphics\_Shape elements are not present if the [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) of the drawing page are not associated with data graphics.

*Attributes:*

**PageName :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the name of the drawing page.

**ViewTransform :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a geometric transform used to convert the Cartesian coordinates of the graphic elements in the [ShapeGraphic Part](#Section_d86834c499d248e2b13ab2c8f016eca0) to the coordinate system of the [**view**](#gt_3f793b0b-9509-4df0-89f9-92f07954beb8) that displays the drawing page. It MUST have the following format:

1. m11 m12 m21 m22 dx dy

m11, m12, m21, m22, dx, dy are float values that represent a 3-by-3 affine transformation matrix as specified in the following table. All values MUST be separated by single spaces.

|  | Column 1 | Column 2 | Column 3 |
| --- | --- | --- | --- |
| Row 1 | m11 | m12 | 0 |
| Row 2 | m12 | m22 | 0 |
| Row 3 | dx | dy | 1 |

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_DataGraphics\_Page">
2. <xsd:sequence>
3. <xsd:element name="Shape" minOccurs="0" maxOccurs="unbounded" type="CT\_DataGraphics\_Shape"/>
4. </xsd:sequence>
5. <xsd:attribute name="PageName" type="xsd:string"/>
6. <xsd:attribute name="ViewTransform" type="xsd:string"/>
7. </xsd:complexType>

##### CT\_DataGraphics

*Referenced by:* [DataGraphics](#Section_9fde0d0eb7a949bba229ce9a7b000d87)

The **CT\_DataGraphics** complex type specifies information used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) all the [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) in a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25).

*Child Elements:*

**Page :** A list of [CT\_DataGraphics\_Page](#Section_1656fc08de3a4c088c7a5ed89568c0b6) elements.

*Attributes:*

**ColorTable :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the list of colors used to render data graphics. It MUST have the following format:

1. R1,G1,B1 R2,G2,B2 … Rn,Gn,Bn

Each color MUST be specified using the [**RGB**](#gt_2c716d3a-e60b-4e52-bbb0-2fdeb298003b) color model as described in the following table. The components of a color MUST be separated by commas. The colors MUST be separated by single spaces.

| Name | Description |
| --- | --- |
| Rn | An integer value that specifies the red component of the color. MUST be greater than or equal to 0 and less than or equal to 255. |
| Gn | An integer value that specifies the green component of the color. MUST be greater than or equal to 0 and less than or equal to 255. |
| Bn | An integer value that specifies the blue component of the color. MUST be greater than or equal to 0 and less than or equal to 255. |

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_DataGraphics">
2. <xsd:sequence>
3. <xsd:element name="Page" minOccurs="1" maxOccurs="unbounded" type="CT\_DataGraphics\_Page"/>
4. </xsd:sequence>
5. <xsd:attribute name="ColorTable" type="xsd:string"/>
6. </xsd:complexType>

### DataGraphicDefinition XML Part

This [part](#Section_66eb42ec5ca8459696291233020c701f) specifies information used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) the [image element](#Section_385a6cb3903347b3acca493e5b363057) [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) in a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25).

#### Global Elements

##### DataGraphicDefs

A [CT\_DataGraphicDefs](#Section_c4e5d982bba44fc19128daa4f175f249) element.

1. <xsd:element name="DataGraphicDefs" type="CT\_DataGraphicDefs"/>

#### Complex Types

##### CT\_IconSetRule

*Referenced by:* [CT\_IconSetDef](#Section_fc407930f282426b9d378c8a09a296ad)

The **CT\_IconSetRule** complex type specifies a [**rule**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) for an [image element](#Section_385a6cb3903347b3acca493e5b363057) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) that contains an [**embedded image**](#gt_c2dd082b-6519-49a2-99e9-77b671621249). The rule is specified by the **Formula** attribute and the **Image** attribute to be rendered when the condition specified in **Formula** has been met.

*Attributes:*

**Name :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies an identifier for the rule. It MUST NOT be an empty string and MUST be unique amongst the identifiers specified by the **Name** attribute of the CT\_IconSetRule elements in this XML part.

**Formula :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that is a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc) specifying a condition to be evaluated. The condition is met if the result of the [formula evaluation](#Section_81f0fbe77a314d34b52f5fa9c9380c33) is not equal to -1.

**Image :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the name of the embedded image to be rendered when the condition specified by **Formula** has been met.

1. <xsd:complexType name="CT\_IconSetRule">
2. <xsd:attribute name="Name" type="xsd:string" use="required"/>
3. <xsd:attribute name="Formula" type="xsd:string" use="required"/>
4. <xsd:attribute name="Image" type="xsd:string" use="required"/>
5. </xsd:complexType>

##### CT\_IconSetDef

*Referenced by:* [CT\_DataGraphicDefs](#Section_c4e5d982bba44fc19128daa4f175f249)

The **CT\_IconSetDef** complex type specifies information used to update an [image element](#Section_385a6cb3903347b3acca493e5b363057) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) that contains an embedded image during [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972).

*Child Elements:*

**IconSetRule :** A list of [CT\_IconSetRule](#Section_3d0b2cb1726b4aa6908159af8bfe6a3b) elements.

*Attributes:*

**DefID :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies an identifier of the CT\_IconSetDef. It MUST NOT be an empty string and MUST be unique amongst the identifiers specified by the **DefID** attribute of the CT\_IconSetDef elements in this XML part.

1. <xsd:complexType name="CT\_IconSetDef">
2. <xsd:sequence>
3. <xsd:element name="IconSetRule" minOccurs="1" maxOccurs="unbounded" type="CT\_IconSetRule"/>
4. </xsd:sequence>
5. <xsd:attribute name="DefID" type="xsd:string" use="required"/>
6. </xsd:complexType>

##### CT\_DataGraphicDefs

*Referenced by:* [DataGraphicDefs](#Section_c32b94b9d8cd4f96825939486e776195)

The **CT\_DataGraphicDefs** complex type specifies information used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) all [image element](#Section_385a6cb3903347b3acca493e5b363057) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) in the [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25).

*Child Elements:*

**IconSetDef :** An optional list of [CT\_IconSetDef](#Section_fc407930f282426b9d378c8a09a296ad) elements. CT\_IconSetDef elements are not present if the document does not have image element data graphic that contain [**embedded images**](#gt_c2dd082b-6519-49a2-99e9-77b671621249).

1. <xsd:complexType name="CT\_DataGraphicDefs">
2. <xsd:sequence>
3. <xsd:element name="IconSetDef" minOccurs="0" maxOccurs="unbounded" type="CT\_IconSetDef"/>
4. </xsd:sequence>
5. </xsd:complexType>

### ShapeInfo XML Part

This [part](#Section_66eb42ec5ca8459696291233020c701f) specifies information about the [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) on a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) in the document.

#### Global Elements

##### Page

A [CT\_Page](#Section_020589d88a1f420c97805dd19dac1cf7) element.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this element.

1. <xsd:element name="Page" type="CT\_Page"/>

#### Complex Types

##### CT\_PageInfo

*Referenced by:* [CT\_Pages](#Section_45b420de6cd2463e95d3b7bc083a0b18)

The **CT\_PageInfo** complex type specifies the name of a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) in the document.

*Attributes:*

**Name :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the name of a drawing page. It MUST be unique amongst all the drawing pages specified by the CT\_PageInfo child elements of CT\_Pages.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_PageInfo">
2. <xsd:attribute name="Name" type="xsd:string"/>
3. </xsd:complexType>

##### CT\_Pages

*Referenced by:* [CT\_Page](#Section_020589d88a1f420c97805dd19dac1cf7)

The **CT\_Pages** complex type specifies the names of all the [drawing pages](#Section_ebba8748c5a54ac7a4191c6a70307640) in the document.

*Child Elements:*

**PageInfo :** A list of [CT\_PageInfo](#Section_3eddc14246c1484497128d5790bdb49d) elements.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_Pages">
2. <xsd:sequence>
3. <xsd:element name="PageInfo" minOccurs="1" maxOccurs="unbounded" type="CT\_PageInfo"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_ShapeData

*Referenced by:* [CT\_ShapeDataItems](#Section_67db1c6345344b4e86ae8bc52c53deec)

The **CT\_ShapeData** complex type specifies a [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) item.

*Attributes:*

**Name :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the label of the shape data item.

**FormattedValue :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the value formatted for display. It MUST NOT be an empty string.

**Value :** An attribute that specifies the value of the shape data item. The type of this attribute is specified by **Type**.

**Format :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the [vFormatString](#Section_3d8a401ea64247d082ad4cfbdd2331b9) used to format the value for display.

**Type :**  An xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the [**data type**](#gt_0d327bc0-a461-4723-8beb-d935fdab4895) of the shape data item. It MUST be a value from the following table:

| Value | Meaning |
| --- | --- |
| 0 | A string value. The type of **Value** MUST be string. |
| 1 | A fixed list of values represented as a string. The type of **Value** MUST be string. |
| 2 | A numerical value. The type of **Value** MUST be double. |
| 3 | A [**Boolean**](#gt_1d79d7a7-ba2c-4b34-931c-7ba8057c87b2) value. The type of **Value** MUST be string and **Value** MUST be "TRUE" or "FALSE". |
| 4 | A variable list of values represented as a string. The type of **Value** MUST be string. |
| 5 | An OLE automation date value as specified in [[MS-OAUT]](%5bMS-OAUT%5d.pdf#Section_bbb05720f72445c78d17f83c3d1a3961) section [2.2.25](http://msdn.microsoft.com/en-us/library/35c9bf2d-b8e8-4d7d-a50f-367da0d99fce/). The type of **Value** MUST be double. |
| 6 | A duration value. The type of **Value** MUST be double. |
| 7 | A currency value. The type of **Value** MUST be double. |

**LangID :**  An xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24) used to format the value.

**UnitLabel :**  An xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the [vUnitLabel](#Section_b8166c01556f4257b5bcc081923ee37d) used to format the value.

**CalendarID :**  An optional xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the [vCalendar](#Section_4c2dd2c033874a6c99af9b277c89b411) used to format a date value. The default value is 0. It MUST be ignored if **Type** is not equal to 5.

**CurrencyID :**  An optional xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the [vCurrencyID](#Section_b13bb6692bc04291b31c4532158fa675) used to format a currency value. The default value is 0. It MUST be ignored if **Type** is not equal to 7.

**ServerAction :**  An optional xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the [vServerAction](#Section_094c818d877844b881c7dd957e2aaf7f) for the formatted value. The default value is 0.

**Unit :**  An xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the [vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789) that **Value** is converted from before formatting it for display.

**DisplayUnit :**  An optional xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the vNumAny that **Value** is converted to before formatting it for display. The default is the value of **Unit**.

**BindingID :**  An optional xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the [CT\_Binding](#Section_5ad7ddf78438400abe979591e0cca24a) identifier for this shape data item. It MUST be one of the values specified by the **BindingID** attribute of the CT\_Binding elements in the [DataBinding XML Part](#Section_311514cd8ef444c9ab42562671155333). If this attribute does not exist then this shape data item is not bound to data.

**HyperlinkID :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the [CT\_Hyperlink](#Section_0cd8a3b16777487ea09dc7f31460d3e0) identifier for this shape data item. It MUST be the value of the **Name** attribute of a CT\_Hyperlink child element of the corresponding [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0) element in this XML part. If this attribute does not exist then this shape data item is not associated with a hyperlink.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_ShapeData">
2. <xsd:attribute name="Name" type="xsd:string" use="required"/>
3. <xsd:attribute name="FormattedValue" type="xsd:string" use="required"/>
4. <xsd:attribute name="Value" type="xsd:string" use="required"/>
5. <xsd:attribute name="Format" type="xsd:string" use="required"/>
6. <xsd:attribute name="Type" use="required">
7. <xsd:simpleType>
8. <xsd:restriction base="xsd:integer">
9. <xsd:enumeration value="0"/>
10. <xsd:enumeration value="1"/>
11. <xsd:enumeration value="2"/>
12. <xsd:enumeration value="3"/>
13. <xsd:enumeration value="4"/>
14. <xsd:enumeration value="5"/>
15. <xsd:enumeration value="6"/>
16. <xsd:enumeration value="7"/>
17. </xsd:restriction>
18. </xsd:simpleType>
19. </xsd:attribute>
20. <xsd:attribute name="LangID" type="xsd:integer" use="required"/>
21. <xsd:attribute name="UnitLabel" type="xsd:integer" use="required"/>
22. <xsd:attribute name="CalendarID" type="xsd:integer" use="optional"/>
23. <xsd:attribute name="CurrencyID" type="xsd:integer" use="optional"/>
24. <xsd:attribute name="ServerAction" type="xsd:integer" use="optional"/>
25. <xsd:attribute name="Unit" type="xsd:integer" use="required"/>
26. <xsd:attribute name="DisplayUnit" type="xsd:integer" use="optional"/>
27. <xsd:attribute name="BindingID" type="xsd:integer" use="optional"/>
28. <xsd:attribute name="HyperlinkID" type="xsd:string" use="optional"/>
29. </xsd:complexType>

##### CT\_ShapeDataItems

*Referenced by:* [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0)

The **CT\_ShapeDataItems** complex type specifies the [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) items associated with a [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Child Elements:*

**ShapeData :** A list of [CT\_ShapeData](#Section_a90cbb53af5347ae9da20afddd00cf2f) elements.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_ShapeDataItems">
2. <xsd:sequence>
3. <xsd:element name="ShapeData" minOccurs="1" maxOccurs="unbounded" type="CT\_ShapeData"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_Hyperlink

*Referenced by:* [CT\_Hyperlinks](#Section_155b1e53317a4af1b82ca10f8e346210)

The **CT\_Hyperlink** complex type specifies a [hyperlink](#Section_1fc5829872d4478d9d86b282342940cc) associated with a [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Attributes:*

**Name :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies an identifier for the hyperlink. It MUST NOT be an empty string and MUST be unique amongst the identifiers specified by the **Name** attribute of the CT\_Hyperlink elements in this XML part.

**Value :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the [**URI**](#gt_e18af8e8-01d7-4f91-8a1e-0fb21b191f95) of an external resource referenced by the hyperlink. It MUST NOT be an empty string. If this attribute exists, **SubAddress**, **SubAddressShape**, and **Zoom** MUST be ignored. If this attribute does not exist, **SubAddress** MUST exist.

**Description :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the description of the hyperlink. If this attribute does not exist then the hyperlink does not have a description.

**SubAddress :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the identifier of the [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) referenced by this hyperlink within the document. It MUST NOT be an empty string. If **SubAddress** does not exist, **SubAddressShape** and **Zoom** MUST be ignored.

**SubAddressShape :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the identifier of the shape referenced by this hyperlink within the drawing page specified by **SubAddress**. It MUST NOT be an empty string. If **SubAddressShape** does not exist, the hyperlink references the entire drawing page.

**Zoom :**  An optional xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the [**zoom level**](#gt_bfb33471-a018-422b-bc63-177c8bc1831f) for the drawing page specified by **SubAddress**. If this attribute does not exist then the zoom level is 100%. If this attribute does exist then it MUST be a value from the following table:

| Value | Description |
| --- | --- |
| -2 | The drawing page is zoomed to fit the width of the [**view**](#gt_3f793b0b-9509-4df0-89f9-92f07954beb8). |
| -1 | The drawing page is zoomed to fit in the view. |
| n > 0 | The drawing page is zoomed to n%. n MUST be greater than 0. |

**Default :** An optional xsd:boolean ([XMLSCHEMA2] section 3.2.2) attribute that specifies the default hyperlink associated with the shape. It MUST exist if and only if this hyperlink is the default. Each shape MUST NOT have more than one default hyperlink. If this attribute exists, the value MUST be 1.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_Hyperlink">
2. <xsd:attribute name="Name" type="xsd:string" use="required"/>
3. <xsd:attribute name="Value" type="xsd:string" use="optional"/>
4. <xsd:attribute name="Description" type="xsd:string" use="optional"/>
5. <xsd:attribute name="SubAddress" type="xsd:string" use="optional"/>
6. <xsd:attribute name="SubAddressShape" type="xsd:string" use="optional"/>
7. <xsd:attribute name="Zoom" use="optional">
8. <xsd:simpleType>
9. <xsd:union>
10. <xsd:simpleType>
11. <xsd:restriction base="xsd:integer">
12. <xsd:minExclusive value="0"/>
13. </xsd:restriction>
14. </xsd:simpleType>
15. <xsd:simpleType>
16. <xsd:restriction base="xsd:integer">
17. <xsd:enumeration value="-2"/>
18. <xsd:enumeration value="-1"/>
19. </xsd:restriction>
20. </xsd:simpleType>
21. </xsd:union>
22. </xsd:simpleType>
23. </xsd:attribute>
24. <xsd:attribute name="Default" type="xsd:boolean" use="optional"/>
25. </xsd:complexType>

##### CT\_Hyperlinks

*Referenced by:* [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0)

The **CT\_Hyperlinks** complex type specifies the [hyperlinks](#Section_1fc5829872d4478d9d86b282342940cc) associated with a [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Child Elements:*

**Hyperlink :** A list of [CT\_Hyperlink](#Section_0cd8a3b16777487ea09dc7f31460d3e0) elements.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_Hyperlinks">
2. <xsd:sequence>
3. <xsd:element name="Hyperlink" minOccurs="1" maxOccurs="unbounded" type="CT\_Hyperlink"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_ShapeInfo

*Referenced by:* [CT\_Page](#Section_020589d88a1f420c97805dd19dac1cf7)

The **CT\_ShapeInfo** complex type specifies information about a [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Child Elements:*

**ShapeDataItems :** An optional [CT\_ShapeDataItems](#Section_67db1c6345344b4e86ae8bc52c53deec) element. CT\_ShapeDataItems element is not present if the shape is not associated with [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) items.

**Hyperlinks :** An optional [CT\_Hyperlinks](#Section_155b1e53317a4af1b82ca10f8e346210) element. CT\_Hyperlinks element is not present if the shape is not associated with [hyperlinks](#Section_1fc5829872d4478d9d86b282342940cc).

*Attributes:*

**Name :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the identifier of the shape. It MUST match exactly the **Name** of the [**XAML**](#gt_891018ae-9381-4d29-82d7-20d6c8537472) element that represents the shape as specified in the [ShapeGraphic Part](#Section_d86834c499d248e2b13ab2c8f016eca0). It MUST NOT be an empty string and MUST be unique amongst all the identifiers specified by the **Name** attribute of the CT\_ShapeInfo elements in this XML part.

**DisplayName :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a display name for the shape.

**Guid :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a [**GUID**](#gt_f49694cc-c350-462d-ab8e-816f0103c6c1) for the shape. If this attribute does not exist then the shape does not have a GUID.

**Layout :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies geometric information for the [**bounding rectangle**](#gt_e349a155-0fb9-4ba1-abd4-e363e21abcd1) of the shape. The structure of this information is specified in the following table:

| Name | Description |
| --- | --- |
| type | A string that specifies the shape type. MUST be "2D" and specifies a bi-dimensional shape. |
| bounds | A structure that specifies the minimum bounding rectangle for the shape as specified in the following table: |

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

The coordinates of the bounding rectangle MUST be relative to the top, left corner of the [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) that contains the shape. All the values MUST be specified in [**pixels**](#gt_fe932605-d3bc-442f-9239-594676bce6ba). The string MUST be formatted using the JavaScript Object Notation as specified in [[RFC4627]](https://go.microsoft.com/fwlink/?LinkId=140879) as follows:

1. {"type":"2D","bounds":{"x":**x-val**,"y":**y-val**,"width":**width-val**,"height":**height-val**}}

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

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_ShapeInfo">
2. <xsd:sequence>
3. <xsd:element name="ShapeDataItems" minOccurs="0" maxOccurs="1" type="CT\_ShapeDataItems"/>
4. <xsd:element name="Hyperlinks" minOccurs="0" maxOccurs="1" type="CT\_Hyperlinks"/>
5. </xsd:sequence>
6. <xsd:attribute name="Name" type="xsd:string" use="required"/>
7. <xsd:attribute name="DisplayName" type="xsd:string"/>
8. <xsd:attribute name="Guid" type="xsd:string" use="optional"/>
9. <xsd:attribute name="Layout" type="xsd:string" use="required"/>
10. </xsd:complexType>

##### CT\_Page

*Referenced by:* [Page](#Section_8add6b5b5d1747ec84418fd27c5b7f77)

The **CT\_Page** complex type specifies [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) information within a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) in the document.

*Child Elements:*

**Pages :** A [CT\_Pages](#Section_45b420de6cd2463e95d3b7bc083a0b18) element.

**ShapeInfo :** An optional list of [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0) elements. CT\_ShapeInfo elements are not present if there are no shapes in the drawing page.

*Attributes:*

**Name :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the name of the drawing page. It MUST NOT be an empty string. It MUST match the **Name** attribute of one of the CT\_PageMetaData elements in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e). It MUST match a name in the **Name** attribute of one of the **PageInfo** child elements of the **Pages** element.

**Zoom :** An xsd:double ([XMLSCHEMA2] section 3.2.5) attribute that specifies the [**zoom level**](#gt_bfb33471-a018-422b-bc63-177c8bc1831f) at which the drawing page is initially displayed. It MUST be a value from the following table:

| Value | Description |
| --- | --- |
| -2.0 | The drawing page is zoomed to fit the width of the [**view**](#gt_3f793b0b-9509-4df0-89f9-92f07954beb8). |
| -1.0 | The drawing page is zoomed to fit the entire drawing page in the view. |
| Any value greater than or equal to 0.0 | The drawing page is zoomed to **Zoom**\*100%. |

**OffsetX :**  An xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the x-coordinate of the point on the drawing page that is centered in the view when the drawing page is initially displayed. The value is relative to the top, left corner of the drawing page and is expressed in the coordinate system of the view that displays the drawing page.

**OffsetY :**  An xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the y-coordinate of the point on the drawing page that is centered in the view when the drawing page is initially displayed. The value is relative to the top, left corner of the drawing page and is expressed in the coordinate system of the view that displays the drawing page.

**DefaultUnitsText :**  An xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the default typographic measurement unit to be used in [formula evaluations](#Section_81f0fbe77a314d34b52f5fa9c9380c33) for this drawing page if none is explicitly specified in a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc). It MUST be one of the following values:

| Value | Meaning |
| --- | --- |
| 48 | Picas and points |
| 49 | Picas and points |
| 50 | Points |
| 51 | Picas |
| 52 | Ciceros and didots |
| 53 | Didots |
| 54 | Ciceros |

**DefaultUnitsAngle :**  An xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the default angle measurement unit to be used in formula evaluations for this drawing page if none is explicitly specified in a formula expression. It MUST be one of the following values:

| Value | Meaning |
| --- | --- |
| 80 | Decimal degrees |
| 81 | Decimal degrees |
| 82 | Degrees-minutes-seconds |
| 83 | Radians |
| 84 | Minutes-seconds |
| 85 | Seconds |

**DefaultUnitsDuration :**  An xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the default duration measurement unit to be used in formula evaluations for this drawing page if none is explicitly specified in a formula expression. It MUST be one of the following values:

| Value | Meaning |
| --- | --- |
| 42 | Days |
| 43 | Weeks |
| 44 | Days |
| 45 | Hours |
| 46 | Minutes |
| 47 | Seconds |

**DefaultUnitsPage :**  An xsd:integer ([XMLSCHEMA2] section 3.3.13) attribute that specifies the default length measurement in page units to be used in formula evaluations for this drawing page if none is explicitly specified in a formula expression. It MUST be one of the following values:

| Value | Meaning |
| --- | --- |
| 43 | Weeks |
| 44 | Days |
| 45 | Hours |
| 46 | Minutes |
| 47 | Seconds |
| 50 | Points |
| 51 | Picas |
| 53 | Didots |
| 54 | Ciceros |
| 63 | Inches |
| 64 | Inches |
| 65 | Inches |
| 66 | Feet |
| 67 | Feet and inches |
| 68 | Miles (decimal) |
| 69 | Centimeters |
| 70 | Millimeters |
| 71 | Meters |
| 72 | Kilometers |
| 73 | Inches (fractional) |
| 74 | Miles (fractional) |
| 75 | Yards |
| 76 | Nautical miles |

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_Page">
2. <xsd:sequence>
3. <xsd:element name="Pages" minOccurs="1" maxOccurs="1" type="CT\_Pages"/>
4. <xsd:element name="ShapeInfo" minOccurs="0" maxOccurs="unbounded" type="CT\_ShapeInfo"/>
5. </xsd:sequence>
6. <xsd:attribute name="Name" type="xsd:string" use="required"/>
7. <xsd:attribute name="Zoom" use="required">
8. <xsd:simpleType>
9. <xsd:union>
10. <xsd:simpleType>
11. <xsd:restriction base="xsd:double">
12. <xsd:enumeration value="-2"/>
13. <xsd:enumeration value="-1"/>
14. </xsd:restriction>
15. </xsd:simpleType>
16. <xsd:simpleType>
17. <xsd:restriction base="xsd:double">
18. <xsd:minInclusive value="0"/>
19. </xsd:restriction>
20. </xsd:simpleType>
21. </xsd:union>
22. </xsd:simpleType>
23. </xsd:attribute>
24. <xsd:attribute name="OffsetX" type="xsd:integer" use="required"/>
25. <xsd:attribute name="OffsetY" type="xsd:integer" use="required"/>
26. <xsd:attribute name="DefaultUnitsText" use="required">
27. <xsd:simpleType>
28. <xsd:restriction base="xsd:integer">
29. <xsd:enumeration value="48"/>
30. <xsd:enumeration value="49"/>
31. <xsd:enumeration value="50"/>
32. <xsd:enumeration value="51"/>
33. <xsd:enumeration value="52"/>
34. <xsd:enumeration value="53"/>
35. <xsd:enumeration value="54"/>
36. </xsd:restriction>
37. </xsd:simpleType>
38. </xsd:attribute>
39. <xsd:attribute name="DefaultUnitsAngle" use="required">
40. <xsd:simpleType>
41. <xsd:restriction base="xsd:integer">
42. <xsd:enumeration value="80"/>
43. <xsd:enumeration value="81"/>
44. <xsd:enumeration value="82"/>
45. <xsd:enumeration value="83"/>
46. <xsd:enumeration value="84"/>
47. <xsd:enumeration value="85"/>
48. </xsd:restriction>
49. </xsd:simpleType>
50. </xsd:attribute>
51. <xsd:attribute name="DefaultUnitsDuration" use="required">
52. <xsd:simpleType>
53. <xsd:restriction base="xsd:integer">
54. <xsd:enumeration value="42"/>
55. <xsd:enumeration value="43"/>
56. <xsd:enumeration value="44"/>
57. <xsd:enumeration value="45"/>
58. <xsd:enumeration value="46"/>
59. <xsd:enumeration value="47"/>
60. </xsd:restriction>
61. </xsd:simpleType>
62. </xsd:attribute>
63. <xsd:attribute name="DefaultUnitsPage" use="required">
64. <xsd:simpleType>
65. <xsd:restriction base="xsd:integer">
66. <xsd:enumeration value="43"/>
67. <xsd:enumeration value="44"/>
68. <xsd:enumeration value="45"/>
69. <xsd:enumeration value="46"/>
70. <xsd:enumeration value="47"/>
71. <xsd:enumeration value="50"/>
72. <xsd:enumeration value="51"/>
73. <xsd:enumeration value="53"/>
74. <xsd:enumeration value="54"/>
75. <xsd:enumeration value="63"/>
76. <xsd:enumeration value="64"/>
77. <xsd:enumeration value="65"/>
78. <xsd:enumeration value="66"/>
79. <xsd:enumeration value="67"/>
80. <xsd:enumeration value="68"/>
81. <xsd:enumeration value="69"/>
82. <xsd:enumeration value="70"/>
83. <xsd:enumeration value="71"/>
84. <xsd:enumeration value="72"/>
85. <xsd:enumeration value="73"/>
86. <xsd:enumeration value="74"/>
87. <xsd:enumeration value="75"/>
88. <xsd:enumeration value="76"/>
89. </xsd:restriction>
90. </xsd:simpleType>
91. </xsd:attribute>
92. </xsd:complexType>

### ShapeOutline XML Part

This [part](#Section_66eb42ec5ca8459696291233020c701f) specifies geometric outline information for the [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) in a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) of the document. The outlines enable interactivity with the shapes on the drawing page. These outlines are in the image map format. For more details about image maps see [[HTML]](https://go.microsoft.com/fwlink/?LinkId=89880) section 13.6.1.

#### Global Elements

##### Page

A [CT\_ShapeOutline\_Page](#Section_dec60ada5c5f4622b28b7c486527b022) element.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment defines the contents of this element.

1. <xsd:element name="Page" type="CT\_ShapeOutline\_Page"/>

#### Complex Types

##### CT\_Path

*Referenced by:* [CT\_Shape](#Section_6defd531916c4391bbb80bc71d0c0c5c)

The **CT\_Path** complex type specifies geometric outline information for a [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Attributes:*

**Shape :**  An optional xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the type of outline. It MUST be "poly". The default value is "poly".

**Data :**  An xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the Cartesian coordinates of the outline. It MUST have the following format:

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

xn and yn MUST be integer values and they MUST occur in pairs. Each xn,yn pair specifies the coordinates of a point that defines one vertex of the outline. The outline is formed by connecting the points in the order they appear in this string attribute. There MUST be at least 4 points. The first and the last point MUST be the same. The coordinates MUST be measured in [**pixels**](#gt_fe932605-d3bc-442f-9239-594676bce6ba) relative to the top, left corner of the [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) that contains the shape. All values MUST be separated by commas.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment defines the contents of this complex type.

1. <xsd:complexType name="CT\_Path">
2. <xsd:attribute name="Shape" type="xsd:string" fixed="poly"/>
3. <xsd:attribute name="Data" use="required">
4. <xsd:simpleType>
5. <xsd:restriction base="xsd:string">
6. <xsd:pattern value="\d\*(,\d\*){3,}"/>
7. </xsd:restriction>
8. </xsd:simpleType>
9. </xsd:attribute>
10. </xsd:complexType>

##### CT\_Shape

*Referenced by:* [CT\_Shapes](#Section_f63911d856514018837174226fb2b6ab)

The **CT\_Shape** complex type specifies geometric outline information for the [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) identified by the **Name** attribute.

*Child Elements:*

**Path :** A list of [CT\_Path](#Section_37b1961441894431bf9967fa8ad6e2f7) elements.

*Attributes:*

**Name :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the identifier of the shape. It MUST be one of the identifiers specified by the **Name** attribute of the [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0) elements in the [ShapeInfo XML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac).

**Layout :** This attribute is unused and MUST be ignored.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment defines the contents of this complex type.

1. <xsd:complexType name="CT\_Shape">
2. <xsd:sequence>
3. <xsd:element name="Path" minOccurs="1" maxOccurs="unbounded" type="CT\_Path"/>
4. </xsd:sequence>
5. <xsd:attribute name="Name" type="xsd:string" use="required"/>
6. <xsd:attribute name="Layout" type="xsd:string"/>
7. </xsd:complexType>

##### CT\_Shapes

*Referenced by:* [CT\_ShapeOutline\_Page](#Section_dec60ada5c5f4622b28b7c486527b022)

The **CT\_Shapes** complex type specifies geometric outline information for multiple [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Child Elements:*

**Shape :** An optional list of [CT\_Shape](#Section_6defd531916c4391bbb80bc71d0c0c5c) elements. CT\_Shape elements are not present if there are no shapes in the [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640).

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment defines the contents of this complex type.

1. <xsd:complexType name="CT\_Shapes">
2. <xsd:sequence>
3. <xsd:element name="Shape" minOccurs="0" maxOccurs="unbounded" type="CT\_Shape"/>
4. </xsd:sequence>
5. </xsd:complexType>

##### CT\_ShapeOutline\_Page

*Referenced by:* [Page](#Section_fffd3fb79dba430dbe611c200624a659)

The **CT\_ShapeOutline\_Page** complex type specifies geometric outline information for the [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) within a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) of the document.

*Child Elements:*

**Shapes :** A [CT\_Shapes](#Section_f63911d856514018837174226fb2b6ab) element.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment defines the contents of this complex type.

1. <xsd:complexType name="CT\_ShapeOutline\_Page">
2. <xsd:sequence>
3. <xsd:element name="Shapes" type="CT\_Shapes"/>
4. </xsd:sequence>
5. </xsd:complexType>

### ShapeTextBinding XML Part

This [part](#Section_66eb42ec5ca8459696291233020c701f) specifies information used to [update](#Section_b30a1cbd988f4d6bbf2e7219c9423972) the [text element](#Section_b061c2bde75d48edb51047479b51c6ec) [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) in a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25).

#### Global Elements

##### Page

A [CT\_ShapeTextBinding\_Page](#Section_c4f3d4ae0c3d4088a40d9e3e1471dd99) element.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this element.

1. <xsd:element name="Page" type="CT\_ShapeTextBinding\_Page"/>

#### Complex Types

##### CT\_TextRun

*Referenced by:* [CT\_ShapeText](#Section_80e46fba919845399e9b8c1147fccc25)

The **CT\_TextRun** complex type specifies information about a [text run](#Section_8a573079d4b14123825dcb299f45d6b0) of a [text element](#Section_b061c2bde75d48edb51047479b51c6ec) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075).

*Attributes:*

**Name :** An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the identifier of the corresponding text run in the text element data graphic. It MUST be unique amongst the **Name** attributes of the CT\_TextRun elements in this XML part.

**Formula :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc) used to recalculate the content of the text run in a text element data graphic during [diagram update](#Section_b30a1cbd988f4d6bbf2e7219c9423972). The result token value of the [formula evaluation](#Section_81f0fbe77a314d34b52f5fa9c9380c33) is converted to a string. If the result token type is [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), the result token value is used without additional formatting. If the result token type is not PtgStr1, the result token value is formatted using the information specified by the **Format** attribute. It MUST exist if the **Color** attribute does not exist. It MUST NOT be an empty string.

**Format :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies formatting information for the result of the evaluation of **Formula**. It MUST exist if and only if the **Formula** attribute exists. It MUST be ignored if the result type of the **Formula** evaluation is not PtgStr1. It MUST have the following format:

1. Formatting@ServerAction@UnitLabel@CurrencyID@DisplayUnit@LanguageID@CalendarID

| Name | Description |
| --- | --- |
| Formatting | A [vFormatString](#Section_3d8a401ea64247d082ad4cfbdd2331b9). |
| ServerAction | A [vServerAction](#Section_094c818d877844b881c7dd957e2aaf7f). |
| UnitLabel | A [vUnitLabel](#Section_b8166c01556f4257b5bcc081923ee37d). |
| CurrencyID | A [vCurrencyID](#Section_b13bb6692bc04291b31c4532158fa675). MUST be ignored if the result of the **Formula** evaluation does not specify a currency. |
| DisplayUnit | A [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b) that the result value of the **Formula** evaluation is converted to before formatting. MUST be zero or the numerical identifier of a token type, as specified in section [2.5.5](#Section_1c7c6787214545db854e2a1e1ca24d69), that is part of the vUnitType grouping. MUST be ignored if equal to zero. |
| LanguageID | A [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24). |
| CalendarID | A [vCalendar](#Section_4c2dd2c033874a6c99af9b277c89b411). MUST be ignored if the result of the **Formula** evaluation does not specify a date. |

**Color :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies a formula expression used to recalculate the color of the text run upon [**refresh**](#gt_d947dfed-8e3a-4451-b978-d69a89ae6060) in a text element data graphic during diagram update. The result of the formula evaluation is converted to a color value. The color value is a string representing an eight-digit hexadecimal number with a prefix of the character "#". The first two digits MUST be equal to FF, the next two specify the intensity of red, the next two specify the intensity of green and the final two specify the intensity of blue. It MUST exist if the **Format** attribute does not exist. It MUST NOT be an empty string.

**DataRefs :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the external data bound to **Formula**. If this attribute does not exist or if it is an empty string then the external data bound to **Formula** is unknown. This attribute is unused if **Formula** does not exist. If a string for this attribute does exist it MUST have the following format:

1. rowID,colID

| Name | Description |
| --- | --- |
| rowID | An integer that specifies the external data [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) that the text run is bound to. MUST be the value of the **BindingID** attribute of a [CT\_Binding](#Section_5ad7ddf78438400abe979591e0cca24a) element in the [DataBinding XML Part](#Section_311514cd8ef444c9ab42562671155333). |
| colID | A string that specifies the [**field**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc) in the [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) that the text run is bound to. MUST be the value of the **Name** attribute of a [CT\_DataColumn](#Section_f5259dad7ba3465092d7ef1d69cc6a70) element in the [DataConnection XML Part](#Section_c489eefb7a014941bc37a4d332e66dc8). |

The rowID and colID values MUST be separated by a comma.

**DataRefsColor :**  An optional xsd:string ([XMLSCHEMA2] section 3.2.1) attribute that specifies the external data bound to **Color**. If this attribute does not exist or if it is an empty string then the external data bound to **Color** is unknown. This attribute is unused if **Color** does not exist. If a string for this attribute does exist it MUST have the format described by **DataRefs**.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_TextRun">
2. <xsd:attribute name="Name" type="xsd:string" use="required"/>
3. <xsd:attribute name="Formula" type="xsd:string" use="optional"/>
4. <xsd:attribute name="Format" type="xsd:string" use="optional"/>
5. <xsd:attribute name="Color" type="xsd:string" use="optional"/>
6. <xsd:attribute name="DataRefs" type="xsd:string" use="optional"/>
7. <xsd:attribute name="DataRefsColor" type="xsd:string" use="optional"/>
8. </xsd:complexType>

##### CT\_ShapeText

*Referenced by:* [CT\_ShapeTextBinding\_Page](#Section_c4f3d4ae0c3d4088a40d9e3e1471dd99)

The **CT\_ShapeText** complex type specifies information about the [text element](#Section_b061c2bde75d48edb51047479b51c6ec) [data graphic](#Section_037c6f2c08934c85aaf588143a8d9075) for a [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b).

*Child Elements:*

**TextRun :** A [CT\_TextRun](#Section_9b45f00ef7384cfba633da8bca9b0af4) element.

*Attributes:*

**Name :**  An xsd:string ([[XMLSCHEMA2]](https://go.microsoft.com/fwlink/?LinkId=90610) section 3.2.1) attribute that specifies the identifier of the corresponding text element data graphic in the [ShapeGraphic Part](#Section_d86834c499d248e2b13ab2c8f016eca0). It MUST be unique amongst all the **Name** attributes of the CT\_ShapeText elements in this XML part.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_ShapeText">
2. <xsd:sequence>
3. <xsd:element name="TextRun" minOccurs="1" maxOccurs="1" type="CT\_TextRun"/>
4. </xsd:sequence>
5. <xsd:attribute name="Name" type="xsd:string" use="required"/>
6. </xsd:complexType>

##### CT\_ShapeTextBinding\_Page

*Referenced by:* [Page](#Section_665d689445e34b03bf469a8c2002e37d)

The **CT\_ShapeTextBinding\_Page** complex type specifies information about the [text element](#Section_b061c2bde75d48edb51047479b51c6ec) [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) for [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) in a [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640).

*Child Elements:*

**ShapeText :** An optional list of [CT\_ShapeText](#Section_80e46fba919845399e9b8c1147fccc25) elements. CT\_ShapeText elements are not present if there are no text element data graphics associated with the shapes of the drawing page.

*Attributes:*

**Name :** This attribute is unused and MUST be ignored.

The following W3C XML Schema ([[XMLSCHEMA1]](https://go.microsoft.com/fwlink/?LinkId=90608) section 2.1) fragment specifies the contents of this complex type.

1. <xsd:complexType name="CT\_ShapeTextBinding\_Page">
2. <xsd:sequence>
3. <xsd:element name="ShapeText" minOccurs="0" maxOccurs="unbounded" type="CT\_ShapeText"/>
4. </xsd:sequence>
5. <xsd:attribute name="Name" type="xsd:string"/>
6. </xsd:complexType>

## Formula Evaluation and Shape Property Recalculation

### Introduction

The following sections specify the elements of a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc).

### Formula ABNF and Full Grammar Definition

A [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc) MUST conform to the following ABNF [[RFC4234]](https://go.microsoft.com/fwlink/?LinkId=90462) grammar.

val = expression

expression = sp (operand / function-call)

operand = token

token = [PtgAcre](#Section_7f6eff5e434a407b8565fe966e0b066c) / [PtgAngDD](#Section_222188be55a34b90b2c39eb9f5878bab) / [PtgAngDft](#Section_57e444144344437fb32a6cf2d3b20d2f) / [PtgAngDMS](#Section_459da555a74b48439f4cbc516a043d30) / [PtgAngRad](#Section_a09cdad2084647eb97edef0ef25726e0) / [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508) / [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882) / [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3) / [PtgDataBinding](#Section_7442cb26af7f46ce9bd348656835396c) / [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6) / [PtgEDay](#Section_f6690609c80a49ed9fffe0dbcfb262a6) / [PtgEHour](#Section_e31838fa14684d478f193c033df45e4c) / [PtgEMin](#Section_f58cc7ec8a1246d89edb4b3e00a08ac6) / [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2) / [PtgESec](#Section_01c1da7178fc485c8ee6b48674eeb35b) / [PtgEWeek](#Section_4aae0088510448abaf3f8ec63f95fd5f) / [PtgFunc](#Section_8206928cdaaf400daa071a566fe1c88b) / [PtgFuncVar](#Section_e483d78ecd494ba7aa042f843a09f8df) / [PtgHectare](#Section_7b405e52f9c34451a7f90ad394edd4aa) / [PtgJmp](#Section_a2fcbad2d4cd4d0bb4338ec960e51755) / [PtgJmpF](#Section_81250f6f47d04c3d93cd60e2d8798784) / [PtgJmpLabel](#Section_5f545b4b47f44ed38932659537323ff9) / [PtgJmpT](#Section_f605c7795ea249bca765efaf9ae6e1f0) / [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd) / [PtgNoOp](#Section_5d39bf2968f3460e8ae0a46d6262dd4d) / [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0) / [PtgNumCM](#Section_bc2d88fb83a94d9588803acf4c0cdc27) / [PtgNumDft](#Section_de132a3cdab04751844609cf15ec3756) / [PtgNumF](#Section_09bdcb1c922444c8b9c3ecdaa03e0c6d) / [PtgNumFI](#Section_a3917f0c42e94c71b55c70ea3aed6819) / [PtgNumI](#Section_ac87036f348249b58cba49659d7598a8) / [PtgNumKM](#Section_6dc5f00e5e7945678ac890f98cd40706) / [PtgNumM](#Section_000148cf28984a299ade5cc03f395c6c) / [PtgNumMI](#Section_e1f97f6e268c4429868c74e9323a7b9e) / [PtgNumMM](#Section_5fc97e8093e041d7bcd6ddd2c66aa589) / [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84) / [PtgNumNM](#Section_bf29f1f677664e9e856811466228244f) / [PtgNumPct](#Section_ef65a2675fd84ba3a1448fb7698f6551) / [PtgNumYards](#Section_7039749a686144d9bcd9da3946769b16) / [PtgPageDft](#Section_fe0d479f9b534257a3d13618b7314aaa) / [PtgPop](#Section_a5016a0a9e794b6dbd38231239a97710) / [PtgPnt](#Section_45fae81bb4e6413394e32daf5c98d790) / [PtgPushTop](#Section_46e71a47bb0d441cb00756eb6182c94a) / [PtgRecalcRef](#Section_1877c57f45234887a93a70db1b9d46e5) / [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69) / [PtgTDurDft](#Section_1200eb57709c4fcc9d20e2bac20fa493) / [PtgTypCD](#Section_bf50076676d84a8d900468c81b728b8e) / [PtgTypCi](#Section_c5f1913417a7448287105b162359a1fa) / [PtgTypDft](#Section_d66b059e323d45739c91ec57ed148b16) / [PtgTypDi](#Section_a56c1d526fe9452b9857c97fd84c3825) / [PtgTypPi](#Section_f3c4aabfb3bb4c68b5b9f175f9620048) / [PtgTypPP](#Section_997b3916e4734908b63936fd05317e60) / [PtgTypPt](#Section_b30a64caae964c76ac0728982f26bb31) / [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b)

function-call = if-expression / fixed-func / variable-func

if-expression = expression sp (PtgJmpF / PtgJmpT) sp expression sp PtgJmp sp PtgJmpLabel sp PtgPop sp expression PtgJmpLabel

fixed-func = [Abs](#Section_ac15962f890348f99b787d74a264ff70) / [ACos](#Section_7762f008b9294ca4bfbbcc6fda5857a2) / [Add](#Section_665760e6818241ccbb499eb6784a17a6) / [Ang360](#Section_e60c124063804a61a949983cd68c2faa) / [ASin](#Section_8d23b8911c7548e0b1aa375b4d35752a) / [ATan](#Section_8fd8722125b34d8684a2e4ec2a8232fe) / [ATan2](#Section_d736ae1d0195473480298025561413ae) / [BitAnd](#Section_7256bb6117d4492980b790555ed5641a) / [BitNot](#Section_5d8adde5567f436ab018aa972a1ca3b4) / [BitOr](#Section_7bb67b2740794448bd12439129b78927) / [BitXor](#Section_72d57fa3d3d643eaab25ac53c0f385ae) / [Blend](#Section_ae98bf6e773b4822ace6c7f408c5d72e) / [CellIsThemed](#Section_a2a02f04068e44b290d7cdc8643f2023) / [Char](#Section_6c2ccd755ee4427c9cb24f414e43995e) / [Cos](#Section_5dd3ee1bd91441b4a0d0d6f9c20020d8) / [Cat](#Section_96f779a649c541879d0698b956760265) / [CosH](#Section_f378fdf119d04ecf80fc73e806342f3d) / [CY](#Section_75f0d1d2078744afa3bcdce85676591c) / [Date](#Section_3112fa9d82044dcfa53fc58f2b2bbb99) / [Deg](#Section_3624b42b2a034f7dadc5dfaac36ee3a9) / [Div](#Section_22f5e4257463428c8b879257310dce28) / [EEQ](#Section_e05c6f8efaa944948905032e95b2d7e7) / [EGE](#Section_fe352fc6b52a4bcda51bfe343e63ed0e) / [EGT](#Section_417a0e1d182f43f39090e0e2bb6cb4e9) / [ELE](#Section_5be29110bd0f4c889977bb5acadd06e8) / [ELT](#Section_82ef1ff2f70f40b5a645846d3f3158f4) / [ENE](#Section_743eb63035bf4b27a9a66455c5582c1b) / [FEQ](#Section_09cc3a4c55b541f8b7e7587543f09f6b) / [FGE](#Section_ecc5f04605fe4bc8b55527b8af39d6e2) / [FGT](#Section_cdd9657a9451406da2c8ce4476c5767f) / [FLE](#Section_7c57da47a59a482189c4c641a88dd2c2) / [FLT](#Section_8343e93a1bee4813aa469079707f8773) / [FNE](#Section_933a601244c940ddb0a0fa936d8a48ca) / [FormatEx](#Section_5043dc904a694be0a82ee78817288f13) / [HSL](#Section_34029dd83de24dccb178577569bfae12) / [Hue](#Section_5c026f3c342d4c4fbb6b7788a9120a7f) / [HueDiff](#Section_4661b142b56e44aeab179b5380bf1c5a) / [Int](#Section_8fc9b021fa0141d1866e224f8bd0719f) / [IntersectX](#Section_19e8ea77229a4f2a89f3f5c837ce9d35) / [IntersectY](#Section_756cf723a35d4f16b9e622cba43c97d1) / [Intup](#Section_35a7b694448342c3b476c74eaa02f40a) / [IsErr](#Section_daad281ef1d249c092d54c023d4a6679) / [IsErrNA](#Section_d1c99e1d3db14b9084eb3bada4f9a4b6) / [IsError](#Section_447254752e8446008be8460fda2f67d0) / [IsErrValue](#Section_afc2ec8cc6ab47c79fadb6151a48f7ca) / [Len](#Section_622fcc92f34b46fca1e9963431a8053d) / [Ln](#Section_4cf5e3dcee9647e698169a0dd75a0883) / [Log10](#Section_3e5556f668894bd289204faeb44fa4b3) / [Lower](#Section_c81fcfe8b9e440ddbd72e0efb23103f9) / [Lum](#Section_aa06018e8235479db01da2d02aa0b9ba) / [LumDiff](#Section_ef9c2489cf8e489383e58757c5b2b57a) / [Magnitude](#Section_8013d759a53c44078499a4b7f6a505e7) / [Mid](#Section_d16240afddeb48dda574b287c66340ed) / [Modulus](#Section_05d2a9ec70254d6bab0b3564a4644544) / [MsoShade](#Section_5c930f5672654b7eb2c921016b7784b5) / [MsoTint](#Section_edb72710046d423283c3e38f94628e00) / [Mul](#Section_9ea4b2faacb44423a91f850be85a98fc) / [Not](#Section_9f1f0c0af6f34965ba55af7aa0c60c93) / [Now](#Section_98767867233649c2a290958683bc9a0d) / [Pct](#Section_b0b58c93e77841779a32c20890802739) / [Pi](#Section_01fcf3f9c3f345b2b5cc5ff9f00afc81) / [Pow](#Section_aecdce6ab67b4f4bbfcb9398b6fec21d) / [Pnt](#Section_bc0addccf70a4e5c981d0ff88d23ba09) / [Rad](#Section_75a45a7ca53e43d3947f8ad7fa30157d) / [Rand](#Section_79d086a25da64ee29829be5af12835bf) / [Replace](#Section_570ef2505bcd40388afa1a875bf50770) / [RGB](#Section_db09d78046eb4ad19dc749ebc187a383) / [Round](#Section_6ec2ac090e1a4fb9b212ff9233a2adb4) / [Sat](#Section_283a1d24a19b48cc8f3eae152ed12e33) / [SatDiff](#Section_9bf7e00cb38441debc494d356e5a29dc) / [SetAtRefEval](#Section_422d1c250b3640efbd8591a3b7b53a26) / [Shade](#Section_9f30bd848ad346d8aae90e2a7b73188f) / [Sin](#Section_3431c35c5fbe4d979c2d11b38b10c8f2) / [SinH](#Section_e47a000ce6d24637942a15d0c4a37332) / [Sqrt](#Section_21f56dff97624702a8d653f31bdf3161) / [StrSameEx](#Section_c0434b6c5f5148ed8cff1fe499c73262) / [Sub](#Section_1bba6fe5ccae4beca707af111c3798e8) / [Tan](#Section_9ee23ba8f95d4c0ea8502374ec465e38) / [TanH](#Section_661cb296b35c4c14bd69d27132354595) / [TextHeight](#Section_d7d7e4d35ac8493bbf43940f49ddea6e) / [Time](#Section_291011bc78944de5aed906836a6332b9) / [Tint](#Section_04d6a79942d04ee08b9126a049a0efa7) / [Tone](#Section_997f3432602d425eb6f804bd1ff70253) / [Trim](#Section_732cea39bbc848cf8194b56bebbe99ee) / [Trunc](#Section_cb4984ba303d46949d72b6579222c282) / [UMinus](#Section_c3d40d6127aa466086661d4778c2cdc4) / [UPlus](#Section_1fec68737b9d45309309cc81970b42c5) / [Upper](#Section_46f69d676fc24c849f066749c3ccd5f6)

variable-func = [And](#Section_2ed042522c4e4127a430bc711205b624) / [Bound](#Section_bbcae47f0b9045e7b977266ed8ff95f9) / [Ceiling](#Section_8f189e0061184600837579b7c614a8ad) / [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) / [DateValue](#Section_b446491e250d45d19cf1e0366b99ab76) / [Day](#Section_fb6a38429d3a46b2a2382da4f0280b58) / [DayOfYear](#Section_46b2e2f7565b493299c9a57846a7e32c) / [Find](#Section_6d05d919d6504b72a245cfc1208aa564) / [Floor](#Section_dab6b8dc3bcf4c858cad5ee336228a6a) / [Hour](#Section_ae1d796b6ace40ceb40365e42e6f053a) / [Index](#Section_387f76986c5a48b9ab7bd2ae1d7a8e56) / [Left](#Section_ad6229bc1c014ce79aba8dd70c559dab) / [Lookup](#Section_73b5f09a88274eba932ba7c6deac33b9) / [Max](#Section_ce8b0939dcef4c6196f733b6165a7739) / [Min](#Section_6b30c62f628542a5bbc43a660cca4a3e) / [Minute](#Section_aadc518bedfa4c5cb9153bdb01b39d3a) / [Month](#Section_94f622168a994fd59202e6dfb43fe4ce) / [Or](#Section_282f7e8edffc4c2f9dc442fbbec44446) / [Right](#Section_d2e4d8ac78d34641811157faec3c6b98) / [Second](#Section_4bff656d866140d4a27b3896e503b624) / [SetAtRef](#Section_e4b35b9aebca46ea8f74c6441aa09f68) / [SetAtRefExpr](#Section_120b16678db140ae9d93fed35414aea8) / [ShapeText](#Section_cacc5273bcda480b94a9e8433103819a) / [Sign](#Section_d09e0bd732f54f288c734b9bd31f4f19) / [StrSame](#Section_d78370656e7e4d50ac6258dd77c7b7cf) / [Substitute](#Section_fbbbca96110b49098720b105389044de) / [Sum](#Section_20d1ad033898478fb5b9f2fb17b49590) / [TimeValue](#Section_1e82ec257009418bab7d70b41fed1fa2) / [WeekDay](#Section_a3daca1f74f7441c85a32e55f8ef6f17) / [Year](#Section_ccd3fce2e5944c2b96f8a9821a70f105)

fixed-func-name = "ABS" / "ACOS" / "\_ADD" / "ANG360" / "ASIN" / "ATAN" / "ATAN2" / "BITAND" / "BITNOT" / "BITOR" / "BITXOR" / "BLEND" / "CHAR" / "COS" / "\_CAT" / "COSH" / "CY" / "DATE" / "DEG" / "\_DIV" / "\_EEQ" / "\_EGE" / "\_EGT" / "\_ELE" / "\_ELT" / "\_ENE" / "\_FEQ" / "\_FGE" / "\_FGT" / "\_FLE" / "\_FLT" / "\_FNE" / "FORMATEX" / "HSL" / "HUE" / "HUEDIFF" / "INT" / "INTERSECTX" / "INTERSECTY" / "INTUP" / "ISERR" / "ISERRNA" / "ISERROR" / "ISERRVALUE" / "LEN" / "LN" / "LOG10" / "LOWER" / "LUM" / "LUMDIFF" / "MAGNITUDE" / "MID" / "MODULUS" / "MSOSHADE" / "MSOTINT" / "\_MUL" / "NOT" / "NOW" / "\_PCT" / "PI" / "POW" / "RAD" / "RAND / "REPLACE" / "RGB" / "ROUND" / "SAT" / "SATDIFF" / "SETATREFEVAL" / "SHADE" / "SIN" / "SINH" / "SQRT" / "STRSAMEEX" / "\_SUB" / "TAN" / "TANH" / "TEXTHEIGHT" / "TIME" / "TINT" / "TONE" / "TRIM" / "TRUNC" / "\_UMINUS" / "\_UPLUS" / "UPPER"

var-func-name = "AND" / "BOUND" / "CEILING" / "DATETIME" / "DATEVALUE" / "DAY" / "DAYOFYEAR" / "FIND" / "FLOOR" / "HOUR" / "INDEX" / "LEFT" / "LOOKUP" / "MAX" / "MIN" / "MINUTE" / "MONTH" / "OR" / "RIGHT" / "SECOND" / "SETATREF" / "SETATREFEXPR" / "SIGN" / "STRSAME" / "SUBSTITUTE" / "SUM" / "TIMEVALUE" / "WEEDKAY" / "YEAR"

color-value = xsd-int

currency = xsd-unsignedInt

date-time-value = xsd-double

unit = xsd-unsignedInt

shape-name = string-value

shape-data = string-value

shape-data-type = xsd-unsignedInt

dimension = xsd-unsignedInt

num-args = xsd-unsignedInt

empty-value = ""

string-value = xsd-string

double-value = xsd-double

unsigned-integer-value = xsd-unsignedInt

bool-value = true-value / false-value

true-value = %x54/%x74 %x52/%x72 %x55/%x75 %x45/%x65

false-value = %x46/%x66 %x41/%x61 %x4C/%x6C %x53/%x73 %x45/%x65

xsd-double = ["+" / "-"] \*digit ["." 1\*digit] [("e" / "E") ["+" / "-"] 1\*digit]

xsd-int = ["+" / "-"] 1\*digit

xsd-unsignedInt = 1\*digit

xsd-string = \*(%x20-7E)

token-separator = %x3A ; colon

sp = %x20 ; space

digit = %x30-39 ; Decimal digits (0-9)

### Function Token Table

This section specifies the list of functions that can be called from a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc). The function display name, ABNF func-name and function type are provided for each function.

| Display name | ABNF func-name | Type |
| --- | --- | --- |
| [Abs](#Section_ac15962f890348f99b787d74a264ff70) | ABS | [PtgFunc](#Section_8206928cdaaf400daa071a566fe1c88b) |
| [ACos](#Section_7762f008b9294ca4bfbbcc6fda5857a2) | ACOS | PtgFunc |
| [Add](#Section_665760e6818241ccbb499eb6784a17a6) | \_ADD | PtgFunc |
| [And](#Section_2ed042522c4e4127a430bc711205b624) | AND | PtgFunc |
| [Ang360](#Section_e60c124063804a61a949983cd68c2faa) | ANG360 | PtgFunc |
| [ASin](#Section_8d23b8911c7548e0b1aa375b4d35752a) | ASIN | PtgFunc |
| [ATan](#Section_8fd8722125b34d8684a2e4ec2a8232fe) | ATAN | PtgFunc |
| [ATan2](#Section_d736ae1d0195473480298025561413ae) | ATAN2 | PtgFunc |
| [BitAnd](#Section_7256bb6117d4492980b790555ed5641a) | BITAND | PtgFunc |
| [BitNot](#Section_5d8adde5567f436ab018aa972a1ca3b4) | BITNOT | PtgFunc |
| [BitOr](#Section_7bb67b2740794448bd12439129b78927) | BITOR | PtgFunc |
| [BitXor](#Section_72d57fa3d3d643eaab25ac53c0f385ae) | BITXOR | PtgFunc |
| [Blend](#Section_ae98bf6e773b4822ace6c7f408c5d72e) | BLEND | PtgFunc |
| [Bound](#Section_bbcae47f0b9045e7b977266ed8ff95f9) | BOUND | [PtgFuncVar](#Section_e483d78ecd494ba7aa042f843a09f8df) |
| [Ceiling](#Section_8f189e0061184600837579b7c614a8ad) | CEILING | PtgFuncVar |
| [CellIsThemed](#Section_a2a02f04068e44b290d7cdc8643f2023) | CELLISTHEMED | PtgFunc |
| [Char](#Section_6c2ccd755ee4427c9cb24f414e43995e) | CHAR | PtgFunc |
| [Cos](#Section_5dd3ee1bd91441b4a0d0d6f9c20020d8) | COS | PtgFunc |
| [Cat](#Section_96f779a649c541879d0698b956760265) | \_CAT | PtgFunc |
| [CosH](#Section_f378fdf119d04ecf80fc73e806342f3d) | COSH | PtgFunc |
| [CY](#Section_75f0d1d2078744afa3bcdce85676591c) | CY | PtgFunc |
| [Date](#Section_3112fa9d82044dcfa53fc58f2b2bbb99) | DATE | PtgFunc |
| [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) | DATETIME | PtgFuncVar |
| [DateValue](#Section_b446491e250d45d19cf1e0366b99ab76) | DATEVALUE | PtgFuncVar |
| [Day](#Section_fb6a38429d3a46b2a2382da4f0280b58) | DAY | PtgFuncVar |
| [DayOfYear](#Section_46b2e2f7565b493299c9a57846a7e32c) | DAYOFYEAR | PtgFuncVar |
| [Deg](#Section_3624b42b2a034f7dadc5dfaac36ee3a9) | DEG | PtgFunc |
| [Div](#Section_22f5e4257463428c8b879257310dce28) | \_DIV | PtgFunc |
| [EEQ](#Section_e05c6f8efaa944948905032e95b2d7e7) | \_EEQ | PtgFunc |
| [EGE](#Section_fe352fc6b52a4bcda51bfe343e63ed0e) | \_EGE | PtgFunc |
| [EGT](#Section_417a0e1d182f43f39090e0e2bb6cb4e9) | \_EGT | PtgFunc |
| [ELE](#Section_5be29110bd0f4c889977bb5acadd06e8) | \_ELE | PtgFunc |
| [ELT](#Section_82ef1ff2f70f40b5a645846d3f3158f4) | \_ELT | PtgFunc |
| [ENE](#Section_743eb63035bf4b27a9a66455c5582c1b) | \_ENE | PtgFunc |
| [FEQ](#Section_09cc3a4c55b541f8b7e7587543f09f6b) | \_FEQ | PtgFunc |
| [FGE](#Section_ecc5f04605fe4bc8b55527b8af39d6e2) | \_FGE | PtgFunc |
| [FGT](#Section_cdd9657a9451406da2c8ce4476c5767f) | \_FGT | PtgFunc |
| [Find](#Section_6d05d919d6504b72a245cfc1208aa564) | FIND | PtgFuncVar |
| [FLE](#Section_7c57da47a59a482189c4c641a88dd2c2) | \_FLE | PtgFunc |
| [Floor](#Section_dab6b8dc3bcf4c858cad5ee336228a6a) | FLOOR | PtgFuncVar |
| [FLT](#Section_8343e93a1bee4813aa469079707f8773) | \_FLT | PtgFunc |
| [FNE](#Section_933a601244c940ddb0a0fa936d8a48ca) | \_FNE | PtgFunc |
| [FormatEx](#Section_5043dc904a694be0a82ee78817288f13) | FORMATEX | PtgFunc |
| [Hour](#Section_ae1d796b6ace40ceb40365e42e6f053a) | HOUR | PtgFuncVar |
| [HSL](#Section_34029dd83de24dccb178577569bfae12) | HSL | PtgFunc |
| [Hue](#Section_5c026f3c342d4c4fbb6b7788a9120a7f) | HUE | PtgFunc |
| [HueDiff](#Section_4661b142b56e44aeab179b5380bf1c5a) | HUEDIFF | PtgFunc |
| [Index](#Section_387f76986c5a48b9ab7bd2ae1d7a8e56) | INDEX | PtgFuncVar |
| [Int](#Section_8fc9b021fa0141d1866e224f8bd0719f) | INT | PtgFunc |
| [IntersectX](#Section_19e8ea77229a4f2a89f3f5c837ce9d35) | INTERSECTX | PtgFunc |
| [IntersectY](#Section_756cf723a35d4f16b9e622cba43c97d1) | INTERSECTY | PtgFunc |
| [Intup](#Section_35a7b694448342c3b476c74eaa02f40a) | INTUP | PtgFunc |
| [IsErr](#Section_daad281ef1d249c092d54c023d4a6679) | ISERR | PtgFunc |
| [IsErrNA](#Section_d1c99e1d3db14b9084eb3bada4f9a4b6) | ISERRNA | PtgFunc |
| [IsError](#Section_447254752e8446008be8460fda2f67d0) | ISSERROR | PtgFunc |
| [IsErrValue](#Section_afc2ec8cc6ab47c79fadb6151a48f7ca) | ISERRVALUE | PtgFunc |
| [Left](#Section_ad6229bc1c014ce79aba8dd70c559dab) | LEFT | PtgFuncVar |
| [Len](#Section_622fcc92f34b46fca1e9963431a8053d) | LEN | PtgFunc |
| [Ln](#Section_4cf5e3dcee9647e698169a0dd75a0883) | LN | PtgFunc |
| [Log10](#Section_3e5556f668894bd289204faeb44fa4b3) | LOG10 | PtgFunc |
| [Lookup](#Section_73b5f09a88274eba932ba7c6deac33b9) | LOOKUP | PtgFuncVar |
| [Lower](#Section_c81fcfe8b9e440ddbd72e0efb23103f9) | LOWER | PtgFunc |
| [Lum](#Section_aa06018e8235479db01da2d02aa0b9ba) | LUM | PtgFunc |
| [LumDiff](#Section_ef9c2489cf8e489383e58757c5b2b57a) | LUMDIFF | PtgFunc |
| [Magnitude](#Section_8013d759a53c44078499a4b7f6a505e7) | MAGNITUDE | PtgFunc |
| [Max](#Section_ce8b0939dcef4c6196f733b6165a7739) | MAX | PtgFuncVar |
| [Mid](#Section_d16240afddeb48dda574b287c66340ed) | MID | PtgFunc |
| [Min](#Section_6b30c62f628542a5bbc43a660cca4a3e) | MIN | PtgFuncVar |
| [Minute](#Section_aadc518bedfa4c5cb9153bdb01b39d3a) | MINUTE | PtgFuncVar |
| [Modulus](#Section_05d2a9ec70254d6bab0b3564a4644544) | MODULUS | PtgFunc |
| [Month](#Section_94f622168a994fd59202e6dfb43fe4ce) | MONTH | PtgFuncVar |
| [MsoShade](#Section_5c930f5672654b7eb2c921016b7784b5) | MSOSHADE | PtgFunc |
| [MsoTint](#Section_edb72710046d423283c3e38f94628e00) | MSOTINT | PtgFunc |
| [Mul](#Section_9ea4b2faacb44423a91f850be85a98fc) | \_MUL | PtgFunc |
| [Not](#Section_9f1f0c0af6f34965ba55af7aa0c60c93) | NOT | PtgFunc |
| [Now](#Section_98767867233649c2a290958683bc9a0d) | NOW | PtgFunc |
| [Or](#Section_282f7e8edffc4c2f9dc442fbbec44446) | OR | PtgFunc |
| [Pct](#Section_b0b58c93e77841779a32c20890802739) | \_PCT | PtgFunc |
| [Pi](#Section_01fcf3f9c3f345b2b5cc5ff9f00afc81) | PI | PtgFunc |
| [Pow](#Section_aecdce6ab67b4f4bbfcb9398b6fec21d) | POW | PtgFunc |
| [Pnt](#Section_bc0addccf70a4e5c981d0ff88d23ba09) | PNT | PtgFunc |
| [Rad](#Section_75a45a7ca53e43d3947f8ad7fa30157d) | RAD | PtgFunc |
| [Rand](#Section_79d086a25da64ee29829be5af12835bf) | RAND | PtgFunc |
| [Replace](#Section_570ef2505bcd40388afa1a875bf50770) | REPLACE | PtgFunc |
| [RGB](#Section_db09d78046eb4ad19dc749ebc187a383) | RGB | PtgFunc |
| [Right](#Section_d2e4d8ac78d34641811157faec3c6b98) | RIGHT | PtgFuncVar |
| [Round](#Section_6ec2ac090e1a4fb9b212ff9233a2adb4) | ROUND | PtgFunc |
| [Sat](#Section_283a1d24a19b48cc8f3eae152ed12e33) | SAT | PtgFunc |
| [SatDiff](#Section_9bf7e00cb38441debc494d356e5a29dc) | SATDIFF | PtgFunc |
| [Second](#Section_4bff656d866140d4a27b3896e503b624) | SECOND | PtgFuncVar |
| [SetAtRef](#Section_e4b35b9aebca46ea8f74c6441aa09f68) | SETATREF | PtgFuncVar |
| [SetAtRefEval](#Section_422d1c250b3640efbd8591a3b7b53a26) | SETATREFEVAL | PtgFunc |
| [SetAtRefExpr](#Section_120b16678db140ae9d93fed35414aea8) | SETATREFEXPR | PtgFuncVar |
| [Shade](#Section_9f30bd848ad346d8aae90e2a7b73188f) | SHADE | PtgFunc |
| [ShapeText](#Section_cacc5273bcda480b94a9e8433103819a) | SHAPETEXT | PtgFuncVar |
| [Sign](#Section_d09e0bd732f54f288c734b9bd31f4f19) | SIGN | PtgFuncVar |
| [Sin](#Section_3431c35c5fbe4d979c2d11b38b10c8f2) | SIN | PtgFunc |
| [SinH](#Section_e47a000ce6d24637942a15d0c4a37332) | SINH | PtgFunc |
| [Sqrt](#Section_21f56dff97624702a8d653f31bdf3161) | SQRT | PtgFunc |
| [StrSame](#Section_d78370656e7e4d50ac6258dd77c7b7cf) | STRSAME | PtgFuncVar |
| [StrSameEx](#Section_c0434b6c5f5148ed8cff1fe499c73262) | STRSAMEEX | PtgFunc |
| [Sub](#Section_1bba6fe5ccae4beca707af111c3798e8) | \_SUB | PtgFunc |
| [Substitute](#Section_fbbbca96110b49098720b105389044de) | SUBSTITUTE | PtgFuncVar |
| [Sum](#Section_20d1ad033898478fb5b9f2fb17b49590) | SUM | PtgFuncVar |
| [Tan](#Section_9ee23ba8f95d4c0ea8502374ec465e38) | TAN | PtgFunc |
| [TanH](#Section_661cb296b35c4c14bd69d27132354595) | TANH | PtgFunc |
| [TextHeight](#Section_d7d7e4d35ac8493bbf43940f49ddea6e) | TEXTHEIGHT | PtgFunc |
| [TextWidth](#Section_afe4155702d94267a3f029b60fa3b527) | TEXTWIDTH | PtgFunc |
| [Time](#Section_291011bc78944de5aed906836a6332b9) | TIME | PtgFunc |
| [TimeValue](#Section_1e82ec257009418bab7d70b41fed1fa2) | TIMEVALUE | PtgFuncVar |
| [Tint](#Section_04d6a79942d04ee08b9126a049a0efa7) | TINT | PtgFunc |
| [Tone](#Section_997f3432602d425eb6f804bd1ff70253) | TONE | PtgFunc |
| [Trim](#Section_732cea39bbc848cf8194b56bebbe99ee) | TRIM | PtgFunc |
| [Trunc](#Section_cb4984ba303d46949d72b6579222c282) | TRUNC | PtgFunc |
| [UMinus](#Section_c3d40d6127aa466086661d4778c2cdc4) | \_UMINUS | PtgFunc |
| [UPlus](#Section_1fec68737b9d45309309cc81970b42c5) | \_UPLUS | PtgFunc |
| [Upper](#Section_46f69d676fc24c849f066749c3ccd5f6) | UPPER | PtgFunc |
| [WeekDay](#Section_a3daca1f74f7441c85a32e55f8ef6f17) | WEEKDAY | PtgFuncVar |
| [Year](#Section_ccd3fce2e5944c2b96f8a9821a70f105) | YEAR | PtgFuncVar |

### Function Token Definitions

This section specifies the [function tokens](#Section_e78706d7cd8c424bbc3bfc74904871ca) that can be contained in a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc). The definition of each function specifies the function name, the type and sequence of expected arguments, and the type of token returned. Information is also included about how the functions are evaluated.

#### Abs

The **Abs** function performs an absolute value calculation.

**ABNF:**

1. Abs = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " ABS():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad)

This function returns a vNum containing the absolute value of **Arg1**.The unit of the return value is equal to the unit of **Arg1**.

#### ACos

The **ACos** function performs the arccosine calculation.

**ABNF:**

1. ACos = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " ACOS():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgAngDft](#Section_57e444144344437fb32a6cf2d3b20d2f), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a PtgAngDft containing the arccosine of the value of **Arg1**. If **Arg1** is less than ­1 or greater than 1, the function returns a PtgErr with an error code of #VALUE!.

#### Add

The **Add** function performs an addition calculation.

**ABNF:**

1. Add = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_ADD():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDoubleEx](#Section_17ab59629f9e4b75a84af0cc80400b88)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDoubleEx

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum or a PtgCy containing the sum of **Arg1** and **Arg2**. The type of the return value is calculated by the following algorithm.

SET returnType = [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

SET returnDimension = 0

SET returnCurrencyID = 0

IF Arg1.Type = PtgCy AND Arg2.Type != PtgCy THEN

SET returnType = PtgCy

SET returnCurrencyID = currencyID of Arg1

ELSE IF Arg1.Type != PtgCy AND Arg2.Type = PtgCy THEN

SET returnType = PtgCy

SET returnCurrencyID = currencyID of Arg2

ELSE IF Arg1.Type = PtgCy AND Arg2.Type = PtgCy THEN

SET returnType = PtgCy

IF currencyID of Arg1 = currencyID of Arg2 OR currencyID of Arg2 = 1 THEN

SET returnCurrencyID = currencyID of Arg1

ELSE IF currencyID of Arg1 = 1 THEN

SET returnCurrencyID = currencyID of Arg2

ELSE

SET returnType = PtgErr

END IF

ELSE IF Arg1.Type = [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6) THEN

SET returnType = Arg1.Type

SET returnDimension = 1

ELSE IF Arg2.Type= PtgDate THEN

SET returnType = Arg2.Type

SET returnDimension = 1

ELSE IF Arg1.Unit = [PtgNumDft](#Section_de132a3cdab04751844609cf15ec3756) AND Arg2.Unit is a [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b) THEN

SET returnType = Arg2.Unit

SET returnDimension = 1

ELSE IF Arg1.Unit is a vUnitType THEN

SET returnType = Arg1.Unit

IF Arg2.Dimension = 0 OR Arg2.Dimension = Arg1.Dimension THEN

SET returnDimension = Arg1.Dimension

ELSE

SET returnDimension = 1

END IF

IF (returnType = [PtgAcre](#Section_7f6eff5e434a407b8565fe966e0b066c) OR returnType = [PtgHectare](#Section_7b405e52f9c34451a7f90ad394edd4aa)) AND returnDimension != 2 THEN

SET returnType = PtgNumDft

END IF

ELSE IF Arg2.Unit is a vUnitType THEN

SET returnType = Arg2.Unit

IF Arg1.Dimension = 0 OR Arg2.Dimension = Arg1.Dimension THEN

SET returnDimension = Arg2.Dimension

ELSE

SET returnDimension = 1

END IF

IF (returnType = PtgAcre OR returnType = PtgHectare) AND returnDimension != 2 THEN

SET returnType = PtgNumDft

END IF

END IF

IF returnType = PtgCy THEN

Return type is PtgCy with currencyID = returnCurrencyID

ELSE IF returnType = PtgErr THEN

Return type is PtgErr with error code of #VALUE!

ELSE IF returnDimension = 0 THEN

Return type is PtgNum

ELSE IF returnDimension = 1 THEN

Return type is returnType

ELSE

Return type is [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84) with unit = returnType and dimension = returnDimension

END IF

#### And

The **And** function converts an argument to a [**Boolean**](#gt_1d79d7a7-ba2c-4b34-931c-7ba8057c87b2) value according to the conversion specified by [vBoolean](#Section_8a0254e2d0bc4714be02f7104ff17c1f).

**ABNF:**

And = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " AND():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBoolcontaining the value of **Arg1**.

#### Ang360

The **Ang360** function normalizes an angle.

**ABNF:**

1. Ang360 = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " ANG360():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [vAngle](#Section_8609b9afc5aa4a0e9c37b1f7929ac9b4)

This function returns a vAngle containing an angle equivalent to **Arg1**, normalized to be greater than or equal to zero and less than 2\*pi, wherethe value of **Arg1** is assumed to have the unit of radians. If **Arg1** is a vAngle, the unit of the return value is equal to the unit of **Arg1**; otherwise the function returns a [PtgAngDft](#Section_57e444144344437fb32a6cf2d3b20d2f).

#### ASin

The **ASin** function performs the arcsine calculation.

**ABNF:**

1. ASin = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " ASIN():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgAngDft](#Section_57e444144344437fb32a6cf2d3b20d2f), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a PtgAngDft containing the arcsine of the value of **Arg1**. If **Arg1** is less than -1 or greater than 1, the function returns a PtgErr with an error code of #VALUE!.

#### ATan

The **ATan** function performs the arctangent calculation.

**ABNF:**

1. ATan = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " ATAN():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgAngDft](#Section_57e444144344437fb32a6cf2d3b20d2f)

This function returns a PtgAngDft containing the arctangent of the value of **Arg1**.

#### ATan2

The **ATan2** function calculates the angle between the positive x-axis and a vector.

**ABNF:**

1. ATan2 = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " ATAN2():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the y-component of the vector.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the x-component of the vector.

**Return Value:**

*Type:* [PtgAngDft](#Section_57e444144344437fb32a6cf2d3b20d2f)

This function returns a PtgAngDft containing the angle between the positive x-axis and the vector represented by **Arg1** and **Arg2**. The value is greater than -pi and less than or equal to pi. If **Arg1** is equal to zero and **Arg2** is equal to zero, the value is zero.

#### BitAnd

The **BitAnd** function performs the bitwise AND operation.

**ABNF:**

1. BitAnd = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " BITAND():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vUnsignedInt

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b)

This function returns a PtgUnsWord containing the value of the bitwise AND operation between **Arg1** and **Arg2**.

#### BitNot

The **BitNot** function performs the bitwise NOT operation.

**ABNF:**

1. BitNot = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " BITNOT():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b)

This function returns a PtgUnsWord containing the value of the bitwise NOT operation on **Arg1**.

#### BitOr

The **BitOr** function performs the bitwise OR operation.

**ABNF:**

1. BitOr = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " BITOR():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vUnsignedInt

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b)

This function returns a PtgUnsWord containing the value of the bitwise OR operation between **Arg1** and **Arg2**.

#### BitXor

The **BitXor** function performs the bitwise XOR operation.

**ABNF:**

1. BitXor = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " BITXOR():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vUnsignedInt

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b)

This function returns a PtgUnsWord containing the value of the bitwise XOR operation between **Arg1** and **Arg2**.

#### Blend

The **Blend** function performs a blend of two colors.

**ABNF:**

1. Blend = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val " BLEND():128"

**Required Arguments:**

*Name:* **Color1**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the first color.

*Name:* **Color2**

*Type:* vColor

An argument that specifies the second color.

*Name:* **Fraction**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the fractional amount of **Color2** in the blended color.

**Return Value:**

*Type:* [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a PtgColorRGB containing the blended color. If **Fraction** is less than 0 or greater than 1, the function returns a PtgErr with an error code of #VALUE!.

#### Bound

The **Bound** function constrains a value by one or more ranges.

**ABNF:**

1. Bound = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val 1\*(val val val) sp unsigned-integer-value ";BOUND():129"

**Required Arguments:**

*Name:* **Number**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the value to be constrained.

*Name:* **BoundType**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

An argument that specifies the bounding type. The valid values are described in the following table.

|  |  |
| --- | --- |
| Value | Meaning |
| 0 | Constrain **Number** inclusive of any of the ranges specified |
| 1 | Constrain **Number** exclusive of all the ranges specified |
| 2 | Do not constrain **Number** |

*Name:* **IgnoreRange**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

An argument that specifies if this range is not included in constraining **Number**.

*Name:* **ValueBeg**

*Type:* vDouble

An argument that specifies the beginning value of the range.

*Name:* **ValueEnd**

*Type:* vDouble

An argument that specifies the ending value of the range.

**Optional Arguments:**

Additional ranges MUST be specified using 1 or more additional groups of **IgnoreRange**, **ValueBeg**, and **ValueEnd**.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum containing the constrained value of **Number,** as described in the following table.

| Condition | Result |
| --- | --- |
| **BoundType** = 0 | If **Number** is in at least one range with **IgnoreRange** equal to FALSE, the function returns **Number.**  Otherwise, consider the set of all **ValueBeg** and **ValueEnd** values that belong to a range with **IgnoreRange** equal to FALSE. The function returns the value in this set that is closest to **Number**. If more than one value has the same minimum distance from **Number**, the function returns the value in this set that has minimum distance from **Number** and that appears earliest in the argument list. |
| **BoundType** = 1 | If **Number** is not in any range with **IgnoreRange** equal to FALSE, the function returns **Number**.  Otherwise, consider the set of all **ValueBeg** and **ValueEnd** values that belong to a range with **IgnoreRange** equal to FALSE and that are not nested inside another range with **IgnoreRange** equal to FALSE. The function returns the value in this set that is closest to **Number**. If **Number** is in the exact middle of a **ValueBeg** value and a **ValueEnd** value in this set, the function returns the **ValueEnd** value. |
| **BoundType** = 2 | The function returns **Number**. |
| **Otherwise** | The function returns a PtgErr with an error code of #VALUE!. |

If **Number** is a [vAngle](#Section_8609b9afc5aa4a0e9c37b1f7929ac9b4), the return value is normalized to be greater than or equal to zero and less than 2\*pi. The unit of the return value is equal to the unit of **Number**. If **Number** is a [ptgPnt](#Section_45fae81bb4e6413394e32daf5c98d790) or the wrong number of arguments is used, the function returns a PtgErr with an error code of #VALUE!.

#### Cat

The **Cat** function performs the concatenation of two strings.

**ABNF:**

1. Cat = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_CAT():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the first string.

*Name:* **Arg2**

*Type:* vString

An argument that specifies the second string.

**Return Value:**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69)

This function returns a PtgStr1 containing the concatenation of **Arg1** and **Arg2**.

#### Ceiling

The **Ceiling** function performs a ceiling calculation.

**ABNF:**

1. Ceiling = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";CEILING():129"

**Required Arguments:**

*Name:* **Number**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the value to be rounded.

**Optional Arguments:**

*Name:* **Multiple**

*Type:* vDouble

An argument that specifies the rounding increment. The default value is 1 if **Number** is greater than or equal to 0, and -1 if **Number** is less than 0.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum containing the next multiple of **Multiple** after **Number** that is further from zero. If **Number** is a multiple of **Multiple**, the value is equal to **Number**. If **Number** is equal to zero or **Multiple** is equal to zero, the value is zero. The unit of the return value is equal to the unit of **Number**. If **Number** and **Multiple** do not have the same sign, the function returns a PtgErr with an error code of #NUM!.

#### CellIsThemed

The **CellIsThemed** function MUST be ignored.

**ABNF:**

1. CellIsThemed = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " CELLISTHEMED():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vBoolean](#Section_8a0254e2d0bc4714be02f7104ff17c1f)

This argument is unused and MUST be ignored.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of FALSE.

#### Char

The **Char** function performs a conversion from an integer to the corresponding character in the [**American National Standards Institute (ANSI) character set**](#gt_100cd8a6-5cb1-4895-9de6-e4a3c224a583).

**ABNF:**

1. Char = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " CHAR():128"

**Required Arguments:**

*Name:* **Number**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

An argument that specifies an integer.

**Return Value:**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a PtgStr1 containing the character that corresponds to **Number**. If **Number** is less than 1 or greater than 255, the function returns a PtgErr with an error code of #VALUE!.

#### Cos

The **Cos** function performs the cosine calculation.

**ABNF:**

1. Cos = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " COS():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing the cosine of the value of **Arg1**, wherethe value of **Arg1** is assumed to have the unit of radians.

#### CosH

The **CosH** function performs the hyperbolic cosine calculation.

**ABNF:**

1. CosH = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " COSH():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing the hyperbolic cosine of **Arg1**, wherethe value of **Arg1** is assumed to have the unit of radians.

#### CY

The **CY** function performs conversion of a value to a currency.

**ABNF:**

1. CY = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) val val " 3;CY():129"

**Required Arguments:**

*Name:* **Number**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the value to be converted to a currency.

*Name:* **CyId**

*Type:* [vCurrencyID](#Section_b13bb6692bc04291b31c4532158fa675)

An argument that specifies the currency.

*Name:* **Format**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the formatting information. It MUST be a numeric format string, as described in [[MSDN-FormattingTypes]](https://go.microsoft.com/fwlink/?LinkId=153833).

**Return Value:**

*Type:* [PtgCY](#Section_a0035c3665bf4a2aa677a9d3810e85c3)

This function returns a PtgCY containing **Number**, **CyId,** and **Format**.

#### Date

The **Date** function returns a date, according to the Gregorian calendar, from values representing a year, month, and day.

**ABNF:**

1. Date = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) val val " DATE():128"

**Required Arguments:**

*Name:* **Year**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

An argument that specifies a year. A value between 0 and 29 corresponds to the range of years between 2000 and 2029. A value between 30 and 100 corresponds to the range of years between 1930 and 2000. A value greater than 100 corresponds to that year. It MUST be less than or equal to 9999.

*Name:* **Month**

*Type:* [vSignedInt](#Section_6e6498757ff54acfa20b54eb9d8cd472)

An argument that specifies an offset in months from December 1st of the previous year.

*Name:* **Day**

*Type:* vSignedInt

An argument that specifies an offset in days from the last day of the previous month.

**Return Value:**

*Type:* [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a PtgDate containing the date, according to the Gregorian calendar, represented by **Year**, **Month**, and **Day**. If the arguments specify a date before January 1st, 1899 or between January 1st, 1900 and November 30th, 1900 inclusively, the function returns a PtgErr with an error code of #NUM!. If either **Month** or **Day** cannot be interpreted as vSignedInt, the function returns a PtgErr with an error code of #VALUE!.

#### DateTime

The **DateTime** function converts a value to a date and time.

**ABNF:**

1. DateTime = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";DATETIME():129"

**Required Arguments:**

*Name:* **DateTimeArg**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies a value representing a date and time.

**Optional Arguments:**

*Name:* **Locale**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**language code identifier (LCID)**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) to use when parsing **DateTimeArg**.

**Return Value:**

*Type:* [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

If **DateTimeArg** is a [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), this function attempts to parse it using date and time format strings according to .NET globalization [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) for **Locale**. If **Locale** is not specified, the LCID is specified by the **DocumentLanguage** field of the Properties **GlobalElement** in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) of the document. For more information about .NET globalization rules, see [[MSDN-ENCLOC]](https://go.microsoft.com/fwlink/?LinkId=153669). If the string is successfully parsed, the function returns a PtgDate containing the parsed date and time. If the string is not successfully parsed, the function returns a PtgErr with an error code of #VALUE!.

If **DateTimeArg** is not a PtgStr1, the function returns a PtgDate containing the value of **DateTimeArg** interpreted as a [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d). If **DateTimeArg** cannot be interpreted as a vDouble, the function returns a PtgErr with an error code of #VALUE!.

#### DateValue

The **DateValue** function returns the date component from a value representing a date and time.

**ABNF:**

1. DateValue = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";DATEVALUE():129"

**Required Arguments:**

*Name:* **DateTimeArg**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies a value representing a date and time.

**Optional Arguments:**

*Name:* **Locale**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) to use when parsing **DateTimeArg**.

**Return Value:**

*Type:* [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function attempts a conversion, as described by the [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) function, of **DateTimeArg** to a PtgDate. If the conversion succeeds, the function returns a PtgDate containing the date component of **DateTimeArg**. If the conversion fails, the function returns a PtgErr with an error code of #VALUE!.

#### Day

The **Day** function returns the day of the month, according to the Gregorian calendar, from a value representing a date and time.

**ABNF:**

1. Day = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";DAY():129"

**Required Arguments:**

*Name:* **DateTimeArg**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies a date and time value.

**Optional Arguments:**

*Name:* **Locale**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) to use when parsing **DateTimeArg**.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function attempts a conversion, as described by the [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) function, of **DateTimeArg** to a [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6). If the conversion succeeds, the function returns a PtgUnsWord containing the day of the month component of **DateTimeArg.** The value is greater than or equal to 1 and less than or equal to 31. If the conversion fails, the function returns a PtgErr with an error code of #VALUE!.

#### DayOfYear

The **DayOfYear** function returns the day of the year, according to the Gregorian calendar, from a value representing a date and time.

**ABNF:**

1. DayOfYear = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";DAYOFYEAR():129"

**Required Arguments:**

*Name:* **DateTimeArg**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies a date and time value.

**Optional Arguments:**

*Name:* **Locale**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) to use when parsing **DateTimeArg**.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function attempts a conversion, as described by the [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) function, of **DateTimeArg** to a [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6). If the conversion succeeds, the function returns a PtgUnsWord containing the day of the year component from **DateTimeArg.** The value is greater than or equal to 1 and less than or equal to 366. If the conversion fails, the function returns a PtgErr with an error code of #VALUE!.

#### Deg

The **Deg** function performs conversion to an angle in degrees.

**ABNF:**

1. Deg = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " DEG():128"

**Required Arguments:**

*Name:* **Angle**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies an angle.

**Return Value:**

*Type:* [PtgAngDD](#Section_222188be55a34b90b2c39eb9f5878bab)

This function returns a PtgAngDD containing the value of **Angle**.

#### Div

The **Div** function performs a division calculation.

**ABNF:**

1. Div = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_DIV():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDoubleEx](#Section_17ab59629f9e4b75a84af0cc80400b88)

An argument that specifies the first operand of the division operation.

*Name:* **Arg2**

*Type:* vDoubleEx

An argument that specifies the second operand of the division operation.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum or a PtgCy containing the result of **Arg1** divided by **Arg2**. The type of the return value is calculated by the following algorithm.

SET returnType = [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

SET returnError = #VALUE!

SET returnDimension = 0

SET returnCurrencyID = 0

IF Arg2.Value = 0 THEN

SET returnType = PtgErr

SET returnError = #DIV/0

ELSE IF Arg1.Type = PtgCy AND Arg2.Type != PtgCy THEN

SET returnType = PtgCy

SET returnCurrencyID = currencyID of Arg1

ELSE IF Arg1.Type != PtgCy AND Arg2.Type = PtgCy THEN

SET returnType = PtgCy

SET returnCurrencyID = currencyID of Arg2

ELSE IF Arg1.Type = PtgCy AND Arg2.Type = PtgCy THEN

SET returnType = PtgCy

IF currencyID of Arg1 = currencyID of Arg2 OR currencyID of Arg2 = 1 THEN

SET returnCurrencyID = currencyID of Arg1

ELSE IF currencyID of Arg1 = 1 THEN

SET returnCurrencyID = currencyID of Arg2

ELSE

SET returnType = PtgErr

SET returnError = #VALUE!

END IF

ELSE IF Arg1.Unit is a [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b) AND Arg2.Unit is not a vUnitType THEN

SET returnType = Arg1.Unit

SET returnDimension = Arg1.Dimension

ELSE IF Arg1.Unit is not a vUnitType AND Arg2.Unit is a vUnitType THEN

SET returnType = Arg2.Unit

SET returnDimension = Arg2.Dimension

ELSE

IF Arg1.Unit = [PtgNumPct](#Section_ef65a2675fd84ba3a1448fb7698f6551) AND Arg2.Unit = PtgNumPct THEN

SET returnDimension = 1

ELSE IF Arg1.Unit = PtgNumPct THEN

SET returnDimension = -Arg2.Dimension

ELSE IF Arg2.Unit = PtgNumPct THEN

SET returnDimension = Arg1.Dimension

ELSE

SET returnDimension = Arg1.Dimension – Arg2.Dimension

END IF

IF Arg1.Unit is a vUnitType THEN

SET returnType = Arg1.Unit

ELSE

SET returnType = Arg2.Unit

END IF

END IF

IF returnType = PtgCy THEN

Return type is PtgCy with currencyID = returnCurrencyID

ELSE IF returnType = PtgErr THEN

Return type is PtgErr with error code of returnError

ELSE IF (returnType = [PtgAcre](#Section_7f6eff5e434a407b8565fe966e0b066c) OR returnType = [PtgHectare](#Section_7b405e52f9c34451a7f90ad394edd4aa)) AND returnDimension != 2 THEN

Return type is [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84) with unit = [PtgNumDft](#Section_de132a3cdab04751844609cf15ec3756) and dimension = returnDimension

ELSE IF returnDimension = 0 THEN

Return type is PtgNum

ELSE IF returnDimension = 1 THEN

Return type is returnType

ELSE

Return type is PtgNumMultiDim with unit = returnType and dimension = returnDimension

END IF

#### EEQ

The **EEQ** function calculates if **Arg1** is equal to **Arg2**.

**ABNF:**

1. EEQ = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_EEQ():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Arg1** is equal to **Arg2**, and a value of FALSE otherwise.

#### EGE

The **EGE** function calculates if **Arg1** is greater than or equal to **Arg2**.

**ABNF:**

1. EGE = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_EGE():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Arg1** is greater than or equal to **Arg2**, and a value of FALSE otherwise.

#### EGT

The **EGT** function calculates if **Arg1** is greater than **Arg2**.

**ABNF:**

1. EGT = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_EGT():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Arg1** is greater than **Arg2**, and a value of FALSE otherwise.

#### ELE

The **ELE** function calculates if **Arg1** is less than or equal to **Arg2**.

**ABNF:**

1. ELE = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_ELE():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Arg1** is less than or equal to **Arg2**, and a value of FALSE otherwise.

#### ELT

The **ELT** function calculates if **Arg1** is less than **Arg2**.

**ABNF:**

1. ELT = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_ELT():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Arg1** is less than **Arg2**, and a value of FALSE otherwise.

#### ENE

The **ENE** function calculates if **Arg1** is not equal to **Arg2**.

**ABNF:**

1. ENE = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_ENE():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Arg1** is not equal to **Arg2**, and a value of FALSE otherwise.

#### FEQ

The **FEQ** function calculates if the absolute value of the difference between **Arg1** and **Arg2** is less thanor equal to1E-9 (0.000000001).

**ABNF:**

1. FEQ = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_FEQ():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if the absolute value of the difference between **Arg1** and **Arg2** is less than or equal to 1E-9, and a value of FALSE otherwise.

#### FGE

The **FGE** function calculates if the difference between **Arg1** and **Arg2** is greater than or equal to -1E-9 (-0.000000001).

**ABNF:**

1. FGE = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_FGE():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if the difference between **Arg1** and **Arg2** is greater than or equal to -1E-9, and a value of FALSE otherwise.

#### FGT

The **FGT** function calculates if **Arg1** is greater than **Arg2** by at least the amount of 2-30.

**ABNF:**

1. FGT = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_FGT():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Arg1** is greater than **Arg2** by at least the amount of 2-30, and a value of FALSE otherwise.

#### Find

The **Find** function performs a search for the first instance of a text string within another text string.

**ABNF:**

1. Find = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val [val [val]] sp ("2" / "3" / "4") ";FIND():129"

**Required Arguments:**

*Name:* **FindText**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the string to be found.

*Name:* **WithinText**

*Type:* vString

An argument that specifies the string to search within.

**Optional Arguments:**

*Name:* **StartNum**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

An argument that specifies the one-based position in the **WithinText** string at which this function starts a search. The default value is one.

*Name:* **IgnoreCase**

*Type:* [vBoolean](#Section_8a0254e2d0bc4714be02f7104ff17c1f)

An argument that specifies whether the search is case insensitive. A value of TRUE specifies that case is ignored. The default value is FALSE.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a PtgUnsWord containing the one-based position in **WithinText** at which **FindText** is found. If **FindText** is empty and **StartNum** is less than or equal to the number of characters in **WithinText**, the function returns a PtgUnsWord containing the value of **StartNum**. If **FindText** is empty and **StartNum** is greater than the number of characters in **WithinText**, the function returns a PtgUnsword containing the value of the number of characters in **WithinText** plus one. If **FindText** is not found, or, if **StartNum** is equal to zero or greater than the number of characters in **WithinText**, the function returns a PtgErr with an error code of #VALUE!. The search is performed according to .NET globalization [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) based on the [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) in the **DocumentLanguage** field of the Properties **GlobalElement** in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) of the document. For more information about .NET globalization rules, see [[MSDN-ENCLOC]](https://go.microsoft.com/fwlink/?LinkId=153669).

#### FLE

The **FLE** function calculates if the difference between **Arg1** and **Arg2** is less than or equal to 1E-9 (0.000000001).

**ABNF:**

1. FLE = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_FLE():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if the difference between **Arg1** and **Arg2** is less than or equal to 1E-9, and a value of FALSE otherwise.

#### Floor

The **Floor** function performs a floor calculation.

**ABNF:**

1. Floor = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";FLOOR():129"

**Required Arguments:**

*Name:* **Number**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the value to be rounded.

**Optional Arguments:**

*Name:* **Multiple**

*Type:* vDouble

An argument that specifies the rounding increment.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

If **Multiple** is not specified, the function returns a vNum containing the largest integer less than or equal to **Number**. If **Multiple** is greater than zero, the function returns a vNum containing the largest multiple of **Multiple** less than or equal to **Number**. If **Multiple** is less than zero, the function returns a vNum containing the smallest multiple of **Multiple** greater than or equal to **Number**. If **Number** is equal to zero or **Multiple** is equal to zero, the value is zero. The unit of the return value is equal to the unit of **Number**. If **Number** and **Multiple** do not have the same sign, the function returns a PtgErr with an error code of #NUM!.

#### FLT

The **FLT** function calculates if **Arg1** is less than **Arg2** by at least the amount of 2-30.

**ABNF:**

1. FLT = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_FLT():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Arg1** is less than **Arg2** by at least the amount of 2-30, and a value of FALSE otherwise.

#### FNE

The **FNE** function calculates if **Arg1** differs from **Arg2** by at least the amount of 2-30.

**ABNF:**

1. FNE = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_FNE():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDouble

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Arg1** differs from **Arg2** by at least the amount of 2-30, and a value of FALSE otherwise.

#### FormatEx

The **FormatEx** function formats a value as a string.

**ABNF:**

1. FormatEx = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val val val val " FORMATEX():128"

**Required Arguments:**

*Name:* **FormatValue**

*Type:* [vDoubleEx](#Section_17ab59629f9e4b75a84af0cc80400b88)

An argument that specifies the value to be formatted.

*Name:* **Format**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69)

An argument that specifies the formatting information used to format **FormatValue**. It MUST have the following format.

"*Formatting*@*ServerAction*@*UnitLabel*@*CurrencyID*"

The elements of the string are separated by "@" and are defined in the following paragraphs.

*Formatting:* A [vFormatString](#Section_3d8a401ea64247d082ad4cfbdd2331b9) that specifies the formatting information.

If *Formatting* is an empty string, a default value from the following table is used.

| Value | Meaning |
| --- | --- |
| "{0} {1}" | Default string format.  This is used if the type of the source token of **FormatValue** is PtgStr1, [PtgEDay](#Section_f6690609c80a49ed9fffe0dbcfb262a6), [PtgEHour](#Section_e31838fa14684d478f193c033df45e4c), [PtgEMin](#Section_f58cc7ec8a1246d89edb4b3e00a08ac6), [PtgESec](#Section_01c1da7178fc485c8ee6b48674eeb35b), [PtgEWeek](#Section_4aae0088510448abaf3f8ec63f95fd5f), or [PtgTDurDft](#Section_1200eb57709c4fcc9d20e2bac20fa493). |
| "{0:0.####} {1}" | Default numeric format for numbers with less than 16 digits to the left of the decimal point.  This is used if the type of the source token of **FormatValue** is [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd), [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0), [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b), or a [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b) other than [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6), PtgEDay, PtgEHour, PtgEMin, PtgESec, PtgEWeek, and PtgTDurDft. |
| "{0:0.####e0} {1}" | Default numeric format for numbers with 16 or more digits to the left of the decimal point.  This is used if the type of the source token of **FormatValue** is PtgBool, PtgCy, PtgMissArg, PtgNum, PtgUnsWord, or a vUnitType other than PtgDate, PtgEDay, PtgEHour, PtgEMin, PtgESec, PtgEWeek, and PtgTDurDft. |
| "G" | Default date and time format when the time value is not equal to zero.  This is used if the type of the source token of **FormatValue** is PtgDate. |
| "d" | Default date and time format when the time value is equal to zero.  This is used if the type of the source token of **FormatValue** is PtgDate. |

If *Formatting* is not empty and **FormatValue** is a PtgDate, *Formatting* MUST be a vFormatString with valid syntax for formatting a date and time value. If *Formatting* is not empty and **FormatValue** is not a PtgDate, *Formatting* MUST be a vFormatString with valid syntax for formatting a numeric value.

*ServerAction*: A [vServerAction](#Section_094c818d877844b881c7dd957e2aaf7f) that specifies the custom formatting to apply to a string after all other formatting has been performed.

*UnitLabel*: A [vUnitLabel](#Section_b8166c01556f4257b5bcc081923ee37d) that specifies the [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) for formatting the unit of the resulting string.

*CurrencyID*: A [vCurrencyID](#Section_b13bb6692bc04291b31c4532158fa675) that specifies the currency identifier.

*Name:* **SrcUnit**

*Type:* vUnitType

An argument that specifies the unit of **FormatValue**. The value is unused and MUST be ignored. If **FormatValue** is a vUnitType, this argument is unused and MUST be ignored.

*Name:* **DstUnit**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad)

An argument that specifies the unit displayed in the resulting string. The value is unused and MUST be ignored. If the type is PtgNum, the type of **SrcUnit** is used.

*Name:* **LangID**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) used to format **FormatValue**.

*Name:* **CalID**

*Type:* [vCalendar](#Section_4c2dd2c033874a6c99af9b277c89b411)

An argument that specifies a calendar system used to format **FormatValue**. If **FormatValue** is not a PtgDate, this argument is unused and MUST be ignored.

**Return Value:**

*Type:* PtgStr1, [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a PtgStr1 containing **FormatValue**, converted into the unit specified by **DstUnit**, formatted as a string as specified by **Format**, **LangID**,and **CalID**.

**FormatValue** is converted into the unit specified by **DstUnit** as follows. If **FormatValue** is not a vUnitType, the value of **FormatValue** is multiplied by a factor that converts the unit of **SrcUnit** into the [custom internal unit type](#Section_1ef4d381d8284ad094e4227751b7b6d2) associated with the type of **SrcUnit**. The resulting value, or the value of **FormatValue** if **FormatValue** is a vUnitType, is multiplied by a factor that converts the custom internal unit type associated with the type of **DstUnit** into the unit of **DstUnit**

If **FormatValue** is a PtgDate, **FormatValue** is formatted using date and time format strings, as described in [[MSDN-FormattingTypes]](https://go.microsoft.com/fwlink/?LinkId=153833). If **FormatValue** is not a PtgDate, the value of **FormatValue** is formatted using numeric format strings, as described in [MSDN-FormattingTypes].

If the formatting does not succeed or an argument is invalid, the function returns a PtgErr with an error code of #VALUE!.

#### Hour

The **Hour** function returns the hour from a value representing a date and time.

**ABNF:**

1. Hour = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";HOUR():129"

**Required Arguments:**

*Name:* **DateTimeArg**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies a date and time value.

**Optional Arguments:**

*Name:* **Locale**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) to use when parsing **DateTimeArg**.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function attempts a conversion, as described by the [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) function, of **DateTimeArg** to a [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6). If the conversion succeeds, the function returns a PtgUnsWord containing the hour component from **DateTimeArg.** The value is greater than or equal to zero and less than or equal to 23. If the conversion fails, the function returns a PtgErr with an error code of #VALUE!.

#### HSL

The **HSL** function transforms a color specified in the [**hue-saturation-luminance (HSL)**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5) into a color specified in the [**RGB**](#gt_2c716d3a-e60b-4e52-bbb0-2fdeb298003b) color space. **HSL** component values are in the range 0 to 240, inclusive.

**ABNF:**

1. HSL = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val " HSL():128"

**Required Arguments:**

*Name:* **Hue**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

An argument that specifies the hue component.

*Name:* **Saturation**

*Type:* vUnsignedInt

An argument that specifies the saturation component.

*Name:* **Luminance**

*Type:* vUnsignedInt

An argument that specifies the luminance component.

**Return Value:**

*Type:* [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882)

This function returns a PtgColorRGB containing the color value specified by **Hue**, **Saturation** and **Luminance**. If **Hue** is greater than 240, this function sets the value to 240 and performs the operation.

#### Hue

The **Hue** function calculates the hue component, as specified in the [**HSL**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5), of a color value.

**ABNF:**

1. Hue = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " HUE():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the color.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b)

This function returns a PtgUnsWord containing the hue of **Arg1**. The value is less than or equal to 240.

#### HueDiff

The **HueDiff** function calculates the difference in hue, as specified in the [**HSL**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5), between two color values.

**ABNF:**

1. HueDiff = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " HUEDIFF():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the first color.

*Name:* **Arg2**

*Type:* vColor

An argument that specifies the second color.

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing a value that is equal to the hue of **Arg1** minus the hue of **Arg2**. This value is greater than or equal to -240 and less than or equal to 240.

#### Index

The **Index** function performs a search of a delimited text string for the substring specified by a [**zero-based index**](#gt_bc60c405-d92b-4a8c-b63b-e404b1cc4dc4).

**ABNF:**

1. Index = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val [val [val]] sp ("2" / "3" / "4") ";INDEX():129"

**Required Arguments:**

*Name:* **IndexPosition**

*Type:* [vSignedInt](#Section_6e6498757ff54acfa20b54eb9d8cd472)

An argument that specifies a zero-based index.

*Name:* **List**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies a string consisting of a list of delimited substrings. If the list begins and/or ends with **Delimiter**, an empty string is assumed to exist at the beginning and/or end of **List**, respectively. Consecutive delimiters indicate an empty string between them.

**Optional Arguments:**

*Name*: **Delimiter**

*Type*: vString

An argument that specifies the delimiter that separates substrings inside **List**. If this argument is an empty string, the default value ";" is used.

*Name*: **ErrorValue**

*Type*: vString

An argument that specifies a string to be returned if **Index** is out of range. The default value is an empty string.

**Return Value:**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69)

This function returns a PtgStr1 equal to the substring at the zero-based index location **IndexPosition** in **List**, which is delimited by **Delimiter**. If **IndexPosition** is less than zero or greater than or equal to the number of items in **List**, the function returns a PtgStr1 that specifies **ErrorValue**. The search is performed according to .NET globalization [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) based on the [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) in the **DocumentLanguage** field of the Properties **GlobalElement** in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) of the document. For more information about .NET globalization rules, see [[MSDN-ENCLOC]](https://go.microsoft.com/fwlink/?LinkId=153669).

#### Int

The **Int** function performs a rounding calculation down to the next lesser integer value.

**ABNF:**

1. Int = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) " INT():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the value to be rounded.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad)

This function returns a vNum containing the next lesser integer value if **Arg1** is not an integer. The unit of the return value is equal to the unit of **Arg1.**

#### IntersectX

The **IntersectX** function calculates the x-coordinate of the point where two lines intersect. Each line is defined by a point on the line and an angle.

**ABNF:**

1. IntersectX = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val val val val " INTERSECTX():128"

**Required Arguments:**

*Name:* **X1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the x-coordinate of the point defining the first line.

*Name:* **Y1**

*Type:* vDouble

An argument that specifies the y-coordinate of the point defining the first line.

*Name:* **Angle1**

*Type:* vDouble

An argument that specifies the angle of the first line relative to the positive x-axis.

*Name:* **X2**

*Type:* vDouble

An argument that specifies the x-coordinate of the point defining the second line.

*Name:* **Y2**

*Type:* vDouble

An argument that specifies the y-coordinate of the point defining the second line.

*Name:* **Angle2**

*Type:* vDouble

An argument that specifies the angle of the second line relative to the positive x-axis.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum containing the x-coordinate of the point where two lines intersect. If any of the arguments **X1**, **Y1**, **X2**, **Y2** is a [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b), the unit of the return value is equal to the unit of the first of those arguments that is a vUnitType. If none of those arguments is a vUnitType, the function returns a [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0). If **Angle1** equals **Angle2**, the function returns a PtgErr with an error code of #DIV/0.

#### IntersectY

The **IntersectY** function calculates the y-coordinate of the point where two lines intersect. Each line is defined by a point on the line and an angle.

**ABNF:**

1. IntersectY = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val val val val " INTERSECTY():128"

**Required Arguments:**

*Name:* **X1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the x-coordinate of the point defining the first line.

*Name:* **Y1**

*Type:* vDouble

An argument that specifies the y-coordinate of the point defining the first line.

*Name:* **Angle1**

*Type:* vDouble

An argument that specifies the angle of the first line relative to the positive x-axis.

*Name:* **X2**

*Type:* vDouble

An argument that specifies the x-coordinate of the point defining the second line.

*Name:* **Y2**

*Type:* vDouble

An argument that specifies the y-coordinate of the point defining the second line.

*Name:* **Angle2**

*Type:* vDouble

An argument that specifies the angle of the second line relative to the positive x-axis.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum containing the y-coordinate of the point where two lines intersect. If any of the arguments **X1**, **Y1**, **X2**, **Y2** is a [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b), the unit of the return value is equal to the unit of the first of those arguments that is a vUnitType. If none of those arguments is a vUnitType, the function returns a [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0). If **Angle1** equals **Angle2**, the function returns a PtgErr with an error code of #DIV/0.

#### Intup

The **Intup** function performs a rounding calculation up to the next greater integer value.

**ABNF:**

1. Intup = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " INTUP():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the value to be rounded.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad)

This function returns a vNum containing the next greater integer value if **Arg1** is not an integer. The unit of the return value is equal to the unit of **Arg1**.

#### IsErr

The **IsErr** function calculates if the operand is a [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2) that does not have an error code of #N/A.

**ABNF:**

1. IsErr = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " ISERR():128"

**Required Arguments:**

*Name:* **Token**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Token** is a PtgErr and the value of **Token** is not equal to #N/A, and a value of FALSE otherwise.

#### IsErrNA

The **IsErrNA** function calculates if the operand is a [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2) with an error code of #N/A.

**ABNF:**

1. IsErrNa = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " ISERRNA():128"

**Required Arguments:**

*Name:* **Token**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Token** is a PtgErr and the value of **Token** is equal to #N/A, and a value of FALSE otherwise.

#### IsError

The **IsError** function calculates if the operand is a [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2).

**ABNF:**

1. IsError = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " ISERROR():128"

**Required Arguments:**

*Name:* **Token**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Token** is a PtgErr, and a value of FALSE otherwise.

#### IsErrValue

The **IsErrValue** function calculates if the operand is a [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2) with an error code of #VALUE!.

**ABNF:**

1. IsErrValue = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " ISERRVALUE():128"

**Required Arguments:**

*Name:* **Token**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool with a value of TRUE if **Token** is a PtgErr and the value of **Token** is equal to #VALUE!, and a value of FALSE otherwise.

#### Left

The **Left** function returns the first character or characters in a text string.

**ABNF:**

1. Left = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";LEFT():129"

**Required Arguments:**

*Name:* **Text**

*Type:*  [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the text string.

**Optional Arguments:**

*Name*: **NumChars**

*Type*: [vSignedLong](#Section_52e79e59b9eb4e8a8a7e92d0cdbdecf8)

An argument that specifies the number of characters to be returned. The default value is one.

**Return Value:**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69)

This function returns a PtgStr1 containing the first **NumChars** characters of **Text.** If **NumChars** is less than zero or greater than the number of characters in **Text**, the function returns a PtgStr1 containing **Text**.

#### Len

The **Len** function performs a calculation of the length of a string.

**ABNF:**

1. Len = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " LEN():128"

**Required Arguments:**

*Name:* **Text**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies a string.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b)

This function returns a PtgUnsWord containing the number of characters in **Text**.

#### Ln

The **Ln** function performs the natural logarithm calculation.

**ABNF:**

1. Ln = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " LN():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [[vNum](#Section_ad426b3dbea2407c8687202b9d13adad)](#Section_1d4fd8bc32844cb5915e14424a11a789), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum containing the natural logarithm of **Arg1**. The unit of the return value is equal to the unit of **Arg1**. If **Arg1** is less than or equal to zero, the function returns a PtgErr with an error code of #NUM!.

#### Log10

The **Log10** function performs the base 10 logarithm calculation.

**ABNF:**

1. Log10 = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " LOG10():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum containing the base 10 logarithm of **Arg1**. The unit of the return value is equal to the unit of **Arg1**. If the value of **Arg1** is less than or equal to zero, the function returns a PtgErr with an error code of #NUM!.

#### Lookup

The **Lookup** function calculates the index location of a substring within a delimited text string.

**ABNF:**

1. Lookup = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val [val] sp ("2" / "3") ";LOOKUP():129"

**Required Arguments:**

*Name:* **Key**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the substring to find.

*Name:* **List**

*Type:* vString

An argument that specifies a string consisting of a list of delimited substrings. If the list begins and/or ends with **Delimiter**, an empty string is assumed to exist at the beginning and/or end of **List**, respectively. Consecutive delimiters indicate an empty string between them.

**Optional Arguments:**

*Name*: **Delimiter**

*Type*: vString

An argument that specifies the delimiter that separates substrings inside **List**. If this argument is an empty string, the default value ";" is used.

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing the [**zero-based index**](#gt_bc60c405-d92b-4a8c-b63b-e404b1cc4dc4) location of the substring in **List** that matches **Key**. The leading and trailing spaces of substrings in **List** are ignored for the comparison. If **Key** contains **Delimiter** or if **Key** is not found in **List**, the function returns a PtgNum containing a value of -1. The comparison is performed according to .NET globalization [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) based on the [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) in the **DocumentLanguage** field of the Properties **GlobalElement** in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) of the document. For more information about .NET globalization rules, see [[MSDN-ENCLOC]](https://go.microsoft.com/fwlink/?LinkId=153669).

#### Lower

The **Lower** function performs lower case conversion.

**ABNF:**

1. Lower = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " LOWER():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the string to convert.

**Return Value:**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69)

This function returns a PtgStr1 containing the string converted to lower case. The conversion is performed according to .NET globalization [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) based on the [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) in the **DocumentLanguage** field of the Properties **GlobalElement** in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) of the document. For more information about .NET globalization rules, see [[MSDN-ENCLOC]](https://go.microsoft.com/fwlink/?LinkId=153669).

#### Lum

The **Lum** function calculates the luminance component, as specified in the [**HSL**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5), of a color value.

**ABNF:**

1. Lum = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " LUM():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the color.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b)

This function returns a PtgUnsWord containing the luminance of **Arg1**. The value is less than or equal to 240.

#### LumDiff

The **LumDiff** function calculates the difference in luminance, as specified in the [**HSL**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5), between two color values.

**ABNF:**

1. LumDiff = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " LUMDIFF():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the first color.

*Name:* **Arg2**

*Type:* vColor

An argument that specifies the second color.

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing a value that is equal to the luminance of **Arg1** minus the luminance of **Arg2**. This value is greater than or equal to -240 and less than or equal to 240.

#### Magnitude

The **Magnitude** function performs a magnitude calculation of a vector.

**ABNF:**

1. Magnitude = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val val " MAGNITUDE():128"

**Required Arguments:**

*Name:* **ConstantRise**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the constant factor for the rise.

*Name:* **Rise**

*Type:* vDouble

An argument that specifies the rise of the vector.

*Name:* **ConstantRun**

*Type:* vDouble

An argument that specifies the constant factor for the run.

*Name:* **Run**

*Type:* vDouble

An argument that specifies the run of the vector.

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

The function returns a PtgNum containing the calculated magnitude. The magnitude calculation is performed using the following formula:

SQRT( (ConstantRise\*Rise)^2 + (ConstantRun\*Run)^2) )

If **Rise** or **Run** is a [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84), the function returns a PtgErr with an error code of #DIM!.

#### Max

The **Max** function performs a search for the largest number from a list of operands.

**ABNF:**

1. Max = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) \*val sp unsigned-integer-value ";MAX():129"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand.

**Optional Arguments:**

Zero or more operands of type vDouble.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad)

This function returns a vNum containing the first occurrence of the largest operand. The unit of the return value is equal to the unit of that operand.

#### Mid

The **Mid** function returns part of a string, starting at the position specified, based on the number of characters specified.

**ABNF:**

1. Mid = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val " MID():128"

**Required Arguments:**

*Name:* **Text**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the text string.

*Name:* **StartPos**

*Type:* [vSignedLong](#Section_52e79e59b9eb4e8a8a7e92d0cdbdecf8)

An argument that specifies the one-based index in **Text** which represents the beginning of the substring to be returned.

*Name:* **NumChars**

*Type:* vSignedLong

An argument that specifies the number of characters to be returned.

**Return Value:**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a PtgStr1 containing a substring of **Text**, starting from **StartPos** up to the character at **StartPos** plus **NumChars** minus one. If **StartPos** is zero, the function returns a PtgErr with an error code of #VALUE!. If **StartPos** is less than zero or greater than the number of characters in **Text**, the function returns a PtgStr1 containing an empty string. If **StartPos** plus **NumChars** minus one exceeds the length of **Text**,or if **NumChars** is less than zero, the function returns a PtgStr1 containing the substring of **Text** starting from **StartPos** up to the end of **Text**.

#### Min

The **Min** function performs a search for the smallest number from a list of operands.

**ABNF:**

1. Min = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) \*val sp unsigned-integer-value ";MIN():129"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand.

**Optional Arguments:**

Zero or more operands of type vDouble.

**Return Value:**

*Type:* [[vNum](#Section_ad426b3dbea2407c8687202b9d13adad)](#Section_1d4fd8bc32844cb5915e14424a11a789)

This function returns a vNum containing the first occurrence of the smallest operand. The unit of the return value is equal to the unit of that operand.

#### Minute

The **Minute** function returns the minute from a value representing a date and time.

**ABNF:**

1. Minute = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";MINUTE():129"

**Required Arguments:**

*Name:* **DateTimeArg**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies a date and time value.

**Optional Arguments:**

*Name:* **Locale**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) to use when parsing **DateTimeArg**.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function attempts a conversion, as described by the [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) function, of **DateTimeArg** to a [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6). If the conversion succeeds, the function returns a PtgUnsWord containing the minute component from **DateTimeArg.** The value is greater than or equal to zero and less than or equal to 59. If the conversion fails, the function returns a PtgErr with an error code of #VALUE!.

#### Modulus

The **Modulus** function performs a modulus calculation.

**ABNF:**

1. Modulus = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " MODULUS():128"

**Required Arguments:**

*Name:* **Number**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the number to be divided by **Divisor**.

*Name:* **Divisor**

*Type:* [vFloat](#Section_1d9b3dd5ba94418ba9853237ae1cf224)

An argument that specifies the divisor of the calculation.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum containing the remainder when **Number** is divided by **Divisor**. If **Number** is a vNum, the unit of the return value is equal to the unit of **Number**; otherwise the function returns a [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0). If **Divisor** is zero, the function returns a PtgErr with an error code of #DIV/0.

#### Month

The **Month** function returns the month, according to the Gregorian calendar, from a value representing a date and time.

**ABNF:**

1. Month = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";MONTH():129"

**Required Arguments:**

*Name:* **DateTimeArg**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies a date and time value.

**Optional Arguments:**

*Name:* **Locale**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) to use when parsing **DateTimeArg**.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function attempts a conversion, as described by the [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) function, of **DateTimeArg** to a [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6). If the conversion succeeds, the function returns a PtgUnsWord containing the month component from **DateTimeArg**. The value is greater than or equal to 1 and less than or equal to 12. If the conversion fails, the function returns a PtgErr with an error code of #VALUE!.

#### MsoShade

The **MsoShade** function performs a modification to a color by decreasing the luminance component, as specified in the [**HSL**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5), by a percentage.

**ABNF:**

1. MsoShade = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " MSOSHADE():128"

**Required Arguments:**

*Name:* **Color**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the color.

*Name:* **Delta**

*Type:* [vSignedInt](#Section_6e6498757ff54acfa20b54eb9d8cd472)

An argument that specifies the percentage to decrease the luminance of **Color**. If the value is less than zero, the luminance is increased.

**Return Value:**

*Type:* [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882)

This function returns a PtgColorRGB containing the shaded color. If **Delta** is greater than zero, the luminance of **Color** is decreased by the percentage specified by the absolute value of **Delta**. If **Delta** is less than zero, the luminance of **Color** is increased by the following amount: the percentage specified by **Delta** multiplied by the result of 240 minus the luminance of **Color**. If **Delta** is less than -100, the function sets the value to -100 and performs the operation. If **Delta** is greater than 100, the function sets the value to 100 and performs the operation.

#### MsoTint

The **MsoTint** function modifies a color by increasing the luminance component, as specified in the [**HSL**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5), by a percentage.

**ABNF:**

1. MsoTint = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " MSOTINT():128"

**Required Arguments:**

*Name:* **Color**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the color.

*Name:* **Delta**

*Type:* [vSignedInt](#Section_6e6498757ff54acfa20b54eb9d8cd472)

An argument that specifies the percentage to increase the luminance of **Color**. If the value is less than zero, the luminance is decreased.

**Return Value:**

*Type:* [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882)

This function returns a PtgColorRGB containing the tinted color. If **Delta** is greater than zero, the luminance of **Color** is increased by the following amount: the percentage specified by **Delta** multiplied by the result of 240 minus the luminance of **Color**. If **Delta** is less than zero, the luminance of **Color** is decreased by the percentage specified by the absolute value of **Delta**. If **Delta** is less than -100, the function sets the value to -100 and performs the operation. If **Delta** is greater than 100, the function sets the value to 100 and performs the operation.

#### Mul

The **Mul** function performs a multiplication calculation.

**ABNF:**

1. Mul = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_MUL():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDoubleEx](#Section_17ab59629f9e4b75a84af0cc80400b88)

An argument that specifies the first operand of the multiplication operation.

*Name:* **Arg2**

*Type:* vDoubleEx

An argument that specifies the second operand of the multiplication operation.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum or a PtgCy containing **Arg1** multiplied by **Arg2**. The type of the return value is calculated by the following algorithm.

SET return type = [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

SET returnDimension = 0

SET returnCurrencyID = 0

IF Arg1.Type = PtgCy AND Arg2.Type != PtgCy THEN

SET returnType = PtgCy

SET returnCurrencyID = currencyID of Arg1

ELSE IF Arg1.Type != PtgCy AND Arg2.Type = PtgCy THEN

SET returnType = PtgCy

SET returnCurrencyID = currencyID of Arg2

ELSE IF Arg1.Type = PtgCy AND Arg2.Type = PtgCy THEN

SET returnType = PtgCy

IF currencyID of Arg1 = currencyID of Arg2 OR currencyID of Arg2 = 1 THEN

SET returnCurrencyID = currencyID of Arg1

ELSE IF currencyID of Arg1 = 1 THEN

SET returnCurrencyID = currencyID of Arg2

ELSE

SET returnType = PtgErr

END IF

ELSE IF Arg1.Unit = [PtgNumPct](#Section_ef65a2675fd84ba3a1448fb7698f6551) AND Arg2.Unit = PtgNumPct

SET returnType = PtgNumPct

SET returnDimension = 1

ELSE IF Arg1.Unit = PtgNumPct

SET returnType = PtgNumPct

SET returnDimension = Arg2.dimension

ELSE IF Arg2.Unit = PtgNumPct

IF Arg1.Unit is a [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b)

SET returnType = Arg1.Unit

SET returnDimension = Arg1.Dimension

ELSE

SET returnType = Arg2.Unit

SET returnDimension = Arg1.Dimension

END IF

ELSE IF Arg1.Unit is a vUnitType THEN

SET returnType = Arg1.Unit

SET returnDimension = Arg1.Dimension + Arg2.Dimension

ELSE IF Arg2.Unit is a vUnitType THEN

SET returnType = Arg2.Unit

SET returnDimension = Arg1.Dimension + Arg2.Dimension

END IF

IF returnType = PtgCy THEN

Return type is PtgCy with currencyID = returnCurrencyID

ELSE IF returnType = PtgErr THEN

Return type is PtgErr with error code of #VALUE!

ELSE IF (returnType = [PtgAcre](#Section_7f6eff5e434a407b8565fe966e0b066c) OR returnType = [PtgHectare](#Section_7b405e52f9c34451a7f90ad394edd4aa)) AND returnDimension != 2 THEN

Return type is [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84) with unit = [PtgNumDft](#Section_de132a3cdab04751844609cf15ec3756) and dimension = returnDimension

ELSE IF returnDimension = 0 THEN

Return type is PtgNum

ELSE IF returnDimension = 1 THEN

Return type is returnType

ELSE

Return type is PtgNumMultiDim with unit = returnType and dimension = returnDimension

END IF

#### Not

The **Not** function performs the [**Boolean**](#gt_1d79d7a7-ba2c-4b34-931c-7ba8057c87b2) NOT operation.

**ABNF:**

1. Not = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " NOT():128"

**Required Arguments:**

*Name:* **Value**

*Type:* [vBoolean](#Section_8a0254e2d0bc4714be02f7104ff17c1f)

An argument that specifies the operand.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool containing the value of the Boolean NOT operation on **Value**.

#### Now

The **Now** function returns the current date and time.

**ABNF:**

1. Now = "NOW():128"

**Return Value:**

*Type:* [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6)

This function returns a PtgDate containing the value of current date and time.

#### Or

The **Or** function converts an argument to a [**Boolean**](#gt_1d79d7a7-ba2c-4b34-931c-7ba8057c87b2) value according to the conversion specified by [vBoolean](#Section_8a0254e2d0bc4714be02f7104ff17c1f).

**ABNF:**

1. Or = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " OR():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool containing the value of **Arg1**.

#### Pct

The **Pct** function performs a percent conversion on the operand.

**ABNF:**

1. Pct = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " \_PCT():128"

**Required Arguments:**

*Name:* **Value**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand that will be converted to a percent.

**Return Value:**

*Type:* [PtgNumPct](#Section_ef65a2675fd84ba3a1448fb7698f6551)

This function returns a PtgNumPct containing **Value** / 100.

#### Pi

The **Pi** function returns the mathematical constant pi.

**ABNF:**

1. Pi = "PI():128"

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns PtgNum a containing the value of the mathematical constant pi.

#### Pnt

The **Pnt** function MUST be ignored.

**ABNF:**

1. Pnt = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " PNT():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

This argument is unused and MUST be ignored.

*Name:* **Arg2**

*Type:* vDouble

This argument is unused and MUST be ignored.

**Return Value:**

*Type:* [PtgNoOp](#Section_5d39bf2968f3460e8ae0a46d6262dd4d)

This function returns a PtgNoOp.

#### Pow

The **Pow** function performs an exponentiation calculation.

**ABNF:**

1. Pow = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " POW():128"

**Required Arguments:**

*Name:* **Base**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the number to be raised to the power of **Exponent**.

*Name:* **Exponent**

*Type:* vDouble

An argument that specifies the exponent by which to raise **Base***.*

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum or PtgErr containing **Base** raised to the power of **Exponent**. The type of the return value is calculated by the following algorithm.

SET returnType = [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

SET returnError = #NUM!

SET returnDimension = 0

IF Base.Value = 0 THEN

IF Exponent.Value = 0 THEN

SET returnType = PtgErr

SET returnError = #NUM!

ELSE IF Exponent.Value < 0 THEN

SET returnType = PtgErr

SET returnError = #DIV/0

END IF

ELSE IF Base.Value < 0 AND Exponent is not an integer THEN

SET returnType = PtgErr

SET returnError = #DIM!

ELSE IF Base.Unit = [PtgNumPct](#Section_ef65a2675fd84ba3a1448fb7698f6551) THEN

SET returnType = PtgNumPct

SET returnDimension = 1

ELSE

SET returnType = Base.Unit

SET returnDimension = Base.Dimension \* Exponent.Value

END IF

IF returnType = PtgErr THEN

Return type is PtgErr with error code of returnError

ELSE IF returnDimension = 0 THEN

Return type is PtgNum

ELSE IF returnDimension = 1 THEN

Return type is returnType

ELSE

Return type is [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84) with unit = returnType and dimension = returnDimension

END IF

#### Rad

The **Rad** function converts a value to an angle in radians.

**ABNF:**

1. Rad = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " RAD():128"

**Required Arguments:**

*Name:* **Angle**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies an angle.

**Return Value:**

*Type:* [PtgAngRad](#Section_a09cdad2084647eb97edef0ef25726e0)

This function returns a PtgAngRad. If **Angle** is a [vAngle](#Section_8609b9afc5aa4a0e9c37b1f7929ac9b4), the value is equal to the value of **Angle**. Otherwise, **Angle** is interpreted as an angle in degrees and the return value is equal to:

pi / 180.0 \* value of **Angle**

#### Rand

The **Rand** function generates a random number.

**ABNF:**

1. Rand = "RAND():128"

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing a random double precision [**floating-point number**](#gt_237156df-b9cf-4b8a-9753-98001801a90d) greater than or equal to 0.0 and less than 1.0.

#### Replace

The **Replace** function replaces part of a text string with another text string.

**ABNF:**

1. Replace = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val val " REPLACE():128"

**Required Arguments:**

*Name:* **SourceText**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the string to perform replacement on.

*Name:* **StartPos**

*Type:* [vSignedLong](#Section_52e79e59b9eb4e8a8a7e92d0cdbdecf8)

An argument that specifies the one-based starting position in **SourceText** where replacement begins.

*Name:* **NumChars**

*Type:* vSignedLong

An argument that specifies the number of characters in **SourceText** to be replaced by **ReplaceText**.

*Name:* **ReplaceText**

*Type:* vString

An argument that specifies the string to use for replacement.

**Return Value:**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a PtgStr1 containing **SourceText** modified with the replaced text. If **StartPos** is less than or equal to zero or greater than the number of characters in **SourceText**, the function returns a PtgStr1 containing **SourceText** with **ReplaceText** appended at the end. If **StartPos** plus **NumChars** minus one exceeds the length of **SourceText**,orif **NumChars** is less than zero, the function returns a PtgStr1 containing **SourceText** truncated starting from **StartPos** and with **ReplaceText** appended at the end.

#### RGB

The **RGB** function calculates an [**RGB**](#gt_2c716d3a-e60b-4e52-bbb0-2fdeb298003b) color value as a combination of red, green and blue components.

**ABNF:**

1. RGB = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val " RGB():128"

**Required Arguments:**

*Name:* **Red**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

An argument that specifies the intensity of red.

*Name:* **Green**

*Type:* vUnsignedInt

An argument that specifies the intensity of green.

*Name:* **Blue**

*Type:* vUnsignedInt

An argument that specifies the intensity of blue.

**Return Value:**

*Type:* [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882)

This function returns a PtgColorRGB containing a color value. The byte that specifies red in the PtgColorRGB is equal to **Red**. The byte that specifies green in the PtgColorRGB is equal to **Green**. The byte that specifies blue in the PtgColorRGB is equal to **Blue**. If the value of any of the arguments is greater than 0xFF, the function sets the argument’s value to the result of the bitwise AND operation between the original value and 0xFF, before performing the operation.

#### Right

The **Right** function returns the last character or characters in a string.

**ABNF:**

1. Right = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";RIGHT():129"

**Required Arguments:**

*Name:* **Text**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the text string.

**Optional Arguments:**

Name: **NumChars**

Type: [vSignedLong](#Section_52e79e59b9eb4e8a8a7e92d0cdbdecf8)

An argument that specifies the number of characters to be returned. The default value is one.

**Return Value:**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69)

This function returns a PtgStr1 containing the last **NumChars** characters of **Text**. If **NumChars** is less than zero or greater than the number of characters in **Text**, the function returns a PtgStr1 containing **Text**. If **NumChars** is equal to zero, the function returns a PtgStr1 containing an empty string.

#### Round

The **Round** function performs a rounding calculation.

**ABNF:**

Round = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " ROUND():128"

**Required Arguments:**

*Name:* **Number**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the number to round.

*Name:* **Digits**

*Type:* [vSignedInt](#Section_6e6498757ff54acfa20b54eb9d8cd472)

An argument that specifies the decimal place to use for the rounding operation.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad)

This function returns a vNum containing the rounded value of **Number**, as described by the following table.

| Condition | Return value |
| --- | --- |
| **Digits** > 0 | The function returns **Number** rounded to **Digits** places to the right of the decimal point. |
| **Digits** = 0 | The function returns **Number** rounded to an integer. |
| **Digits** < 0 | The function returns **Number** rounded to negative **Digits** places to the left of the decimal point. |

The unit of the return value is equal to the unit of **Number**. If **Digits** is less than -308 or greater than 15, the function returns **Number**.

#### Sat

The **Sat** function calculates the saturation component, as specified in the [**HSL**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5), of a color value.

**ABNF:**

1. Sat = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " SAT():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the color.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b)

This function returns a PtgUnsWord containing the saturation of **Arg1**. The value is less than or equal to 240.

#### SatDiff

The **SatDiff** function calculates the difference in saturation, as specified in the [**HSL**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5), between two color values.

**ABNF:**

1. SatDiff = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " SATDIFF():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the first color.

*Name:* **Arg2**

*Type:* vColor

An argument that specifies the second color.

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing a value that is equal to the saturation of **Arg1** minus the saturation of **Arg2**. This value is greater than or equal to -240 and less than or equal to 240.

#### Second

The **Second** function returns the second component from a value representing a date and time.

**ABNF:**

1. Second = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";SECOND():129"

**Required Arguments:**

*Name:* **DateTimeArg**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies a date and time value.

**Optional Arguments:**

*Name:* **Locale**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) to use when parsing **DateTimeArg**.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function attempts a conversion, as described by the [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) function, of **DateTimeArg** to a [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6). If the conversion succeeds, the function returns a PtgUnsWord containing the second component from **DateTimeArg.** The value is greater than or equal to zero and less than or equal to 59. If the conversion fails, the function returns a PtgErr with an error code of #VALUE!.

#### SetAtRef

The **SetAtRef** function conditionally returns an argument.

**ABNF:**

1. SetAtRef = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) [val [val]] sp ("1" / "2" / "3") ";SETATREF():129"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies the operand to return.

**Optional Arguments:**

*Name*: **Unused**

*Type*: vAny

This argument is unused and MUST be ignored.

*Name*: **IgnoreArg**

*Type*: vAny

An argument that specifies whether **Arg1** is returned. The default value is FALSE.

**Return Value:**

*Type:* vAny

This function returns a vAny containing **Arg1** if **IgnoreArg** is FALSE. If **IgnoreArg** is TRUE, the function returns a [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b) containing a value of 0.

#### SetAtRefEval

The **SetAtRefEval** function returns an argument.

**ABNF:**

1. SetAtRefEval = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " SETATREFEVAL():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies the operand to return.

**Return Value:**

*Type:* vAny

This function returns a vAny containing **Arg1**.

#### SetAtRefExpr

The **SetAtRefExpr** function returns an argument.

**ABNF:**

1. SetAtRefExpr = [[val](#section_9dd6fede2e2a4b4f9193df3626a781bd)] sp ("0" / "1") ";SETATREFEXPR():129"

**Optional Arguments:**

*Name:* **Arg1**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies the operand to return. The default value is a [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b) with a value of 0.

**Return Value:**

*Type:* vAny

This function returns a vAny containing **Arg1**.

#### Shade

The **Shade** function decreases the luminance, as specified in the [**HSL**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5), of a color value.

**ABNF:**

1. Shade = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " SHADE():128"

**Required Arguments:**

*Name:* **Color**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the color.

*Name:* **Delta**

*Type:* [vSignedInt](#Section_6e6498757ff54acfa20b54eb9d8cd472)

An argument that specifies the amount to modify the luminance of **Color**.

**Return Value:**

*Type:* [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882)

This function returns a PtgColorRGB containing a color value with decreased luminance. If **Delta** is less than zero, the luminance is increased.

#### ShapeText

The **ShapeText** function is unsupported.

**ABNF:**

1. ShapeText = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) [val] " SHAPETEXT():129"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd)

This argument is unused and MUST be ignored.

**Optional Arguments:**

*Name:* **Arg2**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

This argument is unused and MUST be ignored.

**Return Value:**

*Type:* undefined

This function is not evaluated and does not return a valid result.

#### Sign

The **Sign** function returns the sign of a given number.

**ABNF:**

1. Sign = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";SIGN():129"

**Required Arguments:**

*Name:* **Number**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Optional Arguments:**

*Name:* **FuzzValue**

*Type:* vDouble

An argument that specifies the tolerance. If **FuzzValue** is negative, the absolute value of **FuzzValue** is used. The default value is 1E-9 (0.000000001).

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing the sign of **Number**. If **Number** is greater than the absolute value of **FuzzValue**, the function returns 1. If **Number** is less than the negative of the absolute value of **FuzzValue**, the function returns -1. Otherwise, the function returns 0.

#### Sin

The **Sin** function performs the sine calculation.

**ABNF:**

1. Sin = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " SIN():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing the sine of **Arg1**, wherethe value of **Arg1** is assumed to have the unit of radians.

#### SinH

The **SinH** function performs the hyperbolic sine calculation.

**ABNF:**

1. SinH = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " SINH():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing the hyperbolic sine of **Arg1**, wherethe value of **Arg1** is assumed to have the unit of radians.

#### Sqrt

The **Sqrt** function performs a square root calculation.

**ABNF:**

1. Sqrt = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " SQRT():128"

**Required Arguments:**

*Name:* **Number**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum containing the square root of **Number**. The unit of the return value is equal to the unit of **Number**. If the dimension of **Number** is greater than 2, the function returns a [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84) with value equal to the square root of **Number**, unit equal to the unit of **Number**, and dimension equal to the dimension of **Number** divided by 2. If **Number** is less than 0, the function returns a PtgErr with an error code of #NUM!. If the dimension of **Number** is not divisible by two, the function returns a PtgErr with an error code of #DIM!.

#### StrSame

The **Strsame** function determines whether two text strings are the same.

**ABNF:**

1. StrSame = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val [val] sp ("2" / "3") ";STRSAME():129"

**Required Arguments:**

*Name:* **FirstText**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the first string to be compared.

*Name:* **SecondText**

*Type:* vString

An argument that specifies the second string to be compared.

**Optional Arguments:**

*Name*: **IgnoreCase**

*Type*: [vBoolean](#Section_8a0254e2d0bc4714be02f7104ff17c1f)

An argument that specifies whether the comparison is case insensitive. A value of TRUE indicates that case is ignored. The default value is FALSE.

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool containing TRUE if the strings are the same and FALSE otherwise. The comparison is performed according to .NET globalization [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) based on the [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) with a value of 1033, which maps to US English. For more information about .NET globalization rules, see [[MSDN-ENCLOC]](https://go.microsoft.com/fwlink/?LinkId=153669).

#### StrSameEx

The **StrSameEx** function determines whether two text strings are the same.

**ABNF:**

1. StrSameEx = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val val " STRSAMEEX():128"

**Required Arguments:**

*Name:* **FirstText**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the first string to be compared.

*Name:* **SecondText**

*Type:* vString

An argument that specifies the second string to be compared.

*Name:* **Locale**

*Type:*  [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies the [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) of the culture used by the string comparison, according to .Net globalization [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f). A value of zero specifies the invariant culture. For more information about .NET globalization rules, see [[MSDN-ENCLOC]](https://go.microsoft.com/fwlink/?LinkId=153669).

*Name:* **Flag**

*Type:* [vSignedLong](#Section_52e79e59b9eb4e8a8a7e92d0cdbdecf8)

An argument that specifies additional rules to be used in the comparison. The value is a combination of values from the following table. The rules given in the following table correspond to members of the .NET CompareOptions enumeration, as described in [[MSDN-CompareOptions]](https://go.microsoft.com/fwlink/?LinkId=180505).

| Value | Meaning |
| --- | --- |
| 0x00 | No additional rules apply. |
| 0x01 | Ignore case. |
| 0x10 | Ignore non-spacing combining characters. |
| 0x100 | Ignore symbols. |
| 0x1000 | All non-alphanumeric characters come before all alphanumeric characters. |
| 0x10000 | Ignore differences between hiragana and katakana characters that represent the same phonetic sound. |
| 0x20000 | Ignore character width, or differences between the single-byte and double-byte representations of the same character. |

**Return Value:**

*Type:* [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508)

This function returns a PtgBool containing TRUE if the strings are the same and FALSE otherwise. If the **Locale** or **Flag** arguments are invalid, this function returns a PtgBool containing a value of FALSE.

#### Sub

The **Sub** function performs a subtraction calculation.

**ABNF:**

1. Sub = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " \_SUB():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDoubleEx](#Section_17ab59629f9e4b75a84af0cc80400b88)

An argument that specifies the first operand of the calculation.

*Name:* **Arg2**

*Type:* vDoubleEx

An argument that specifies the second operand of the calculation.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a vNum or a PtgCy containing **Arg1** minus **Arg2**. The type of the return value is calculated by the following algorithm.

SET returnType = [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

SET returnDimension = 0

SET returnCurrencyID = 0

IF Arg1.Type = PtgCy AND Arg2.Type != PtgCy THEN

SET returnType = PtgCy

SET returnCurrencyID = currencyID of Arg1

ELSE IF Arg1.Type != PtgCy AND Arg2.Type = PtgCy THEN

SET returnType = PtgCy

SET returnCurrencyID = currencyID of Arg2

ELSE IF Arg1.Type = PtgCy AND Arg2.Type = PtgCy THEN

SET returnType = PtgCy

IF currencyID of Arg1 = currencyID of Arg2 OR currencyID of Arg2 = 1 THEN

SET returnCurrencyID = currencyID of Arg1

ELSE IF currencyID of Arg1 = 1 THEN

SET returnCurrencyID = currencyID of Arg2

ELSE

SET returnType = PtgErr

END IF

ELSE IF Arg1.Type = [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6) AND Arg2.Type = PtgDate THEN

SET returnType = [PtgTDurDft](#Section_1200eb57709c4fcc9d20e2bac20fa493)

SET returnDimension = 1

ELSE IF Arg1.Unit = [PtgNumDft](#Section_de132a3cdab04751844609cf15ec3756) AND Arg2.Unit is a [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b) THEN

SET returnType = Arg2.Unit

SET returnDimension = 1

ELSE IF Arg1.Unit is a vUnitType THEN

SET returnType = Arg1.Unit

IF Arg2.Dimension = 0 OR Arg2.Dimension = Arg1.Dimension THEN

SET returnDimension = Arg1.Dimension

ELSE

SET returnDimension = 1

END IF

IF (returnType = [PtgAcre](#Section_7f6eff5e434a407b8565fe966e0b066c) OR returnType = [PtgHectare](#Section_7b405e52f9c34451a7f90ad394edd4aa)) AND returnDimension != 2 THEN

SET returnType = PtgNumDft

END IF

ELSE IF Arg2.Unit is a vUnitType THEN

SET returnType = Arg2.Unit

IF Arg1.Dimension = 0 OR Arg2.Dimension = Arg1.Dimension THEN

SET returnDimension = Arg2.Dimension

ELSE

SET returnDimension = 1

END IF

IF (returnType = PtgAcre OR returnType = PtgHectare) AND returnDimension != 2 THEN

SET returnType = PtgNumDft

END IF

END IF

IF returnType = PtgCy THEN

Return type is PtgCy with currencyID = returnCurrencyID

ELSE IF returnType = PtgErr THEN

Return type is PtgErr with error code of #VALUE!

ELSE IF returnDimension = 0 THEN

Return type is PtgNum

ELSE IF returnDimension = 1 THEN

Return type is returnType

ELSE

Return type is [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84) with unit = returnType and dimension = returnDimension

END IF

#### Substitute

The **Substitute** function performs text substitution by finding a substring in a text string and replacing it with another text string.

**ABNF:**

1. Substitute = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val [val [val]] sp ("3" / "4" /"5") ";SUBSTITUTE():129"

**Required Arguments:**

*Name:* **SourceText**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the string to perform substitution on.

*Name:* **FindText**

*Type:* vString

An argument that specifies the substring in **SourceText** to be substituted.

*Name:* **SubstituteText**

*Type:* vString

An argument that specifies the string to use for substitution.

**Optional Arguments:**

*Name*: **WhichInstance**

*Type*: [vSignedLong](#Section_52e79e59b9eb4e8a8a7e92d0cdbdecf8)

An argument that specifies which instance, among multiple instances of **FindText** in **SourceText,** is to be substituted. The default value is 0.

*Name*: **IgnoreCase**

*Type*: [vBoolean](#Section_8a0254e2d0bc4714be02f7104ff17c1f)

An argument that specifies whether the search for **FindText** in **SourceText** is case insensitive. A value of TRUE specifies that case is ignored. The default value is FALSE.

**Return Value:**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function returns a PtgStr1 containing **SourceText** modified with the substituted text. If **WhichInstance** is zero, the function returns a PtgErr with an error code of #VALUE!. If **WhichInstance** is less than zero or greater than the number of **FindText** instances in **SourceText**, or if **FindText** is not found in **SourceText**, the function returns a PtgStr1 containing **SourceText** without any substitution. The search is performed according to .NET globalization [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) based on the [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) in the **DocumentLanguage** field of the Properties **GlobalElement** in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) of the document. For more information about .NET globalization rules, see [[MSDN-ENCLOC]](https://go.microsoft.com/fwlink/?LinkId=153669).

#### Sum

The **Sum** function performs the sum calculation.

**ABNF:**

1. Sum = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) \*val sp unsigned-integer-value ";SUM():129"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the first operand of the calculation.

**Optional Arguments:**

Zero or more subsequent operands of type vDouble.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad)

This function returns a vNum containing the sum of all operands. If any of the arguments is a [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b), the unit of the return value is equal to the unit of the first argument that is a vUnitType. Otherwise, the function returns a [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0).

#### Tan

The **Tan** function performs the tangent calculation.

**ABNF:**

1. Tan = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " TAN():128"

**Required Arguments:**

*Name:* **Angle**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing the tangent of the value of **Arg1**, wherethe value of **Arg1** is assumed to have the unit of radians.

#### TanH

The **TanH** function performs the hyperbolic tangent calculation.

**ABNF:**

1. TanH = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " TANH():128"

**Required Arguments:**

*Name:* **Angle**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing the hyperbolic tangent of the value of **Arg1**, wherethe value of **Arg1** is assumed to have the unit of radians.

#### TextHeight

The **TextHeight** function calculates the height of the text specified in a [CT\_ShapeText](#Section_80e46fba919845399e9b8c1147fccc25) element in the [ShapeTextBinding XML Part](#Section_2e21d026e11d46ee860a75c6b75bd84a).

**ABNF:**

1. TextHeight = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val val " TEXTHEIGHT():128"

**Required Arguments:**

*Name:* **Unused**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

This argument is unused and MUST be ignored.

*Name:* **Unused**

*Type:* vAny

This argument is unused and MUST be ignored.

*Name:* **Sheet**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the **Name** of a CT\_ShapeText element in the ShapeTextBinding XML Part.

*Name:* **OriginalHeight**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the height of the text in **Sheet** prior to any [diagram updates](#Section_b30a1cbd988f4d6bbf2e7219c9423972).

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing the height of the [text element](#Section_b061c2bde75d48edb51047479b51c6ec).

#### TextWidth

The **TextWidth** function calculates the width of the text specified in a [CT\_ShapeText](#Section_80e46fba919845399e9b8c1147fccc25) element in the [ShapeTextBinding XML Part](#Section_2e21d026e11d46ee860a75c6b75bd84a).

**ABNF:**

1. TextWidth = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val val val " TEXTWIDTH():128"

**Required Arguments:**

*Name:* **Unused**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

This argument is unused and MUST be ignored.

*Name:* **Unused**

*Type:* vAny

This argument is unused and MUST be ignored.

*Name:* **Sheet**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the **Name** of a CT\_ShapeText element in the ShapeTextBinding XML Part.

*Name:* **OriginalWidth**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the width of the text in **Sheet** prior to any [diagram updates](#Section_b30a1cbd988f4d6bbf2e7219c9423972).

**Return Value:**

*Type:* [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a PtgNum containing the width of the [text element](#Section_b061c2bde75d48edb51047479b51c6ec).

#### Time

The **Time** function returns a time from values representing an hour, minute, and second.

**ABNF:**

1. Time = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) val val " TIME():128"

**Required Arguments:**

*Name:* **Hour**

*Type:* [vUnsignedInt](#Section_44ec4229ed484f498abffdb0490d6f1a)

An argument that specifies an offset in hours from midnight.

*Name:* **Minute**

*Type:* vUnsignedInt

An argument that specifies an offset in minutes from **Hour**.

*Name:* **Second**

*Type:* vUnsignedInt

An argument that specifies an offset in seconds from **Minute**.

**Return Value:**

*Type:* [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6)

This function returns a PtgDate containing a time of day. The date-time-value component of the return value is the signed fractional value of the double precision number representing the input date-time, as specified in [[MS-OAUT]](%5bMS-OAUT%5d.pdf#Section_bbb05720f72445c78d17f83c3d1a3961) section [2.2.25](http://msdn.microsoft.com/en-us/library/35c9bf2d-b8e8-4d7d-a50f-367da0d99fce/).

#### TimeValue

The **TimeValue** function returns the time component from a value representing a date and time.

**ABNF:**

1. TimeValue = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";TIMEVALUE():129"

**Required Arguments:**

*Name:* **DateTimeArg**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies a value representing a date and time.

**Optional Arguments:**

*Name:* **Locale**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) to use when parsing **DateTimeArg**.

**Return Value:**

*Type:* [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function attempts a conversion, as described by the [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) function, of **DateTimeArg** to a PtgDate. If the conversion succeeds, the function returns a PtgDate containing the time component of **DateTimeArg**. If the conversion fails, the function returns a PtgErr with an error code of #VALUE!.

#### Tint

The **Tint** function increases the luminance, as specified in the [**HSL**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5), of a color value.

**ABNF:**

1. Tint = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " TINT():128"

**Required Arguments:**

*Name:* **Color**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the color.

*Name:* **Delta**

*Type:* [vSignedInt](#Section_6e6498757ff54acfa20b54eb9d8cd472)

An argument that specifies the amount to modify the luminance of **Color**.

**Return Value:**

*Type:* [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882)

This function returns a PtgColorRGB containing a color value with increased luminance. If **Delta** is less than zero, the luminance is decreased.

#### Tone

The **Tone** function decreases the saturation, as specified in the [**HSL**](#gt_e801f5cb-4827-4f1b-9337-d13e22a43972) [**color space**](#gt_5d3fb5ea-c686-4d3b-b0ff-aef33fe1aee5), of a color value.

**ABNF:**

1. Tone = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " TONE():128"

**Required Arguments:**

*Name:* **Color**

*Type:* [vColor](#Section_9b24cbcbe0544d9ea92dccd6dde23908)

An argument that specifies the color.

*Name:* **Delta**

*Type:* [vSignedInt](#Section_6e6498757ff54acfa20b54eb9d8cd472)

An argument that specifies the amount to modify the saturation of **Color**.

**Return Value:**

*Type:* [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882)

This function returns a PtgColorRGB containing a color value with decreased saturation. If **Delta** is less than zero, the saturation is increased.

#### Trim

The **Trim** function performs the removal of all [**whitespace**](#gt_d812ae04-029d-4f94-b205-962d2142a117) from a string except for single spaces between words.

**ABNF:**

1. Trim = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " TRIM():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the string from which to remove whitespace.

**Return Value:**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69)

This function returns a PtgStr1 containing the string with whitespace removed. The following [**Unicode**](#gt_c305d0ab-8b94-461a-bd76-13b40cb8c4d8) characters are considered as whitespace.

| Value | Meaning |
| --- | --- |
| "\u0009" | Character Tabulation |
| "\u000a" | Line Feed |
| "\u000b" | Line Tabulation |
| "\u000c" | Form Feed |
| "\u000d" | Carriage Return |
| "\u0020" | Space |
| "\u00a0" | No-Break Space |
| "\u1680" | Ogham Space Mark |
| "\u180e" | Mongolian Vowel Separator |
| "\u2000" | En Quad |
| "\u2001" | Em Quad |
| "\u2002" | En Space |
| "\u2003" | Em Space |
| "\u2004" | Three-Per-Em Space |
| "\u2005" | Four-Per-Em Space |
| "\u2006" | Six-Per-Em Space |
| "\u2007" | Figure Space |
| "\u2008" | Punctuation Space |
| "\u2009" | Thin Space |
| "\u200a" | Hair Space |
| "\u202f" | Narrow No-Break Space |
| "\u205f" | Medium Mathematical Space |
| "\u3000" | Ideographic Space |

#### Trunc

The **Trunc** function performs a truncation operation.

**ABNF:**

1. Trunc = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) val " TRUNC():128"

**Required Arguments:**

*Name:* **Number**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the number to truncate.

*Name:* **Digits**

*Type:* [vSignedInt](#Section_6e6498757ff54acfa20b54eb9d8cd472)

An argument that specifies the decimal place to use for the truncation operation.

**Return Value:**

*Type:* [[vNum](#Section_ad426b3dbea2407c8687202b9d13adad)](#Section_d895c0dbaac64ac3b9390507f58282d0)

This function returns a vNum containing the truncated value of **Number**, as described by the following table.

| Condition | Result |
| --- | --- |
| **Digits** > 0 | The function returns **Number** truncated to **Digits** places to the right of the decimal point. |
| **Digits** = 0 | The function returns **Number** truncated to an integer. |
| **Digits** < 0 | The function returns **Number** truncated to negative **Digits** places to the left of the decimal point. |

The unit of the return value is equal to the unit of **Number**. If **Digits** is less than -308 or greater than 15, the function returns **Number**.

#### UMinus

The **UMinus** function performs the negation calculation.

**ABNF:**

1. UMinus = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " \_UMINUS():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [[vNum](#Section_ad426b3dbea2407c8687202b9d13adad)](#Section_1d4fd8bc32844cb5915e14424a11a789)

This function returns a vNum containing the negation of the value of **Arg1**. The unit of the return value is equal to the unit of **Arg1**.

#### UPlus

The **UPlus** function performs the identity calculation.

**ABNF:**

1. UPlus = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " \_UPLUS():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d)

An argument that specifies the operand of the calculation.

**Return Value:**

*Type:* [vNum](#Section_ad426b3dbea2407c8687202b9d13adad)

This function returns a vNum containing the value of **Arg1**. The unit of the return value is equal to the unit of **Arg1**.

#### Upper

The **Upper** function performs upper case conversion.

**ABNF:**

1. Upper = [val](#section_9dd6fede2e2a4b4f9193df3626a781bd) " UPPER():128"

**Required Arguments:**

*Name:* **Arg1**

*Type:* [vString](#Section_fe954fee44b243f1b639ae8ad580be3b)

An argument that specifies the string to convert.

**Return Value:**

*Type:* [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69)

This function returns a PtgStr1 containing the string converted to upper case. The conversion is performed according to .NET globalization [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) based on the [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) in the **DocumentLanguage** field of the Properties **GlobalElement** in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) of the document. For more information about .NET globalization rules, see [[MSDN-ENCLOC]](https://go.microsoft.com/fwlink/?LinkId=153669).

#### WeekDay

The **WeekDay** function returns the day of the week, according to the Gregorian calendar, from a value representing a date and time.

**ABNF:**

1. WeekDay = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";WEEKDAY():129"

**Required Arguments:**

*Name:* **DateTimeArg**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies a date and time value.

**Optional Arguments:**

*Name:* **Locale**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) to use when parsing **DateTimeArg**.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function attempts a conversion, as described by the [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) function, of **DateTimeArg** to a [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6). If the conversion succeeds, the function returns a PtgUnsWord containing the weekday component from **DateTimeArg.** The value is greater than or equal to 1 and less than or equal to 7, where a value of 1 corresponds to Monday. If the conversion fails, the function returns a PtgErr with an error code of #VALUE!.

#### Year

The **Year** function returns the year, according to the Gregorian calendar, from a value representing a date and time.

**ABNF:**

1. Year = [val](#Section_9dd6fede2e2a4b4f9193df3626a781bd) [val] sp ("1" / "2") ";YEAR():129"

**Required Arguments:**

*Name:* **DateTimeArg**

*Type:* [vAny](#Section_3a9fd4fb2af1480b9cd1c40c8640d1c0)

An argument that specifies a date and time value.

**Optional Arguments:**

*Name:* **Locale**

*Type:* [vLanguageID](#Section_eee4cf5d442246928202e581f8be4d24)

An argument that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) to use when parsing **DateTimeArg**.

**Return Value:**

*Type:* [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

This function attempts a conversion, as described by the [DateTime](#Section_7e46243bfaaa4fff8e620dea9bf123d1) function, of **DateTimeArg** to a [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6). If the conversion succeeds, the function returns a PtgUnsWord containing the year component from **DateTimeArg**. The value is greater than or equal to 1 and less than or equal to 9999. If the conversion fails, the function returns a PtgErr with an error code of #VALUE!.

### Parse Token Table

This section specifies the [parse tokens](#Section_5ae5d80a9aaf40948a2d92cf8e499589) (Ptgs) that can be part of a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc). The numerical token type ID and token type name are provided for each token.

| ID | Ptg |
| --- | --- |
| 36 | [PtgAcre](#Section_7f6eff5e434a407b8565fe966e0b066c) |
| 81 | [PtgAngDD](#Section_222188be55a34b90b2c39eb9f5878bab) |
| 80 | [PtgAngDft](#Section_57e444144344437fb32a6cf2d3b20d2f) |
| 82 | [PtgAngDMS](#Section_459da555a74b48439f4cbc516a043d30) |
| 83 | [PtgAngRad](#Section_a09cdad2084647eb97edef0ef25726e0) |
| 97 | [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508) |
| 106 | [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882) |
| 111 | [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3) |
| 1 | [PtgDataBinding](#Section_7442cb26af7f46ce9bd348656835396c) |
| 40 | [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6) |
| 44 | [PtgEDay](#Section_f6690609c80a49ed9fffe0dbcfb262a6) |
| 45 | [PtgEHour](#Section_e31838fa14684d478f193c033df45e4c) |
| 46 | [PtgEMin](#Section_f58cc7ec8a1246d89edb4b3e00a08ac6) |
| 224 | [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2) |
| 47 | [PtgESec](#Section_01c1da7178fc485c8ee6b48674eeb35b) |
| 43 | [PtgEWeek](#Section_4aae0088510448abaf3f8ec63f95fd5f) |
| 128 | [PtgFunc](#Section_8206928cdaaf400daa071a566fe1c88b) |
| 129 | [PtgFuncVar](#Section_e483d78ecd494ba7aa042f843a09f8df) |
| 37 | [PtgHectare](#Section_7b405e52f9c34451a7f90ad394edd4aa) |
| 163 | [PtgJmp](#Section_a2fcbad2d4cd4d0bb4338ec960e51755) |
| 160 | [PtgJmpF](#Section_81250f6f47d04c3d93cd60e2d8798784) |
| 164 | [PtgJmpLabel](#Section_5f545b4b47f44ed38932659537323ff9) |
| 161 | [PtgJmpT](#Section_f605c7795ea249bca765efaf9ae6e1f0) |
| 229 | [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd) |
| 240 | [PtgNoOp](#Section_5d39bf2968f3460e8ae0a46d6262dd4d) |
| 32 | [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0) |
| 69 | [PtgNumCM](#Section_bc2d88fb83a94d9588803acf4c0cdc27) |
| 64 | [PtgNumDft](#Section_de132a3cdab04751844609cf15ec3756) |
| 66 | [PtgNumF](#Section_09bdcb1c922444c8b9c3ecdaa03e0c6d) |
| 67 | [PtgNumFI](#Section_a3917f0c42e94c71b55c70ea3aed6819) |
| 65 | [PtgNumI](#Section_ac87036f348249b58cba49659d7598a8) |
| 72 | [PtgNumKM](#Section_6dc5f00e5e7945678ac890f98cd40706) |
| 71 | [PtgNumM](#Section_000148cf28984a299ade5cc03f395c6c) |
| 68 | [PtgNumMI](#Section_e1f97f6e268c4429868c74e9323a7b9e) |
| 70 | [PtgNumMM](#Section_5fc97e8093e041d7bcd6ddd2c66aa589) |
| 41 | [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84) |
| 76 | [PtgNumNM](#Section_bf29f1f677664e9e856811466228244f) |
| 33 | [PtgNumPct](#Section_ef65a2675fd84ba3a1448fb7698f6551) |
| 75 | [PtgNumYards](#Section_7039749a686144d9bcd9da3946769b16) |
| 63 | [PtgPageDft](#Section_fe0d479f9b534257a3d13618b7314aaa) |
| 225 | [PtgPnt](#Section_45fae81bb4e6413394e32daf5c98d790) |
| 147 | [PtgPop](#Section_a5016a0a9e794b6dbd38231239a97710) |
| 149 | [PtgPushTop](#Section_46e71a47bb0d441cb00756eb6182c94a) |
| 2 | [PtgRecalcRef](#Section_1877c57f45234887a93a70db1b9d46e5) |
| 96 | [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69) |
| 42 | [PtgTDurDft](#Section_1200eb57709c4fcc9d20e2bac20fa493) |
| 52 | [PtgTypCD](#Section_bf50076676d84a8d900468c81b728b8e) |
| 54 | [PtgTypCi](#Section_c5f1913417a7448287105b162359a1fa) |
| 48 | [PtgTypDft](#Section_d66b059e323d45739c91ec57ed148b16) |
| 53 | [PtgTypDi](#Section_a56c1d526fe9452b9857c97fd84c3825) |
| 51 | [PtgTypPi](#Section_f3c4aabfb3bb4c68b5b9f175f9620048) |
| 49 | [PtgTypPP](#Section_997b3916e4734908b63936fd05317e60) |
| 50 | [PtgTypPt](#Section_b30a64caae964c76ac0728982f26bb31) |
| 98 | [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b) |

### Parse Token Definitions

This section specifies the tokens that can be part of a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc). The definition of each token specifies the **TokenValue** and **TokenType** as described in the [parse tokens](#Section_5ae5d80a9aaf40948a2d92cf8e499589) section. In the ABNF string specified for each token, the **TokenValue** is the left side of the [token-separator](#section_9dd6fede2e2a4b4f9193df3626a781bd) and the **TokenType** is the right side of the token-separator.

#### PtgAcre

The **PtgAcre** structure is a [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84) that specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of acres.

**ABNF:**

1. PtgAcre = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) ",36,2" token-separator "41"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgAngDD

The **PtgAngDD** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of decimal degrees.

**ABNF:**

PtgAngDD = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "81"

double-value is an [angleInternalUnitNumber](#Section_51d7e07cd65741efa808ebb426864c91).

#### PtgAngDft

The **PtgAngDft** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7). The unit is specified by the **DefaultUnitsAngle** attribute of [CT\_Page](#Section_020589d88a1f420c97805dd19dac1cf7).

**ABNF:**

1. PtgAngDft = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "80"

double-value is an [angleInternalUnitNumber](#Section_51d7e07cd65741efa808ebb426864c91).

#### PtgAngDMS

The **PtgAngDMS** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of degrees-minutes-seconds.

**ABNF:**

1. PtgAngDMS = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "82"

double-value is an [angleInternalUnitNumber](#Section_51d7e07cd65741efa808ebb426864c91).

#### PtgAngRad

The **PtgAngRad** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of radians.

**ABNF:**

1. PtgAngRad = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "83"

double-value is an [angleInternalUnitNumber](#Section_51d7e07cd65741efa808ebb426864c91).

#### PtgBool

The **PtgBool** structure specifies a [**Boolean**](#gt_1d79d7a7-ba2c-4b34-931c-7ba8057c87b2) value.

**ABNF:**

1. PtgBool = [bool-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "97"

bool-value specifies a Boolean. It MUST be a case insensitive value from the following table:

| Value | Meaning |
| --- | --- |
| false | Value is FALSE |
| true | Value is TRUE |

#### PtgColorRGB

The **PtgColorRGB** structure specifies an [**RGB**](#gt_2c716d3a-e60b-4e52-bbb0-2fdeb298003b) color value represented as a 4-byte signed integer.

**ABNF:**

1. PtgColorRGB = [color-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "106"

The most significant byte of color-value MUST be equal to 0xFF. The second most significant byte of color-value specifies the intensity of red. The third most significant byte of color-value specifies the intensity of green. The least significant byte of color-value specifies the intensity of blue.

#### PtgCy

The **PtgCy** structure specifies a currency value.

**ABNF:**

1. PtgCy = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) "," currency "," string-value token-separator "111"

double-value is a double precision [**floating-point number**](#gt_237156df-b9cf-4b8a-9753-98001801a90d).

currency is an unsigned integer that specifies the currency. It MUST be a value defined by [vCurrencyID](#Section_b13bb6692bc04291b31c4532158fa675).

string-value specifies the formatting information. It MUST be a numeric format string, as described in [[MSDN-FormattingTypes]](https://go.microsoft.com/fwlink/?LinkId=153833).

#### PtgDataBinding

The **PtgDataBinding** structure specifies a reference to a [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) item whose value is bound to external data, as described in [Data Binding](#Section_b9b194892da24d77a8a51c99bb866fe1).

**ABNF:**

1. PtgDataBinding = [shape-name](#section_9dd6fede2e2a4b4f9193df3626a781bd) "," shape-data "," shape-data-type "," dimension token-separator "1"

shape-name specifies the **Name** attribute of the [CT\_ShapeInfo](#Section_95489a96487e434db23af35fd0f877a0) element in the [ShapeInfo XML Part](#Section_bf230dbc228e4108bd8f93c9cbd4fcac) of this document that contains the shape data item.

shape-data specifies the **Name** attribute of a [CT\_ShapeData](#Section_a90cbb53af5347ae9da20afddd00cf2f) element in the ShapeInfo XML Part of this document. This element is the referenced shape data item and is a subordinate element of the element referenced by shape-name.

shape-data-type specifies the measurement unit of the value referenced by shape-data. It MUST be an entry under the ID column in the Parse Token Table in the [Parse Token Table](#Section_1c7c6787214545db854e2a1e1ca24d69) section.

dimension specifies the dimension of the value referenced by shape-data.

#### PtgDate

The **PtgDate** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of time.

**ABNF:**

1. PtgDate = [date-time-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "40"

date-time-value specifies a date and time, as specified in [[MS-OAUT]](%5bMS-OAUT%5d.pdf#Section_bbb05720f72445c78d17f83c3d1a3961) section [2.2.25](http://msdn.microsoft.com/en-us/library/35c9bf2d-b8e8-4d7d-a50f-367da0d99fce/).

#### PtgEDay

The **PtgEDay** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of days.

**ABNF:**

1. PtgEDay = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "44"

double-value is a [durationInternalUnitNumber](#Section_79862c6e7c6b4af5b33e935edf99db5b).

#### PtgEHour

The **PtgEHour** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of hours.

**ABNF:**

1. PtgEHour = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "45"

double-value is a [durationInternalUnitNumber](#Section_79862c6e7c6b4af5b33e935edf99db5b).

#### PtgEMin

The **PtgEMin** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of minutes.

**ABNF:**

1. PtgEMin = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "46"

double-value is a [durationInternalUnitNumber](#Section_79862c6e7c6b4af5b33e935edf99db5b).

#### PtgErr

Th**PtgErr** structure specifies an error code.

**ABNF:**

1. PtgErr = [string-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "224"

string-value specifies a string and MUST be a value from the following table.

| Value | Meaning |
| --- | --- |
| #DIM! | A dimensional value that exceeds the dimension range |
| #DIV/0 | Division by 0 |
| #VALUE! | An argument or operand of the wrong type |
| #REF! | A reference to a cell that does not exist |
| #NUM! | An invalid number |
| #N/A | Not available value |

#### PtgESec

The **PtgESec** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of seconds.

**ABNF:**

1. PtgESec = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "47"

double-value is a [durationInternalUnitNumber](#Section_79862c6e7c6b4af5b33e935edf99db5b).

#### PtgEWeek

The **PtgEWeek** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of weeks.

**ABNF:**

1. PtgEWeek = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "43"

double-value is a [durationInternalUnitNumber](#Section_79862c6e7c6b4af5b33e935edf99db5b).

#### PtgFunc

The **PtgFunc** structure specifies a function with a fixed number of arguments.

**ABNF:**

1. PtgFunc = [fixed-func-name](#section_9dd6fede2e2a4b4f9193df3626a781bd) "()" token-separator "128"

fixed-func-namespecifies a function name which MUST be an entry under the ABNF func-name column in the Function Token Table as specified in the [Function Token Table](#Section_a5e026e746284da18c6f1e6823e72c02) section. The corresponding entry under the Type column for that function MUST be PtgFunc.

#### PtgFuncVar

The **PtgFuncVar** structure specifies a function with a variable number of arguments.

**ABNF:**

1. PtgFuncVar = [num-args](#section_9dd6fede2e2a4b4f9193df3626a781bd) ";" var-func-name "()" token-separator "129"

num-args specifies the number of arguments.

var-func-namespecifies a function name which MUST be an entry under the ABNF func-name column in the Function Token Table as specified in the [Function Token Table](#Section_a5e026e746284da18c6f1e6823e72c02) section. The corresponding entry under the Type column for that function MUST be PtgFuncVar.

#### PtgHectare

The **Ptg** structure is a [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84) that specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of hectares.

**ABNF:**

1. PtgHectare = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) ",37,2" token-separator "41"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgJmp

The **PtgJmp** structure is a [control token](#Section_41b89226e2a7471fb70c6d01003be607) that specifies the position of the next token to be evaluated in the [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc).

**ABNF:**

1. PtgJmp = [string-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "163"

string-value is the label that specifies the position.

#### PtgJmpF

The **PtgJmpF** structure is a [control token](#Section_41b89226e2a7471fb70c6d01003be607) that specifies the position of the next token to be evaluated in the [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc) if the token that is currently at the top of the [evaluation stack](#Section_f70486783f2d445fae706da7e34fe640) evaluates to a value of FALSE. If the token that is currently at the top of the evaluation stack evaluates to a value of TRUE, it is popped from the evaluation stack.

**ABNF:**

1. PtgJmpF = [string-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "160"

string-value is the label that specifies the position.

#### PtgJmpLabel

The **PtgJmpLabel** structure is a [control token](#Section_41b89226e2a7471fb70c6d01003be607) that specifies a position in the [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc).

**ABNF:**

1. PtgJmpLabel = [string-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "164"

string-value is the unique label of the current position.

#### PtgJmpT

The **PtgJmpT** structure is a [control token](#Section_41b89226e2a7471fb70c6d01003be607) that specifies the position of the next token to be evaluated in the [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc) if the token that is currently at the top of the [evaluation stack](#Section_f70486783f2d445fae706da7e34fe640) evaluates to a value of TRUE. If the token that is currently at the top of the evaluation stack evaluates to a value of FALSE, it is popped from the evaluation stack.

**ABNF:**

1. PtgJmpT = [string-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "161"

string-value is the label that specifies the position.

#### PtgMissArg

The **PtgMissArg** structure specifies a function argument that does not have a value.

**ABNF:**

1. PtgMissArg = [empty-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "229"

#### PtgNoOp

The **PtgNoOp** structure is a token that specifies the operand is not added to the [evaluation stack](#Section_f70486783f2d445fae706da7e34fe640).

**ABNF:**

1. PtgNoOp = [[double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd)] token-separator "240"

The double-value, if present, is unused and MUST be ignored.

#### PtgNum

The **PtgNum** structure specifies a double precision [**floating-point number**](#gt_237156df-b9cf-4b8a-9753-98001801a90d).

**ABNF:**

1. PtgNum = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "32"

double-value is a double precision floating-point number.

#### PtgNumCM

The **PtgNumCM** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of centimeters.

**ABNF:**

1. PtgNumCM = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "69"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgNumDft

The **PtgNumDft** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7). If the **DocumentMeasurementSystem** attribute of CT\_Properties in the [App XML Part](#Section_2c93e85a56504f499b49db4ecd19f55e) has a value of "Metric," the unit is millimeters. If **DocumentMeasurementSystem** has a value of "US Units," the unit is inches.

**ABNF:**

1. PtgNumDft = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "64"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgNumF

The **PtgNumF** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of feet.

**ABNF:**

1. PtgNumF = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "66"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgNumFI

The **PtgNumFI** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of feet and inches.

**ABNF:**

1. PtgNumFI = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "67"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgNumI

The **PtgNumI** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of inches.

**ABNF:**

1. PtgNumI = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "65"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgNumKM

The **PtgNumKM** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of kilometers.

**ABNF:**

1. PtgNumKM = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "72"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgNumM

The **PtgNumM** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of meters.

**ABNF:**

1. PtgNumM = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "71"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgNumMI

The **PtgNumMI** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of miles.

**ABNF:**

1. PtgNumMI = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "68"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgNumMM

The **PtgNumMM** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of millimeters.

**ABNF:**

1. PtgNumMM = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "70"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgNumMultiDim

The **PtgNumMultiDim** structure specifies a multidimensional [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7).

**ABNF:**

1. PtgNumMultiDim = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) "," unit "," dimension token-separator "41"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

unit is an integer that specifies the unit of measurement. It MUST be equal to an entry under the ID column in the Parse Token Table as specified in the [Parse Token Table](#Section_1c7c6787214545db854e2a1e1ca24d69) section. The corresponding entry under the Ptg column for that token MUST be a [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b).

dimension is an integer that specifies the number of dimensions.

#### PtgNumNM

The **PtgNumNM** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of nautical miles.

**ABNF:**

1. PtgNumNM = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "76"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgNumPct

The **PtgNumPct** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) expressed as a percentage. The value is normalized such that the value of 1 corresponds to 100 percent.

**ABNF:**

1. PtgNumPct = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "33"

double-value is a double precision [**floating-point number**](#gt_237156df-b9cf-4b8a-9753-98001801a90d).

#### PtgNumYards

The **PtgNumYards** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of yards.

**ABNF:**

1. PtgNumYards = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "75"

double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981).

#### PtgPageDft

The **PtgPageDft** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7).

**ABNF:**

1. PtgPageDft = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "63"

The type of double-value is determined by the unit specified in the **DefaultUnitsPage** attribute of [CT\_Page](#Section_020589d88a1f420c97805dd19dac1cf7) as follows:

| Value | Meaning |
| --- | --- |
| **DefaultUnitsPage** is greater than or equal to 43 and less than or equal to 47 | double-value is a [durationInternalUnitNumber](#Section_79862c6e7c6b4af5b33e935edf99db5b) |
| **DefaultUnitsPage** is greater than or equal to 50 and less than or equal to 54 | double-value is a [typographicInternalUnitNumber](#Section_9ba74244fcbe42229c3a78fcbd9db05e) |
| **DefaultUnitsPage** is greater than or equal to 63 and less than or equal to 76 | double-value is a [lengthInternalUnitNumber](#Section_e88a0a895198463e833075e8f4f1d981) |

#### PtgPnt

The **PtgPnt** token is not supported and MUST NOT be used.

**ABNF:**

1. PtgPnt = [string-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "225"

#### PtgPop

The **PtgPop** token specifies the removal of the top element of the [evaluation stack](#Section_f70486783f2d445fae706da7e34fe640).

**ABNF:**

1. PtgPop = [[string-value](#section_9dd6fede2e2a4b4f9193df3626a781bd)] token-separator "147"

The string-value, if present, is unused and MUST be ignored.

#### PtgPushTop

The **PtgPushTop** token specifies the duplication of the top element of the [evaluation stack](#Section_f70486783f2d445fae706da7e34fe640) and the insertion of this duplicate on to the top of the evaluation stack.

**ABNF:**

1. PtgPushTop = [[string-value](#section_9dd6fede2e2a4b4f9193df3626a781bd)] token-separator "149"

The string-value, if present, is unused and MUST be ignored.

#### PtgRecalcRef

The **PtgRecalcRef** structure specifies a reference to a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc).

**ABNF:**

1. PtgRecalcRef = [string-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "2"

string-value specifies the name of the referenced formula expression and MUST match the name of one of the [CT\_FormulaReferences](#Section_c376a1af36d2469686af16d31ea7afeb) elements in the [Data Graphics XML Part](#Section_23f5bf65461648cfbeda817f02b9d087) of this document.

#### PtgStr1

The **PtgStr1** structure specifies a string.

**ABNF:**

1. PtgStr1 = [string-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "96"

string-value is a string. String length MUST be less than or equal to 65535.

#### PtgTDurDft

The **PtgTDurDft** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7). The unit is specified by the **DefaultUnitsDuration** attribute of [CT\_Page](#Section_020589d88a1f420c97805dd19dac1cf7).

**ABNF:**

1. PtgTDurDft = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "42"

double-value specifies a [durationInternalUnitNumber](#Section_79862c6e7c6b4af5b33e935edf99db5b).

#### PtgTypCD

The **PtgTypCD** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of ciceros and didots.

**ABNF:**

1. PtgTypCD = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "52"

double-value specifies a [typographicInternalUnitNumber](#Section_9ba74244fcbe42229c3a78fcbd9db05e).

#### PtgTypCi

The **PtgTypCi** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of ciceros.

**ABNF:**

1. PtgTypCi = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "54"

double-value specifies a [typographicInternalUnitNumber](#Section_9ba74244fcbe42229c3a78fcbd9db05e).

#### PtgTypDft

The **PtgTypDft** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7). The unit is specified by the **DefaultUnitsText** attribute of [CT\_Page](#Section_020589d88a1f420c97805dd19dac1cf7).

**ABNF:**

1. PtgTypDft = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "48"

double-value is a [typographicInternalUnitNumber](#Section_9ba74244fcbe42229c3a78fcbd9db05e).

#### PtgTypDi

The **PtgTypDi** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of didots.

**ABNF:**

1. PtgTypDi = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "53"

double-value specifies a [typographicInternalUnitNumber](#Section_9ba74244fcbe42229c3a78fcbd9db05e).

#### PtgTypPi

The **PtgTypPi** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of picas.

**ABNF:**

1. PtgTypPi = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "51"

double-value is a [typographicInternalUnitNumber](#Section_9ba74244fcbe42229c3a78fcbd9db05e).

#### PtgTypPP

The **PtgTypPP** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of pica points.

**ABNF:**

1. PtgTypPP = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "49"

double-value is a [typographicInternalUnitNumber](#Section_9ba74244fcbe42229c3a78fcbd9db05e).

#### PtgTypPt

The **PtgTypPt** structure specifies a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) with a unit of points.

**ABNF:**

1. PtgTypPt = [double-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "50"

double-value is a [typographicInternalUnitNumber](#Section_9ba74244fcbe42229c3a78fcbd9db05e).

#### PtgUnsWord

The **PtgUnsWord** structure specifies a 2-byte unsigned integer.

**ABNF:**

1. PtgUnsWord = [unsigned-integer-value](#section_9dd6fede2e2a4b4f9193df3626a781bd) token-separator "98"

unsigned-integer-value specifies an integer and MUST be greater than or equal to 0 and less than or equal to 65535.

### Custom Input Type Definitions

This section specifies custom input types. Custom input types are used to specify the token types for function arguments. Functions that require a particular class of token inputs, and thus require a conversion from many possible source token types, declare arguments with custom input types and rely on the custom input type to derive a valid input argument from a source token. The definition of each custom input type specifies how the source token is derived, the computed properties of the custom input type and under which cases the input argument is invalid.

#### vBoolean

The **vBoolean** custom input type specifies a [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508) that is derived from a source token that MUST be a [vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789), [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), or [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd). It contains the following computed properties.

**Value**

This property specifies a [**Boolean**](#gt_1d79d7a7-ba2c-4b34-931c-7ba8057c87b2) derived from the source token as follows.

If the source token is a PtgBool or PtgStr1 with a value of TRUE (case insensitive), **Value** is equal to TRUE. If the source token is a PtgBool or PtgStr1 with a value of FALSE (case insensitive), **Value** is equal to FALSE.

If the source token is a [vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b) or PtgCy with a value of zero, **Value** is equal to FALSE. If the source token is a vNum, PtgUnsWord or PtgCy with a value not equal to zero, **Value** is equal to TRUE.

If the source token is a PtgMissArg, **Value** is equal to FALSE.

In all other cases, the input argument is not valid.

**Type**

This property specifies a token type of PtgBool.

#### vColor

The **vColor** custom input type specifies a [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882) derived from a source token that MUST be a PtgColorRGB, [vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789), [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), or [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd). It contains the following computed properties.

**Value**

This property specifies an [**RGB**](#gt_2c716d3a-e60b-4e52-bbb0-2fdeb298003b) color value represented as a 4-byte signed integer derived from the source token value as follows.

If the source token type is a PtgColorRGB, **Value** is equal to the source token value.

If the source token can be interpreted as a [vSignedLong](#Section_52e79e59b9eb4e8a8a7e92d0cdbdecf8), **Value** is equal to the color in the [color table](#Section_40036cbbe7fd48c5b634ca12f18a8adb) that is indexed by the value of the source token interpreted as a vSignedLong; if the value of the source token interpreted as a vSignedLong is not a valid index in the color table, **Value** is equal to the color at index zero in the color table.

In all other cases, the input argument is not valid.

**Type**

This property specifies a token type of PtgColorRGB.

#### vDouble

The **vDouble** custom input type specifies a [vNum](#Section_ad426b3dbea2407c8687202b9d13adad) derived from a source token that MUST be a [vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789), [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), or [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd). It contains the following computed properties.

**Value**

This property specifies a double precision [**floating-point number**](#gt_237156df-b9cf-4b8a-9753-98001801a90d) derived from the source token value as follows.

If the source token value can be converted to a double, as described in [[MSDN-ToDouble]](https://go.microsoft.com/fwlink/?LinkId=153666), **Value** is the result of the conversion.

If the source token value is equal to TRUE (case insensitive), **Value** is equal to 1. If the source token value is equal to FALSE (case insensitive), **Value** is equal to 0.

If the source token is a PtgMissArg, **Value** is equal to 0.

In all other cases, the input argument is not valid.

**Type**

This property specifies a token type. If the source token is a [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b), **Type** is equal to the token type of the source token. Otherwise, **Type** is equal to the token type of [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0).

**Dimension**

This property specifies the dimension of the value. If the source token type is a [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84), **Dimension** is equal to the dimension of the source token. If the source token is any other vUnitType, **Dimension** is equal to one. Otherwise, **Dimension** is equal to zero.

**Unit**

This property specifies the measurement unit of the value. If the source token is a PtgNumMultiDim, **Unit** is equal to the unit of the source token. Otherwise, **Unit** is equal to **Type**.

#### vDoubleEx

The **vDoubleEx** custom input type specifies either a [vNum](#Section_ad426b3dbea2407c8687202b9d13adad) or [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3) derived from a source token that MUST be a [vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789), [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), PtgCy, or [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd). It is the [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d) custom input type with the following exceptions and additional computed properties.

**Type**

This property specifies a token type. If the source token is a PtgCy, **Type** is equal to the token type of PtgCy. Otherwise, **Type** is equal to the type of the source token interpreted as a vDouble.

**Currency**

This property specifies the currency ID. If the source token is a PtgCy, **Currency** is equal to the currency of the source token. Otherwise, **Currency** is equal to 0 which specifies the unknown currency type, as described in [vCurrencyID](#Section_b13bb6692bc04291b31c4532158fa675).

#### vFloat

The **vFloat** custom input type specifies a single precision [**floating-point number**](#gt_237156df-b9cf-4b8a-9753-98001801a90d) derived from a source token that MUST be a [vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789), [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), or [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd). It is a [vDouble](#Section_6b18e891cafc4cea9e6f2daaf85c5f2d) with the additional restriction that the **Value** property MUST conform to the range of values specified for a single precision floating-point number, as defined in [[IEEE754]](https://go.microsoft.com/fwlink/?LinkId=89903).

If the source token is not a vDouble or the value of the source token interpreted as a vDouble does not conform to the range of values specified for a single precision floating-point number, the input argument is not valid.

#### vSignedInt

The **vSignedInt** custom input type specifies a signed integer derived from a source token that MUST be a [vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789), [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), or [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd). It is a [vSignedLong](#Section_52e79e59b9eb4e8a8a7e92d0cdbdecf8) with the additional restriction that the **Value** property MUST be greater than or equal to -32768 and less than or equal to 32767.

If the source token is not a vSignedLong or the value of the source token interpreted as a vSignedLong is less than -32768 or greater than 32767, the input argument is not valid.

#### vSignedLong

The **vSignedLong** custom input type specifies a signed long integer derived from a source token that MUST be a [vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789), [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), or a [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd). It contains the following computed properties.

**Value**

This property specifies a signed long integer derived from the source token value as follows.

If the source token value can be converted to a double, as described in [[MSDN-ToDouble]](https://go.microsoft.com/fwlink/?LinkId=153666), **Value** is equal to a signed long integer calculated as follows.

Use the conversion method described in [MSDN-ToDouble] to obtain a double value. If the double value is not an integer, round the value towards zero to the next integer. If the integer is less than -2147483648 or greater than 2147483647, the input argument is not valid.

If the source token value is equal to TRUE (case insensitive), **Value** is equal to 1. If the source token value is equal to FALSE (case insensitive), **Value** is equal to 0.

If the source token is a PtgMissArg, **Value** is equal to 0.

In all other cases, the input argument is not valid.

**Type**

This property specifies a token type. If the source token is a vNumAny, **Type** is equal to the token type of the source token. Otherwise, **Type** is equal to the token type of [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0).

#### vString

The **vString** custom input type specifies a [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69) derived from a source token that MUST be a [vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789), PtgStr1, or [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd). It contains the following computed properties.

**Value**

This property specifies a string derived from the source token value as follows.

If the source token is a PtgStr1, **Value** is equal to the source token value.

If the source token is a vNumAny, **Value** is the string form of the source token value.

If the source token is a PtgMissArg, **Value** is equal to the empty string.

In all other cases, the input argument is not valid.

**Type**

This property specifies a token type of PtgStr1.

#### vUnsignedInt

The **vUnsignedInt** custom input type specifies an unsigned integer derived from a source token that MUST be a [vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789), [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), or a [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd). It is a [vUnsignedLong](#Section_1bf83d6915fa4b6dad66865028333cae) with the additional restriction that **Value** MUST be less than 65536. **Value** is equal to the result of the value of the source token interpreted as a [vSignedLong](#Section_52e79e59b9eb4e8a8a7e92d0cdbdecf8) modulo 65536.

#### vUnsignedLong

The **vUnsignedLong** custom input type specifies an unsigned long integer derived from a source token that MUST be a [vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789), [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), or a [PtgMissArg](#Section_c613ca4769e54f2aa8a84f0c9c8530cd). It contains the following computed properties.

**Value**

This property specifies an unsigned long integer derived from the source token value as follows.

If the source token value can be converted to a double, as described in [[MSDN-ToDouble]](https://go.microsoft.com/fwlink/?LinkId=153666), **Value** is equal to an unsigned long integer calculated as follows.

Use the conversion method described in [MSDN-ToDouble] to obtain a double value. If the double value is not an integer, round the value towards zero to the next integer. If the integer is greater than or equal to zero, **Value** is equal to the integer modulo 4294967296. Otherwise **Value** is equal to the sum of the integer modulo 4294967296 and 4294967296.

If the source token value is equal to TRUE (case insensitive), **Value** is equal to 1. If the source token value is equal to FALSE (case insensitive), **Value** is equal to 0.

If the source token is a PtgMissArg, **Value** is equal to 0.

In all other cases, the input argument is not valid.

**Type**

This property specifies a token type. If the source token is a [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b), **Type** is equal to the token type of the source token. Otherwise, **Type** is equal to the token type of [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0).

### Custom Token Groupings

This section defines groupings of tokens, referred to as custom token groupings. A custom token grouping specifies a set of tokens that represents a specific concept. Custom token groupings can contain other custom token groupings. The association of a token to a custom token grouping is not exclusive. These grouping do not exist in the format of the file. The purpose of these groupings is to improve the readability of the document by allowing tokens representing similar concepts to be referred to collectively.

#### vAngle

The **vAngle** custom token grouping is an aggregation of types that represent an angle. The vAngle custom token grouping contains the following tokens.

[PtgAngDD](#Section_222188be55a34b90b2c39eb9f5878bab), [PtgAngDft](#Section_57e444144344437fb32a6cf2d3b20d2f), [PtgAngDMS](#Section_459da555a74b48439f4cbc516a043d30), [PtgAngRad](#Section_a09cdad2084647eb97edef0ef25726e0)

#### vAny

The **vAny** custom token grouping is an aggregation of types that represent data. The vAny custom token grouping contains the following tokens.

[vNumAny](#Section_1d4fd8bc32844cb5915e14424a11a789), [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882), [PtgCy](#Section_a0035c3665bf4a2aa677a9d3810e85c3), [PtgErr](#Section_a2cbc6aecf6f4d0eac103b71a11cf6d2)

#### vNum

The **vNum** custom token grouping is an aggregation of types that represent either a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7) or a double precision [**floating-point number**](#gt_237156df-b9cf-4b8a-9753-98001801a90d). The vNum custom token grouping contains the following tokens.

[vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b), [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0)

#### vNumAny

The **vNumAny** custom token grouping is an aggregation of types that represent any number. The vNumAny custom token grouping contains the following tokens.

[vNum](#Section_ad426b3dbea2407c8687202b9d13adad), [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508), [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b)

#### vUnitType

The **vUnitType** custom token grouping is an aggregation of types that represent a [Unit Number](#Section_b455d99896fc43d3822a35f666f2b2d7). The vUnitType custom token grouping contains the following tokens.

[vAngle](#Section_8609b9afc5aa4a0e9c37b1f7929ac9b4), [PtgAcre](#Section_7f6eff5e434a407b8565fe966e0b066c), [PtgDate](#Section_661d6eb308e34587b15343cf0ad0a1f6), [PtgEDay](#Section_f6690609c80a49ed9fffe0dbcfb262a6), [PtgEHour](#Section_e31838fa14684d478f193c033df45e4c), [PtgEMin](#Section_f58cc7ec8a1246d89edb4b3e00a08ac6), [PtgESec](#Section_01c1da7178fc485c8ee6b48674eeb35b), [PtgEWeek](#Section_4aae0088510448abaf3f8ec63f95fd5f), [PtgHectare](#Section_7b405e52f9c34451a7f90ad394edd4aa), [PtgNumCM](#Section_bc2d88fb83a94d9588803acf4c0cdc27), [PtgNumDft](#Section_de132a3cdab04751844609cf15ec3756), [PtgNumF](#Section_09bdcb1c922444c8b9c3ecdaa03e0c6d), [PtgNumFI](#Section_a3917f0c42e94c71b55c70ea3aed6819), [PtgNumI](#Section_ac87036f348249b58cba49659d7598a8), [PtgNumKM](#Section_6dc5f00e5e7945678ac890f98cd40706), [PtgNumM](#Section_000148cf28984a299ade5cc03f395c6c), [PtgNumMI](#Section_e1f97f6e268c4429868c74e9323a7b9e), [PtgNumMM](#Section_5fc97e8093e041d7bcd6ddd2c66aa589), [PtgNumMultiDim](#Section_e93d3e0048ec4f74a46336021c9e5f84), [PtgNumNM](#Section_bf29f1f677664e9e856811466228244f), [PtgNumPct](#Section_ef65a2675fd84ba3a1448fb7698f6551), [PtgNumYards](#Section_7039749a686144d9bcd9da3946769b16), [PtgPageDft](#Section_fe0d479f9b534257a3d13618b7314aaa), [PtgTDurDft](#Section_1200eb57709c4fcc9d20e2bac20fa493), [PtgTypCD](#Section_bf50076676d84a8d900468c81b728b8e), [PtgTypCi](#Section_c5f1913417a7448287105b162359a1fa), [PtgTypDft](#Section_d66b059e323d45739c91ec57ed148b16), [PtgTypDi](#Section_a56c1d526fe9452b9857c97fd84c3825), [PtgTypPi](#Section_f3c4aabfb3bb4c68b5b9f175f9620048), [PtgTypPP](#Section_997b3916e4734908b63936fd05317e60), [PtgTypPt](#Section_b30a64caae964c76ac0728982f26bb31)

### Custom Internal Unit Types

This section specifies internal unit types. All internal unit values MUST conform to a double precision [**floating-point number**](#gt_237156df-b9cf-4b8a-9753-98001801a90d).

#### angleInternalUnitNumber

The **angleInternalUnitNumber** custom internal unit type specifies a value with a unit of radians.

#### durationInternalUnitNumber

The **durationInternalUnitNumber** custom internal unit type specifies a value with a unit of days.

#### lengthInternalUnitNumber

The **lengthInternalUnitNumber** custom internal unit type specifies a value with a unit of inches.

#### typographicInternalUnitNumber

The **typographicInternalUnitNumber** custom internal unit type specifies a value with a unit of inches.

### Custom Structures

This section specifies custom structures. A custom structure specifies a type and a set of valid values.

#### vCalendar

The **vCalendar** custom structure is an unsigned integer that specifies the calendar system to use when formatting dates and times.

It MUST be a value from the following table.

| Value | Description |
| --- | --- |
| 0 | Western |
| 1 | Arabic Hijri |
| 2 | Hebrew Lunar |
| 3 | Chinese National |
| 4 | Japanese Emperor |
| 5 | Thai Buddhist |
| 6 | Korean Danki |
| 7 | Japanese Saka Era |
| 8 | Transliterated English |
| 9 | Transliterated French |
| 10 | Gregorian US English |
| 11 | Gregorian Middle East French |
| 12 | Gregorian Arabic |
| 13 | Um-al-Qura |

#### vCurrencyID

The **vCurrencyID** custom structure is an unsigned integer that specifies a currency.

It MUST be a value from the following table.

| Value | Meaning | Value | Meaning |
| --- | --- | --- | --- |
| 0 | Unknown currency type | 1 | Omit currency symbol |
| 10 | European Union: euro | 11 | United States: dollar |
| 12 | Austria: schilling | 13 | Australia: dollar |
| 14 | Belgium: franc | 15 | Canada: dollar |
| 16 | Switzerland: franc | 17 | China (Mainland): yuan |
| 18 | Germany: mark | 19 | Denmark: krone |
| 20 | Spain: peseta | 21 | Finland: markka |
| 22 | France: franc | 23 | United Kingdom: pound |
| 24 | Greece: drachma | 25 | Hong Kong SAR: dollar |
| 26 | Hungary: forint | 27 | Indonesia: rupiah |
| 28 | Ireland: punt | 29 | Israel: shekel |
| 30 | Italy: lira | 31 | Japan: yen |
| 32 | Korea, Republic of (South): won | 33 | Luxembourg: franc |
| 34 | Mexico: peso | 35 | Malaysia: ringgit |
| 36 | Netherlands: guilder | 37 | Norway: krone |
| 38 | New Zealand: dollar | 39 | Philippines: peso |
| 40 | Poland: zloty (obsolete, use 89) | 41 | Portugal: escudo |
| 42 | Romania: leu | 43 | Russia: rouble (obsolete, use 90) |
| 44 | Sweden: kroner | 45 | Singapore: dollar |
| 46 | Thailand: baht | 47 | Taiwan: dollar |
| 48 | European Currency Unit (ECU) | 49 | Yugoslavia: dinar (obsolete, use 91) |
| 50 | South Africa: rand | 56 | Argentina: peso |
| 57 | Bermuda: dollar | 58 | Bolivia: boliviano |
| 59 | Brazil: cruzeiro real (obsolete, use 88) | 60 | Bahamas: dollar |
| 61 | Chile: peso | 62 | Colombia: peso |
| 63 | Costa Rica: colon | 64 | Czech Republic: koruna |
| 65 | Dominican Republic: peso | 66 | Ecuador: sucre |
| 67 | Egypt: pound | 68 | Honduras: lempira |
| 69 | India: rupee | 70 | Jamaica: dollar |
| 71 | Jordan: dinar | 72 | Kuwait: dinar |
| 73 | Macao SAR: pataca | 74 | Nicaragua: cordoba oro |
| 75 | Panama: balboa | 76 | Peru: nuevo sol |
| 77 | Pakistan: rupee | 78 | Paraguay: guarani |
| 79 | Saudi Arabia: riyal | 80 | Slovenia: tolar |
| 81 | Slovakia: koruna | 82 | El Salvador: colon |
| 83 | Turkey: new lira | 84 | Trinidad and Tobago: dollar |
| 85 | Uruguay: peso | 86 | Venezuela: bolivar |
| 87 | Viet Nam: dong | 88 | Brazil: real |
| 89 | Poland: zloty | 90 | Russia: rouble |
| 91 | Serbia: dinar | 92 | Belarus: ruble |
| 93 | Ukraine: hryvnia | 94 | Afghanistan: afghani |
| 95 | Albania: lek | 96 | Algeria: dinar |
| 97 | Andorra: peseta | 98 | Angola: kwanza |
| 99 | East Caribbean Dollar | 100 | Armenia: dram |
| 101 | Aruba: guilder | 102 | Azerbaijan: manat |
| 103 | Bahrain: dinar | 104 | Bangladesh: taka |
| 105 | Barbados: dollar | 106 | Belarus: ruble |
| 107 | Belize: dollar | 108 | CFA Franc BCEAO |
| 109 | Bhutan: ngultrum | 110 | Bosnia and Herzegovina: convertible marks |
| 111 | Botswana: pula | 112 | Brunei: dollar |
| 113 | Bulgaria: lev (historic) | 114 | Bulgaria: lev |
| 115 | Burundi: franc | 116 | Cambodia: riel |
| 117 | CFA Franc BEAC | 118 | Cape Verde: escudo |
| 119 | Cayman Islands: dollar | 120 | Comoros: franc |
| 121 | Congo (DRC): franc | 122 | Croatia: kuna |
| 123 | Cuba: peso | 124 | Cyprus: pound |
| 125 | Djibouti: franc | 126 | Timor-Leste: escudo |
| 127 | Eritrea: nakfa | 128 | Estonia: kroon |
| 129 | Ethiopia: birr | 130 | Falkland Islands (Islas Malvinas): pound |
| 131 | Fiji Islands: dollar | 132 | CFP Franc |
| 133 | The Gambia: dalasi | 134 | Georgia: lari |
| 135 | Ghana: cedi | 136 | Gibraltar: pound |
| 137 | Guatemala: quetzal | 138 | Guinea: franc |
| 139 | Guinea-Bissau: peso | 140 | Guyana: dollar |
| 141 | Haiti: gourde | 142 | Iceland: krona |
| 143 | Iran: rial | 144 | Iraq: dinar |
| 145 | Kazakhstan: tenge | 146 | Kenya: shilling |
| 147 | North Korean: won | 148 | Kyrgyzstan: som |
| 149 | Laos: kip | 150 | Latvia: lats |
| 151 | Lebanon: pound | 152 | Lesotho: loti |
| 153 | Liberia: dollar | 154 | Libya: dinar |
| 155 | Lithuania: litus | 156 | Former Yugoslav Republic of Macedonia: denar |
| 157 | Madagascar: franc | 158 | Malawi: kwacha |
| 159 | Maldives: rufiyaa | 160 | Malta: lira |
| 161 | Mauritania: ouguiya | 162 | Mauritius: rupee |
| 163 | Moldova: leu | 164 | Mongolia: tugrik |
| 165 | Morocco: dirham | 166 | Mozambique: metical |
| 167 | Myanmar: kyat | 168 | Namibia: dollar |
| 169 | Nepal: rupee | 170 | Bonaire, Curaçao, Saba, Sint Eustatius, and Sint Maarten: guilder |
| 171 | Nigeria: naira | 172 | Oman: rial |
| 173 | Papua New Guinea: kina | 174 | Qatar: rial |
| 175 | Rwanda: franc | 176 | Saint Helena, Ascension and Tristan da Cunha: pound |
| 177 | Samoa: tala | 178 | São Tomé and Príncipe: dobra |
| 179 | Seychelles: rupee | 180 | Seirra Leone: leone |
| 181 | Solomon Islands: dollar | 182 | Somalia: shilling |
| 183 | Sri Lanka: rupee | 184 | Sudan: dinar |
| 185 | Suriname: guilder | 186 | Swaziland: lilangeni |
| 187 | Syria: pound | 188 | Tajikistan: ruble |
| 189 | Tajikistan: somoni | 190 | Tanzania: shilling |
| 191 | Tonga: pa'anga | 192 | Tunisia: dinar |
| 193 | Turkmenistan: manat | 194 | Uganda: shilling |
| 195 | United Arab Emirates: dirham | 196 | Uzbekistan: sum |
| 197 | Vanuatu: vatu | 198 | Yemen: rial |
| 199 | Zambia: kwacha | 200 | Zimbabwe: dollar |
| 201 | Venezuela: bolivar fuerte | 202 | Madagascar: ariary |
| 203 | Serbia: dinar |  |  |

#### vFormatString

The **vFormatString** custom structure is a string that specifies the formatting information to use when representing a value as a string. It can be used to format either a numeric value or a date and time value.

The formatting information has the following syntax when formatting a numeric value.

{*index*[:*formatString*]} [{*index*}]

Elements in square brackets are optional. Curly brackets are required characters.

The formatting information has the following syntax when formatting a date and time value.

*formatString*

*index* and *formatString* fields are described in the following table.

| Value | Meaning |
| --- | --- |
| *index* | The item to be formatted. Index zero is the value to be displayed, and index 1 is the optional display unit string. MUST be 0 or 1. |
| *formatString* | A numeric format string or a date and time format string, as described in [[MSDN-FormattingTypes]](https://go.microsoft.com/fwlink/?LinkId=153833). |

#### vLanguageID

The **vLanguageID** custom structure is an unsigned integer that specifies an [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6).

#### vServerAction

The **vServerAction** custom structure is an unsigned integer that specifies the custom formatting to apply to a string.

It MUST be a value from the following table.

| Value | Meaning |
| --- | --- |
| 0 | Apply no custom formatting |
| 1 | Convert string to upper case |
| 2 | Convert string to lower case |

#### vUnitLabel

The **vUnitLabel** custom structure is an unsigned integer that specifies the form and casing [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) for formatting the unit of a numeric value as a string. Casing is performed using a [**culture name**](#gt_b43d5a25-2069-42c0-bb00-b2b70accebac) of "".

It MUST be a value from the following table.

| Value | Meaning |
| --- | --- |
| 0 | No unit displayed |
| 1 | Localized abbreviated form, upper case |
| 2 | Localized long form, upper case |
| 3 | Universal upper case |
| 4 | Localized abbreviated form, lower case |
| 5 | Localized long form, lower case |
| 6 | Universal lower case |

# Structure Examples

The Visio Graphics Service (.vdw) File Format contains a "VisioServerData" [stream](#Section_1fc77f54c9d74549aca5a7bed511c016) as well as other supporting [**streams**](#gt_f3529cd8-50da-4f36-aa0b-66af455edbb6). This section describes what some of the components within the "VisioServerData" stream look like for some simple documents.

## Document with a Shape on a Page

This section describes some of the contents of the "VisioServerData" [stream](#Section_1fc77f54c9d74549aca5a7bed511c016) for a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25) consisting of one [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) with a rectangle with blue fill color and the text "Sample Text" on it.

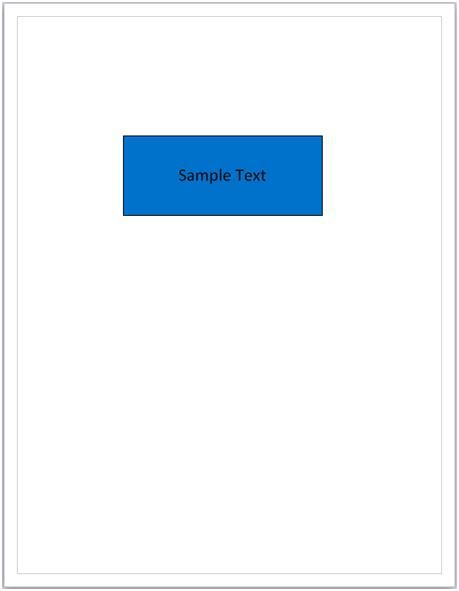


Figure 2: Document with Shape on a Page

### App XML Part

1. <Properties
2. xmlns="http://schemas.openxmlformats.org/officeDocument/2006/extended-properties" xmlns:vt="http://schemas.openxmlformats.org/officeDocument/2006/docPropsVTypes">
3. <Application>Microsoft Visio 2010</Application>
4. <Company>Microsoft Corporation</Company>
5. <AppVersion>14.0.4023.1000</AppVersion>
6. <DocumentVersion>14.0.14</DocumentVersion>
7. <DocumentLanguage>1033</DocumentLanguage>
8. <DocumentMeasurementSystem>US Units</DocumentMeasurementSystem>
9. <DocumentCurrencyID>11</DocumentCurrencyID>
10. <PagesMetaData>
11. <PageMetaData ID="1" Name="Page-1"/>
12. </PagesMetaData>
13. </Properties>

The following table provides more information about element values in the preceding sample [App XML](#Section_2c93e85a56504f499b49db4ecd19f55e).

| Element Name | Value | Notes |
| --- | --- | --- |
| Application | Microsoft Visio 2010 | The name of the application that created the document is "Microsoft Visio 2010". |
| Company | Microsoft Corporation | The name of the company that created the document is "Microsoft Corporation". |
| AppVersion | 14.0.4023.1000 | Version of the application that created the document is 14.0.4023.1000. |
| DocumentVersion | 14.0.14 | Version of the document is 14.0.14. |
| DocumentLanguage | 1033 | [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) of the document is 1033, which is the .Net language identifier for US English. |
| DocumentMeasurementSystem | US Units | Measurement system of the document is US Units. |
| DocumentCurrencyID | 11 | The [currency ID](#Section_b13bb6692bc04291b31c4532158fa675) that is used in the document is 11, which is the identifier for United States dollar. |
| ID | 1 | Index of the drawing page is 1. |
| Name | Page-1 | Name of the drawing page is "Page-1". |

### ShapeInfo XML Part

1. <Page
2. Name="Page-1"
3. Zoom="-1.000000"
4. OffsetX="407"
5. OffsetY="528"
6. DefaultUnitsText="50"
7. DefaultUnitsAngle="81"
8. DefaultUnitsDuration="44"
9. DefaultUnitsPage="73">
10. <Pages>
11. <PageInfo Name="Page-1" />
12. </Pages>
13. <ShapeInfo
14. Name="\_1"
15. DisplayName="Sheet.1"
16. Layout="{"type":"2D","bounds":{"x":215,"y":239,"width":362,"height":146}}" />
17. </Page>

The following table provides more information about attribute values in the preceding sample [ShapeInfo XML](#Section_bf230dbc228e4108bd8f93c9cbd4fcac).

| Attribute Name | Value | Notes |
| --- | --- | --- |
| Name | Page-1 | The name of the [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) is "Page-1". |
| Zoom | -1.000000 | The [**zoom level**](#gt_bfb33471-a018-422b-bc63-177c8bc1831f) at which the drawing page is initially displayed is -1.0, which specifies that the drawing page is zoomed to fit the entire drawing page in the [**view**](#gt_3f793b0b-9509-4df0-89f9-92f07954beb8). |
| OffsetX | 407 | The x-coordinate in pixels of the point on the drawing page that is centered in the view when the drawing page is initially displayed is 407. The value is relative to the top, left corner of the drawing page and is expressed in the coordinate system of the view that displays the drawing page. |
| OffsetY | 528 | The y-coordinate in pixels of the point on the drawing page that is centered in the view when the drawing page is initially displayed is 528. The value is relative to the top, left corner of the drawing page and is expressed in the coordinate system of the view that displays the drawing page. |
| DefaultUnitsText | 50 | "Points" is used as the default typographic measurement unit in [formula evaluations](#Section_81f0fbe77a314d34b52f5fa9c9380c33) for this drawing page. |
| DefaultUnitsAngle | 81 | "Decimal degrees" is used as the default angle measurement unit in formula evaluations for this drawing page. |
| DefaultUnitsDuration | 44 | "Days" is used as the default duration measurement unit in formula evaluations for this drawing page. |
| DefaultUnitsPage | 73 | "Inches (fractional)" is used as the default unit for length measurement in formula evaluations for this drawing page. |
| Name | \_1 | The [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) identifier is "\_1". |
| DisplayName | Sheet.1 | The shape name used for display purposes is "Sheet.1". |
| Layout | {"type":"2D","bounds":{"x":215,"y":239,"width":362,"height":146}} | 215 and 239 are the upper left x-coordinate and y-coordinate, respectively, of the [**bounding rectangle**](#gt_e349a155-0fb9-4ba1-abd4-e363e21abcd1) of the shape in [**pixels**](#gt_fe932605-d3bc-442f-9239-594676bce6ba). 362 and 146 are the width and height, respectively, of the bounding rectangle of the shape in pixels. |

### ShapeOutline XML Part

1. <Page>
2. <Shapes>
3. <Shape Name="\_1">
4. <Path Shape="poly" Data="216,384,576,384,576,240,216,240,216,384"/>
5. </Shape>
6. </Shapes>
7. </Page>

The following table provides more information about attribute values in the preceding sample [ShapeOutline XML](#Section_e81d9654a5d846cfbba59b380b3a8ade).

| Attribute Name | Value | Notes |
| --- | --- | --- |
| Name | \_1 | The [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) identifier of the shape is "\_1". |
| Shape | Poly | The type of outline specified for the shape is polygon. |
| Data | 216,384,576,384,576,240,216,240,216,384 | The Cartesian coordinates in [**pixels**](#gt_fe932605-d3bc-442f-9239-594676bce6ba) of each vertex of the polygon are (216,384), (576,384), (576,240), (216,240). The first vertex (216,384) is repeated to close the polygon. |

### XAML

1. <Canvas
2. Height="1056"
3. Name="Workspace"
4. Width="816"
5. xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation">
6. <Canvas Name="D">
7. <Canvas.Background>
8. <SolidColorBrush Color="#FFFFFFFF"/>
9. </Canvas.Background>
10. <Canvas
11. Tag="Shape"
12. Name="\_1"
13. RenderTransform="1, 0, 0, 1, 216, -672">
14. <Rectangle
15. Canvas.Left="-0.5"
16. Canvas.Top="911.5"
17. Height="145"
18. Name="\_1\_G0\_P0"
19. Width="361"
20. StrokeThickness="1"
21. StrokeDashCap="Round"
22. StrokeEndLineCap="Round"
23. StrokeLineJoin="Round"
24. StrokeStartLineCap="Round">
25. <Rectangle.Stroke>
26. <SolidColorBrush Color="#FF000000"/>
27. </Rectangle.Stroke>
28. <Rectangle.Fill>
29. <SolidColorBrush Color="#FF0070C0"/>
30. </Rectangle.Fill>
31. </Rectangle>
32. <Canvas>
33. <Glyphs
34. FontRenderingEmSize="32"
35. FontUri="CommonFonts/4C1AABE5-BA71-4F2C-8263-66731F995C4B.odttf"
36. OriginX="100.5"
37. OriginY="993.6"
38. UnicodeString="Sample Text">
39. <Glyphs.Fill>
40. <SolidColorBrush Color="#FF000000"/>
41. </Glyphs.Fill>
42. </Glyphs>
43. </Canvas>
44. </Canvas>
45. </Canvas>
46. </Canvas>

The following table provides more information about elements in the preceding sample [XAML](#Section_d86834c499d248e2b13ab2c8f016eca0).

| Node | Notes |
| --- | --- |
| Top most canvas node with name "Workspace" | Canvas for the container of the whole [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640). 816 and 1056 are the width and height respectively at 100% zoom in [**pixels**](#gt_fe932605-d3bc-442f-9239-594676bce6ba). |
| Canvas node with name "D" | Canvas for the drawing page. |
| Canvas node with name "\_1" | Canvas for the [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b). Identifier of the shape is "\_1". |
| Rectangle node | Graphical representation of the shape. |
| Rectangle.Stroke node | The line color of the shape is black. |
| Rectangle.Fill node | The fill color of the shape is blue. |
| Canvas node with Glyphs element as its child node | Canvas that groups all the text of the shape. |
| Glyphs node | Representation of the text in the shape. |
| FontRenderingEmSize attribute | The pixel size of the text is 32. |
| FontUri attribute | [**Font**](#gt_f8aa3f46-99d1-49bb-858f-b4bfa546c1c2) of the text is Calibri. |
| UnicodeString attribute | The text in the shape is "Sample Text". |
| Glyphs.Fill node | The color of the text is black. |

## Document with Recalculated Visual Properties

This section describes components of the "VisioServerData" [stream](#Section_1fc77f54c9d74549aca5a7bed511c016) for a [Web drawing](#Section_124a373ef5ca47bfbde823c9db618c25) consisting of one [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) containing a rectangle [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) with [text element](#Section_b061c2bde75d48edb51047479b51c6ec) and [formatting element](#Section_b08c0179146b4b2dbdce35c55c7f7737) [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075). The rectangle has a [data binding](#Section_b9b194892da24d77a8a51c99bb866fe1) to a cell within a [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) of data in a [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) named "Current Product List" in the "Northwind" database. The two [**fields**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc) that identify the cells within the row are "ProductID" and "ProductName". The "ProductID" field is used as the [**primary key**](#gt_e4d8c530-39c1-4fc6-8ccc-8d51a221158d). The rectangle shape has associated text that displays the label "ProductName" and its value. Also, the fill color of the rectangle shape is determined by the value of the "ProductName" field as specified in the following table.

| Value of the "ProductName" field | Fill color of the rectangle |
| --- | --- |
| "Chai" | Red |
| "Coffee" | Yellow |
| Neither "Chai" nor "Coffee" | Blue |

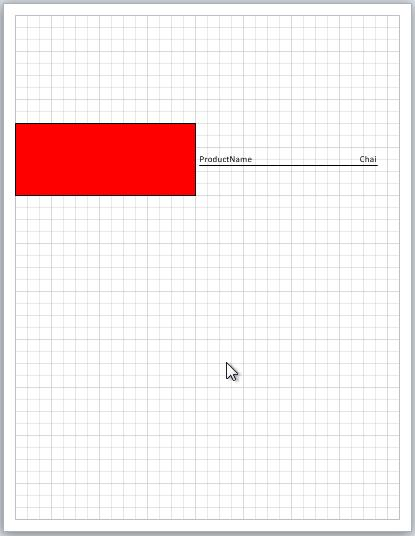


Figure 3: Document with Recalculated Visual Properties

### DataBinding XML Part

1. <BindingConnections>
2. <BindingConnection RecordsetID="2">
3. <Bindings>
4. <Binding BindingID="1">
5. <PrimaryKeys>
6. <PrimaryKey Value="1" />
7. </PrimaryKeys>
8. </Binding>
9. </Bindings>
10. </BindingConnection>
11. </BindingConnections>

The following table provides more information about attribute values in the preceding sample [DataBinding XML](#Section_311514cd8ef444c9ab42562671155333).

| Attribute Name | Value | Notes |
| --- | --- | --- |
| RecordsetID | 2 | The identifier of the [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) that has [**rows**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) of data bound to [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) is 2. |
| BindingID | 1 | The [CT\_Binding](#Section_5ad7ddf78438400abe979591e0cca24a) identifier is 1. |
| Value | 1 | The value of the [**field**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc) that is a component of the [**primary key**](#gt_e4d8c530-39c1-4fc6-8ccc-8d51a221158d) (ProductID) is 1. |

### DataConnection XML Part

1. <Connections>
2. <DataConnections>
3. <DataConnection ID="1" ConnectionString="Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security Info=True;Initial Catalog=Northwind;Data Source=SQLServerDataSource;Use Procedure for Prepare=1;Auto Translate=True;Packet Size=4096;Workstation ID=MACHINENAME;Use Encryption for Data=False;Tag with column collation when possible=False" />
4. </DataConnections>
5. <DataRecordsets>
6. <DataRecordset ID="2" ConnectionID="1" Command="select \* from "Northwind"."dbo"."Current Product List"" Pages="1">
7. <PrimaryKeys>
8. <PrimaryKey Key="ProductID" />
9. </PrimaryKeys>
10. <DataColumns>
11. <DataColumn Name="ProductID" DisplayName="ProductID" Type="3" />
12. <DataColumn Name="ProductName" DisplayName="ProductName" Type="202" />
13. </DataColumns>
14. </DataRecordset>
15. </DataRecordsets>
16. </Connections>

The following table provides more information about attribute values in the preceding sample [DataConnection XML](#Section_c489eefb7a014941bc37a4d332e66dc8).

| Attribute Name | Value | Notes |
| --- | --- | --- |
| ID | 1 | The identifier of the [data connection](#Section_28d3d364de644cada48144231aed4d7b) is 1. |
| ConnectionString | Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security Info=True;Initial Catalog=Northwind;Data Source=SQLServerDataSource;Use Procedure for Prepare=1;Auto Translate=True;Packet Size=4096;Workstation ID=MACHINENAME;Use Encryption for Data=False;Tag with column collation when possible=False | The [**connection string**](#gt_03a9d0ca-2f10-4f3d-b910-052714a96f7d) to the [**data source**](#gt_e091613c-6901-4874-b9b2-27273ead1075) is "Provider=SQLOLEDB.1;Integrated Security=SSPI;Persist Security Info=True;Initial Catalog=Northwind;Data Source=SQLServerDataSource;Use Procedure for Prepare=1;Auto Translate=True;Packet Size=4096;Workstation ID=MACHINENAME;Use Encryption for Data=False;Tag with column collation when possible=False". |
| ID | 2 | The identifier of the [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) is 2. |
| ConnectionID | 1 | The identifier of the data connection corresponding to the recordset is 1. |
| Command | select \* from "Northwind"."dbo"."Current Product List" | The [**query**](#gt_37fbc661-f744-48fa-9d8e-f34513cab9c2) for the recordset is "select \* from "Northwind"."dbo"."Current Product List"". |
| Pages | 1 | The [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) with index 1 is the only drawing page that contains [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) bound to data in the recordset. |
| Key | ProductID | A [**field**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc) named "ProductID" is a component of the [**primary key**](#gt_e4d8c530-39c1-4fc6-8ccc-8d51a221158d) of the recordset. |
| Name | ProductID | The recordset has a field named "ProductID". |
| DisplayName | ProductID | The display name of the field is "ProductID". |
| Type | 3 | The [**ADO**](#gt_3477d72a-874b-4a2a-b6a9-1431c128df49) [**data type**](#gt_0d327bc0-a461-4723-8beb-d935fdab4895) of the field is integer. |
| Name | ProductName | The recordset has a field named "ProductName". |
| DisplayName | ProductName | The display name of the field is "ProductName". |
| Type | 202 | The ADO data type of the field is wide variable character. |

### DataGraphic XML Part

1. <DataGraphics ColorTable="0,0,0 255,255,255 255,0,0 0,255,0 0,0,255 255,255,0 255,0,255 0,255,255 128,0,0 0,128,0 0,0,128 128,128,0 128,0,128 0,128,128 192,192,192 230,230,230 205,205,205 179,179,179 154,154,154 128,128,128 102,102,102 77,77,77 51,51,51 26,26,26 0,112,192 150,175,207 191,206,225 0,145,251 240,240,240 0,0,0 166,210,74 200,227,142 31,71,125 69,91,23 157,187,97 255,255,255 163,184,204 54,73,89 204,0,68 69,94,115 255,0,0 255,60,60 ">
2. <Page
3. PageName="Page-1"
4. ViewTransform="96.000000 0.000000 0.000000 -96.000000 0.000000 1056.000000">
5. <Shape
6. ID="{020D279F-6024-4139-8CDF-A7637F3217E7}"
7. ShapeName="\_1"
8. DataRefs="1,ProductName">
9. <Formulas>
10. <Sheet
11. Name="\_1"
12. FillBackground="1.242.131.0:2 1.5:32 <>:14 173519920:160 1.1.3.0:2 191:98 206:98 225:98 RGB():128 150:98 175:98 207:98 RGB():128 LUMDIFF():128 TINT():128 173519989:163 173519920:164 :147 FALSE:97 CELLISTHEMED():128 173519972:160 191:98 206:98 225:98 RGB():128 173519988:163 173519972:164 :147 255:98 60:98 60:98 RGB():128 173519988:164 173519989:164 THEMEGUARD():128"
13. FillForeground="1.242.131.0:2 1.5:32 <>:14 165284145:160 1.242.131.0:2 165284212:163 165284145:164 :147 FALSE:97 CELLISTHEMED():128 165284195:160 150:98 175:98 207:98 RGB():128 165284211:163 165284195:164 :147 0:98 112:98 192:98 RGB():128 165284211:164 165284212:164 THEMEGUARD():128" />
14. </Formulas>
15. <FormulaReferences
16. Ref-1.1.3.0="1.242.131.0:2 1.5:32 <>:14 165284145:160 1.242.131.0:2 165284212:163 165284145:164 :147 FALSE:97 CELLISTHEMED():128 165284195:160 150:98 175:98 207:98 RGB():128 165284211:163 165284195:164 :147 0:98 112:98 192:98 RGB():128 165284211:164 165284212:164 THEMEGUARD():128"
17. Ref-1.1.3.1="1.242.131.0:2 1.5:32 <>:14 173519920:160 1.1.3.0:2 191:98 206:98 225:98 RGB():128 150:98 175:98 207:98 RGB():128 LUMDIFF():128 TINT():128 173519989:163 173519920:164 :147 FALSE:97 CELLISTHEMED():128 173519972:160 191:98 206:98 225:98 RGB():128 173519988:163 173519972:164 :147 255:98 60:98 60:98 RGB():128 173519988:164 173519989:164 THEMEGUARD():128"
18. Ref-1.242.131.0=""Chai":96 \_1,"ProductName",96,1:1 TRUE:97 3;STRSAME():129 173438756:160 255:98 0:98 0:98 RGB():128 173438833:163 173438756:164 :147 "Coffee":96 \_1,"ProductName",96,1:1 TRUE:97 3;STRSAME():129 173438821:160 255:98 255:98 0:98 RGB():128 173438832:163 173438821:164 :147 1.5:32 173438832:164 173438833:164" />
19. </Shape>
20. </Page>
21. </DataGraphics>

The following table provides more information about attribute values in the preceding sample [DataGraphic XML](#Section_23f5bf65461648cfbeda817f02b9d087).

| Attribute Name | Value | Notes |
| --- | --- | --- |
| ColorTable | 0,0,0 255,255,255 255,0,0 0,255,0 0,0,255 255,255,0 255,0,255 0,255,255 128,0,0 0,128,0 0,0,128 128,128,0 128,0,128 0,128,128 192,192,192 230,230,230 205,205,205 179,179,179 154,154,154 128,128,128 102,102,102 77,77,77 51,51,51 26,26,26 0,112,192 150,175,207 191,206,225 0,145,251 240,240,240 0,0,0 166,210,74 200,227,142 31,71,125 69,91,23 157,187,97 255,255,255 163,184,204 54,73,89 204,0,68 69,94,115 255,0,0 255,60,60 | The [color table](#Section_40036cbbe7fd48c5b634ca12f18a8adb) used to render [data graphics](#Section_037c6f2c08934c85aaf588143a8d9075) specified using the [**RGB**](#gt_2c716d3a-e60b-4e52-bbb0-2fdeb298003b) color model are (0,0,0), (255,255,255), (255,0,0), (0,255,0), (0,0,255), (255,255,0), (255,0,255), (0,255,255), (128,0,0), (0,128,0), (0,0,128), (128,128,0), (128,0,128), (0,128,128), (192,192,192), (230,230,230), (205,205,205), (179,179,179), (154,154,154), (128,128,128), (102,102,102), (77,77,77), (51,51,51), (26,26,26), (0,112,192), (150,175,207), (191,206,225), (0,145,251), (240,240,240), (0,0,0), (166,210,74), (200,227,142), (31,71,125), (69,91,23), (157,187,97), (255,255,255), (163,184,204), (54,73,89), (204,0,68), (69,94,115), (255,0,0), (255,60,60). |
| PageName | Page-1 | The name of the [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) is "Page-1". |
| ViewTransform | 96.000000 0.000000 0.000000 -96.000000 0.000000 1056.000000 | 96.000000 0.000000 0.000000 -96.000000 0.000000 1056.000000 represent a 3-by-3 affine transformation matrix as specified in [CT\_DataGraphics\_Page](#Section_1656fc08de3a4c088c7a5ed89568c0b6). |
| ID | {020D279F-6024-4139-8CDF-A7637F3217E7} | The identifier of the data graphic is {020D279F-6024-4139-8CDF-A7637F3217E7} |
| ShapeName | \_1 | The identifier of the [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) is "\_1". |
| DataRefs | 1,ProductName | The **BindingID** attribute of the [CT\_Binding](#Section_5ad7ddf78438400abe979591e0cca24a) element in [DataBinding XML Part](#Section_311514cd8ef444c9ab42562671155333) corresponding to the [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) external data [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) the shape is bound to is one. The **Name** attribute of the [CT\_DataColumn](#Section_f5259dad7ba3465092d7ef1d69cc6a70) element in [DataConnection XML Part](#Section_c489eefb7a014941bc37a4d332e66dc8) corresponding to the [**field**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc) in the recordset the text run is bound to is "ProductName". |
| Name | \_1 | The name of the corresponding [sheet element](#Section_94a21e97d2934aee9999f20030a1ee56) data graphic is "\_1". |
| FillBackground | 1.242.131.0:2 1.5:32 <>:14 173519920:160 1.1.3.0:2 191:98 206:98 225:98 RGB():128 150:98 175:98 207:98 RGB():128 LUMDIFF():128 TINT():128 173519989:163 173519920:164 :147 FALSE:97 CELLISTHEMED():128 173519972:160 191:98 206:98 225:98 RGB():128 173519988:163 173519972:164 :147 255:98 60:98 60:98 RGB():128 173519988:164 173519989:164 THEMEGUARD():128 | The [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc) used to recalculate the background color which is a part of the fill property of the corresponding sheet element data graphic is "1.242.131.0:2 1.5:32 <>:14 173519920:160 1.1.3.0:2 191:98 206:98 225:98 RGB():128 150:98 175:98 207:98 RGB():128 LUMDIFF():128 TINT():128 173519989:163 173519920:164 :147 FALSE:97 CELLISTHEMED():128 173519972:160 191:98 206:98 225:98 RGB():128 173519988:163 173519972:164 :147 255:98 60:98 60:98 RGB():128 173519988:164 173519989:164 THEMEGUARD():128". |
| FillForeground | 1.242.131.0:2 1.5:32 <>:14 165284145:160 1.242.131.0:2 165284212:163 165284145:164 :147 FALSE:97 CELLISTHEMED():128 165284195:160 150:98 175:98 207:98 RGB():128 165284211:163 165284195:164 :147 0:98 112:98 192:98 RGB():128 165284211:164 165284212:164 THEMEGUARD():128 | The formula expression used to recalculate the foreground color which is a part of the fill property of the corresponding sheet element data graphic is "1.242.131.0:2 1.5:32 <>:14 165284145:160 1.242.131.0:2 165284212:163 165284145:164 :147 FALSE:97 CELLISTHEMED():128 165284195:160 150:98 175:98 207:98 RGB():128 165284211:163 165284195:164 :147 0:98 112:98 192:98 RGB():128 165284211:164 165284212:164 THEMEGUARD():128". |
| Ref-1.1.3.0 | 1.242.131.0:2 1.5:32 <>:14 165284145:160 1.242.131.0:2 165284212:163 165284145:164 :147 FALSE:97 CELLISTHEMED():128 165284195:160 150:98 175:98 207:98 RGB():128 165284211:163 165284195:164 :147 0:98 112:98 192:98 RGB():128 165284211:164 165284212:164 THEMEGUARD():128 | "Ref-1.1.3.0" is the name of the formula reference for the formula expression "1.242.131.0:2 1.5:32 <>:14 165284145:160 1.242.131.0:2 165284212:163 165284145:164 :147 FALSE:97 CELLISTHEMED():128 165284195:160 150:98 175:98 207:98 RGB():128 165284211:163 165284195:164 :147 0:98 112:98 192:98 RGB():128 165284211:164 165284212:164 THEMEGUARD():128". |
| Ref-1.1.3.1 | 1.242.131.0:2 1.5:32 <>:14 173519920:160 1.1.3.0:2 191:98 206:98 225:98 RGB():128 150:98 175:98 207:98 RGB():128 LUMDIFF():128 TINT():128 173519989:163 173519920:164 :147 FALSE:97 CELLISTHEMED():128 173519972:160 191:98 206:98 225:98 RGB():128 173519988:163 173519972:164 :147 255:98 60:98 60:98 RGB():128 173519988:164 173519989:164 THEMEGUARD():128 | "Ref-1.1.3.1" is the name of the formula reference for the formula expression "1.242.131.0:2 1.5:32 <>:14 173519920:160 1.1.3.0:2 191:98 206:98 225:98 RGB():128 150:98 175:98 207:98 RGB():128 LUMDIFF():128 TINT():128 173519989:163 173519920:164 :147 FALSE:97 CELLISTHEMED():128 173519972:160 191:98 206:98 225:98 RGB():128 173519988:163 173519972:164 :147 255:98 60:98 60:98 RGB():128 173519988:164 173519989:164 THEMEGUARD():128". |
| Ref-1.242.131.0 | "Chai":96 \_1,"ProductName",96,1:1 TRUE:97 3;STRSAME():129 173438756:160 255:98 0:98 0:98 RGB():128 173438833:163 173438756:164 :147 "Coffee":96 \_1,"ProductName",96,1:1 TRUE:97 3;STRSAME():129 173438821:160 255:98 255:98 0:98 RGB():128 173438832:163 173438821:164 :147 1.5:32 173438832:164 173438833:164 | "Ref-1.242.131.0" is the name of the formula reference for the formula expression ""Chai":96 \_1,"ProductName",96,1:1 TRUE:97 3;STRSAME():129 173438756:160 255:98 0:98 0:98 RGB():128 173438833:163 173438756:164 :147 "Coffee":96 \_1,"ProductName",96,1:1 TRUE:97 3;STRSAME():129 173438821:160 255:98 255:98 0:98 RGB():128 173438832:163 173438821:164 :147 1.5:32 173438832:164 173438833:164". |

### ShapeInfo XML Part

1. <Page
2. Name="Page-1"
3. Zoom="0.500000"
4. OffsetX="407"
5. OffsetY="528"
6. DefaultUnitsText="50"
7. DefaultUnitsAngle="81"
8. DefaultUnitsDuration="44"
9. DefaultUnitsPage="73">
10. <Pages>
11. <PageInfo Name="Page-1" />
12. </Pages>
13. <ShapeInfo
14. Name="\_1"
15. DisplayName="Sheet.1"
16. Layout="{"type":"2D","bounds":{"x":23,"y":239,"width":727,"height":146}}">
17. <ShapeDataItems>
18. <ShapeData
19. Name="ProductID"
20. FormattedValue="1"
21. Value="1.000000"
22. Format="{0:0.####} {1}"
23. Type="2"
24. LangID="1033"
25. UnitLabel="0"
26. Unit="32"
27. BindingID="1" />
28. <ShapeData
29. Name="ProductName"
30. FormattedValue="Chai"
31. Value="Chai"
32. Format="{0}"
33. Type="0"
34. LangID="1033"
35. UnitLabel="0"
36. Unit="96"
37. BindingID="1" />
38. </ShapeDataItems>
39. </ShapeInfo>
40. </Page>

The following table provides more information about attribute values in the preceding sample [ShapeInfo XML](#Section_bf230dbc228e4108bd8f93c9cbd4fcac).

| Attribute Name | Value | Notes |
| --- | --- | --- |
| Name | Page-1 | Name of the [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640) is "Page-1". |
| Zoom | 0.500000 | The [**zoom level**](#gt_bfb33471-a018-422b-bc63-177c8bc1831f) at which the drawing page is initially displayed is 50%. |
| OffsetX | 407 | The x-coordinate in pixels of the point on the drawing page that is centered in the [**view**](#gt_3f793b0b-9509-4df0-89f9-92f07954beb8) when the drawing page is initially displayed is 407. The value is relative to the top, left corner of the drawing page and is expressed in the coordinate system of the view that displays the drawing page. |
| OffsetY | 528 | The y-coordinate in pixels of the point on the drawing page that is centered in the view when the drawing page is initially displayed is 528. The value is relative to the top, left corner of the drawing page and is expressed in the coordinate system of the view that displays the drawing page. |
| DefaultUnitsText | 50 | "Points" is used as the default typographic measurement unit in formula evaluations for this drawing page. |
| DefaultUnitsAngle | 81 | "Decimal degrees" is used as the default angle measurement unit in formula evaluations for this drawing page. |
| DefaultUnitsDuration | 44 | "Days" is used as the default duration measurement unit in formula evaluations for this drawing page. |
| DefaultUnitsPage | 73 | "Inches (fractional)" is used as the default unit for length measurement in formula evaluations for this drawing page, if none is explicitly specified in the formula expression. |
| Name | \_1 | The [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b) identifier is "\_1". |
| DisplayName | Sheet.1 | The shape name used for display purposes is "Sheet.1". |
| Layout | {"type":"2D","bounds":{"x":23,"y":239, "width":727, "height":146}} | 23 and 239 are the upper left x-coordinate and y-coordinate, respectively, of the [**bounding rectangle**](#gt_e349a155-0fb9-4ba1-abd4-e363e21abcd1) of the shape in [**pixels**](#gt_fe932605-d3bc-442f-9239-594676bce6ba). The coordinates are relative to the top, left corner of the drawing page that contains the shape. 727 and 146 are the width and height, respectively, of the bounding rectangle of the shape in pixels. |
| Name | ProductID | The label of the [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) item is "ProductID". |
| FormattedValue | 1 | The formatted value of the shape data item is 1. |
| Value | 1.000000 | The value of the shape data item is 1.000000. |
| Format | {0:0.####} {1}  {0} | "0.####" is the format string used to format the value in the first ShapeData element. A unit string will be displayed after the value if the value has a unit associated with it.  "{0}" is the format string used to format the value in the second ShapeData element. No formatting is required. The formatted value is the same as the value. |
| Type | 2 | The value is a numerical value. |
| LangID | 1033 | The [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) used to format the value is 1033, which maps to US English. |
| UnitLabel | 0 | No [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) with respect to format (abbreviated or full) and casing are applied to the unit label. |
| Unit | 32 | The [vUnitType](#Section_c9e68c3c1b3e4dd6af70f30cf7f15a0b) of the value is [ptgNum](#Section_d895c0dbaac64ac3b9390507f58282d0). |
| BindingID | 1 | The [CT\_Binding](#Section_5ad7ddf78438400abe979591e0cca24a) identifier for the shape data item is 1. |

### ShapeTextBinding XML Part

1. <Page Name="">
2. <ShapeText Name="\_3">
3. <TextRun
4. Name="txt\_3\_fld0"
5. Formula="\_1,&#34;ProductName&#34;,96,1:1 &#34;{0:0.####}@0@0@11@0@1033@0&#34;:96 :65 :252 1033:98 0:32 6;FORMATEX():129 "
6. Format="{0:0.####}@0@0@11@0@1033@0"
7. DataRefs="1,ProductName"/>
8. </ShapeText>
9. </Page>

The following table provides more information about attribute values in the preceding sample [ShapeTextBinding XML](#Section_2e21d026e11d46ee860a75c6b75bd84a).

| Attribute Name | Value | Notes |
| --- | --- | --- |
| Name | "" | Value is ignored |
| Name | \_3 | The identifier of the corresponding [text element](#Section_b061c2bde75d48edb51047479b51c6ec) data graphic is "\_3". |
| Name | txt\_3\_fld0 | The identifier of the corresponding text element data graphic is "txt\_3\_fld0". |
| Formula | \_1,&#34;ProductName&#34;,96,1:1 &#34;{0:0.####}@0@0@11@0@1033@0&#34;:96 :65 :252 1033:98 0:32 6;FORMATEX():129 | The [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc) used to recalculate the content of the [text run](#Section_8a573079d4b14123825dcb299f45d6b0) in the text element data graphic upon [**refresh**](#gt_d947dfed-8e3a-4451-b978-d69a89ae6060) is "\_1,&#34;ProductName&#34;,96,1:1 &#34;{0:0.####}@0@0@11@0@1033@0&#34;:96 :65 :252 1033:98 0:32 6;FORMATEX():129". |
| Format | {0:0.####}@0@0@11@0@1033@0 | The format string used to format the result of the formula evaluation is "{0:0.####}". No custom formatting needs to be applied to string. No [**rules**](#gt_b4fb40b2-72f2-4fd8-875b-277270553c4f) with respect to form (abbreviated or full) and casing are applied to the unit label. The [**LCID**](#gt_c7f99c66-592f-4053-b62a-878c189653b6) used to format the value is 1033, which maps to US English. |
| DataRefs | 1,ProductName | The BindingID attribute of the [CT\_Binding](#Section_5ad7ddf78438400abe979591e0cca24a) element in [DataBinding XML Part](#Section_311514cd8ef444c9ab42562671155333) corresponding to the external data [**row**](#gt_a87817fc-9b18-49a1-925e-9be9e1d92665) the text run is bound to is one. The Name attribute of the [CT\_DataColumn](#Section_f5259dad7ba3465092d7ef1d69cc6a70) element in [DataConnection XML Part](#Section_c489eefb7a014941bc37a4d332e66dc8) corresponding to the [**field**](#gt_f819dd42-7f44-4613-8231-d5ad47f2bbcc) in the [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) the text run is bound to is "ProductName". |

### XAML

1. <Canvas
2. Height="1056"
3. Name="Workspace" Width="816"
4. xmlns="http://schemas.microsoft.com/winfx/2006/xaml/presentation">
5. <Canvas Name="D">
6. <Canvas.Background>
7. <SolidColorBrush Color="#FFFFFFFF"/>
8. </Canvas.Background>
9. <Canvas
10. Tag="Shape"
11. Name="\_1"
12. RenderTransform="1, 0, 0, 1, 24, -672">
13. <Rectangle
14. Canvas.Left="-0.5"
15. Canvas.Top="911.5"
16. Height="145"
17. Name="\_1\_G0\_P0"
18. Width="361"
19. StrokeThickness="1"
20. StrokeDashCap="Round"
21. StrokeEndLineCap="Round"
22. StrokeLineJoin="Round"
23. StrokeStartLineCap="Round">
24. <Rectangle.Stroke>
25. <SolidColorBrush Color="#FF000000"/>
26. </Rectangle.Stroke>
27. <Rectangle.Fill>
28. <SolidColorBrush Color="#FFFF0000"/>
29. </Rectangle.Fill>
30. </Rectangle>
31. <Canvas Name="\_2">
32. <Canvas.RenderTransform>
33. <MatrixTransform Matrix="1, 0, 0, 1, 366, -57.544364"/>
34. </Canvas.RenderTransform>
35. <Path
36. Name="Path1"
37. Data="M1.92 1054.08 L358.08 1054.08 L358.08 1029.01 L1.92 1029.01 L1.92 1054.08 Z"/>
38. <Path
39. Name="Path2"
40. Data="M1.92 1054.08 L358.08 1054.08"/>
41. <Path
42. Name="Path3"
43. StrokeThickness="0.5"
44. StrokeDashCap="Round"
45. StrokeEndLineCap="Round"
46. StrokeLineJoin="Round"
47. StrokeStartLineCap="Round"
48. Data="M1.92 1054.08 L358.08 1054.08 L358.08 1029.01 L1.92 1029.01 L1.92 1054.08">
49. <Path.Stroke>
50. <SolidColorBrush Opacity="0" Color="#FF000000"/>
51. </Path.Stroke>
52. </Path>
53. <Path
54. Name="Path4"
55. StrokeThickness="0.5"
56. StrokeDashCap="Round"
57. StrokeEndLineCap="Round"
58. StrokeLineJoin="Round"
59. StrokeStartLineCap="Round"
60. Data="M1.92 1054.08 L358.08 1054.08">
61. <Path.Stroke>
62. <SolidColorBrush Color="#FF000000"/>
63. </Path.Stroke>
64. </Path>
65. <Canvas>
66. <Glyphs
67. FontRenderingEmSize="18.667"
68. FontUri="CommonFonts/4C1AABE5-BA71-4F2C-8263-66731F995C4B.odttf"
69. OriginX="3.2534"
70. OriginY="1047.1"
71. UnicodeString="ProductName">
72. <Glyphs.Fill>
73. <SolidColorBrush Color="#FF000000"/>
74. </Glyphs.Fill>
75. </Glyphs>
76. </Canvas>
77. <Canvas Name="\_3">
78. <TextBlock
79. Tag="Right;Center;;1.2;0.013888888888889;0.013888888888889;0.013888888888889;0.013888888888889"
80. Height="0.26116"
81. Visibility="Collapsed"
82. Width="2.545"
83. RenderTransform="96, 0, 0, 96, 113.756384, 1029.008729"
84. TextWrapping="Wrap">
85. <Run
86. Name="txt\_3\_fld0"
87. FontFamily="Calibri"
88. FontSize="0.19444"
89. Foreground="#FF000000">
90. Chai
91. </Run>
92. </TextBlock>
93. <Canvas>
94. <Glyphs
95. FontRenderingEmSize="18.667"
96. FontUri="CommonFonts/4C1AABE5-BA71-4F2C-8263-66731F995C4B.odttf"
97. OriginX="323.76"
98. OriginY="1047.1"
99. UnicodeString="Chai">
100. <Glyphs.Fill>
101. <SolidColorBrush Color="#FF000000"/>
102. </Glyphs.Fill>
103. </Glyphs>
104. </Canvas>
105. </Canvas>
106. </Canvas>
107. </Canvas>
108. </Canvas>
109. </Canvas>

The following table provides more information about elements in the preceding sample [XAML](#Section_d86834c499d248e2b13ab2c8f016eca0).

| Node | Notes |
| --- | --- |
| Topmost canvas node with name "Workspace" | Canvas for the container of the whole [drawing page](#Section_ebba8748c5a54ac7a4191c6a70307640). 816 and 1056 are the width and height, respectively, at 100% zoom in [**pixels**](#gt_fe932605-d3bc-442f-9239-594676bce6ba). |
| Canvas node with name "D" | Canvas for the drawing page. |
| Canvas node with name "\_1" | Canvas for the rectangle [shape](#Section_6f93675ffa84476c92dd9d3f32d9b22b). Identifier of the shape is "\_1". |
| Rectangle node | Graphical representation of the rectangle shape. |
| Rectangle.Stroke node | The line color of the shape is black. |
| Rectangle.Fill node | The fill color of the shape is red. |
| Canvas node with name "\_2" | Canvas for text callout shape. The identifier of the shape is "\_2". |
| Glyphs node | Representation of the text in the shape. |
| FontRenderingEmSize attribute | Pixel size of text is 18.667. |
| FontUri attribute | [**Font**](#gt_f8aa3f46-99d1-49bb-858f-b4bfa546c1c2) of the text is Calibri. |
| UnicodeString attribute | The text in the shape is "Sample Text". |
| Glyphs.Fill node | The color of the text is black. |
| Canvas node with name "\_3" | Canvas for shape with identifier "\_3". |
| TextBlock Node within Canvas node with name "\_3" | Contains high level information to be able to compose text after it has been refreshed. |
| Run node with name "txt\_3\_fld0" | Maps to corresponding TextRun node with name "txt\_3\_fld0" in the [ShapeTextBinding XML Part](#Section_2e21d026e11d46ee860a75c6b75bd84a). |

## Formula Evaluation

This example describes how a [formula expression](#Section_1639b4e843e14d409531adf5c2fe8bbc) is parsed from the file and evaluated. This formula expression returns a [PtgColorRGB](#Section_824daa8a2c0e4a46912624580eab2882) based on the value obtained for the data bound [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) item named "ProductName". If the value is "Chai" the color returned is 0xFFFF0000 (red). If the value is "Coffee" the color returned is 0xFFFFFF00 (yellow).

The formula expression is persisted in the [DataGraphic XML Part](#Section_23f5bf65461648cfbeda817f02b9d087) as follows.

1. "Chai":96 \_1,"ProductName",96,1:1 TRUE:97 3;STRSAME():129 173438756:160 255:98 0:98 0:98 RGB():128 173438833:163 173438756:164 :147 "Coffee":96 \_1,"ProductName",96,1:1 TRUE:97 3;STRSAME():129 173438821:160 255:98 255:98 0:98 RGB():128 173438832:163 173438821:164 :147 1.5:32 173438832:164 173438833:164

The following table describes the [parse tokens](#Section_5ae5d80a9aaf40948a2d92cf8e499589) in the formula expression and the values of the tokens on the [evaluation stack](#Section_f70486783f2d445fae706da7e34fe640) during [formula evaluation](#Section_81f0fbe77a314d34b52f5fa9c9380c33).

| File Representation of Parse Token | Token Parsing Operation | Evaluation Stack State |
| --- | --- | --- |
| "Chai":96 | This is a [PtgStr1](#Section_acd019e2c6c245e68cbd229441afdb69), an [operand token](#Section_0706e8d56456401c85fab5f6351a4a6e). The value of the token is pushed onto the stack. | "Chai" |
| \_1,"ProductName",96,1:1 | This is a [PtgDataBinding](#Section_7442cb26af7f46ce9bd348656835396c), an operand token. The value of the cell in the [recordset](#Section_d7cde52f3fed4fd6a692ad6ecb672ed2) referenced by the [data binding](#Section_b9b194892da24d77a8a51c99bb866fe1) is a PtgStr1. It is pushed onto the stack. | "Coffee"  "Chai" |
| TRUE:97 | This is a [PtgBool](#Section_f4171c65cc2f43458b8b16adb7361508), an operand token. A [**Boolean**](#gt_1d79d7a7-ba2c-4b34-931c-7ba8057c87b2) value of TRUE is pushed onto the stack. | TRUE  "Coffee"  "Chai" |
| 3;STRSAME():129 | This is a [PtgFuncVar](#Section_e483d78ecd494ba7aa042f843a09f8df), a [function token](#Section_e78706d7cd8c424bbc3bfc74904871ca) for a function that takes three arguments. The top three tokens are popped off the stack and evaluated by the [StrSame](#Section_d78370656e7e4d50ac6258dd77c7b7cf) function. The resulting value of FALSE is pushed onto the stack. | FALSE |
| 173438756:160 | This is a [PtgJmpF](#Section_81250f6f47d04c3d93cd60e2d8798784), a [control token](#Section_41b89226e2a7471fb70c6d01003be607). It compares the value of the token at the top of the stack to FALSE.  The comparison succeeds and all subsequent tokens are parsed and ignored until a [PtgJmpLabel](#Section_5f545b4b47f44ed38932659537323ff9) token with a value of 173438756 is reached. | FALSE |
| 255:98 | This is a [PtgUnsWord](#Section_9651517171494cd9a74403c2fcca6d5b) that is parsed but ignored. | FALSE |
| 0:98 | This is a PtgUnsWord that is parsed but ignored. | FALSE |
| 0:98 | This is a PtgUnsWord that is parsed but ignored. | FALSE |
| RGB():128 | This is a [PtgFunc](#Section_8206928cdaaf400daa071a566fe1c88b) that is parsed but ignored. | FALSE |
| 173438833:163 | This is a [PtgJmp](#Section_a2fcbad2d4cd4d0bb4338ec960e51755) that is parsed but ignored. | FALSE |
| 173438756:164 | This is a PtgJmpLabel, a control token. Evaluation resumes at this position. Nothing is pushed on to the stack. | FALSE |
| :147 | This is a [PtgPop](#Section_a5016a0a9e794b6dbd38231239a97710), a control token that pops the top token off the stack. |  |
| "Coffee":96 | This is a PtgStr1, an operand token. The value of the token is pushed onto the stack. | "Coffee" |
| \_1,"ProductName",96,1:1 | This is a PtgDataBinding, an operand token. The value of the cell in the recordset referenced by the data binding is pushed onto the stack. | "Coffee"  "Coffee" |
| TRUE:97 | This is a PtgBool, an operand token. A Boolean value of TRUE is pushed onto the stack. | TRUE  "Coffee"  "Coffee" |
| 3;STRSAME():129 | This is a PtgFuncVar, a function token for a function that takes three arguments. The top three tokens are popped off the stack and evaluated by the StrSame function. The resulting value of TRUE is pushed onto the stack. | TRUE |
| 173438821:160 | This is a PtgJmpF, a control token. It compares the value of the token at the top of the stack to FALSE.  The comparison fails and the top token is popped off the stack. |  |
| 255:98 | This is a PtgUnsWord, an operand token. The numeric value 255 is pushed onto the stack. | 255 |
| 255:98 | This is a PtgUnsWord, an operand token. The numeric value 255 is pushed onto the stack. | 255  255 |
| 0:98 | This is a PtgUnsWord, an operand token. The numeric value 0 is pushed onto the stack. | 0  255  255 |
| RGB():128 | This is a PtgFunc, a function token for a function that takes three arguments. The top three tokens are popped off the stack and evaluated by the [RGB](#Section_db09d78046eb4ad19dc749ebc187a383) function. The function returns a PtgColorRGB with a value of 0xFFFFFF00. The corresponding decimal value of -256 is pushed onto the stack. | -256 |
| 173438832:163 | This is a PtgJmp, a control token. All subsequent tokens are parsed and ignored until a PtgJmpLabel token with a value of 173438832 is reached. | -256 |
| 173438821:164 | This is a PtgJmpLabel that is parsed but ignored. | -256 |
| :147 | This is a PtgPop that is parsed but ignored. | -256 |
| 1.5:32 | This is a [PtgNum](#Section_d895c0dbaac64ac3b9390507f58282d0) that is parsed but ignored. | -256 |
| 173438832:164 | This is a PtgJmpLabel, a control token. Evaluation resumes at this position. Nothing is pushed on to the stack. | -256 |
| 173438833:164 | This is a PtgJmpLabel, a control token. It is parsed but ignored because another control token is not being evaluated. | -256 |

At this point all the parse tokens in the formula expression have been parsed. The single token that is left on the stack is a PtgColorRGB with a value of -256, or 0xFFFFFF00 (yellow). This token is popped off the evaluation stack and returned as the result of the formula evaluation. The formula evaluation is now complete.

# Security

## Security Considerations for Implementers

Implementers or consumers of this file format need to take into consideration that these files contain sensitive information. Implementers are encouraged to treat these files as sensitive resources and protect them appropriately. These files often contain the following:

* Internal information such as server names, table names, and [**list**](#gt_04ce231e-214c-44fd-b7ba-7cc19eee79bf) names can be embedded in the [**connection strings**](#gt_03a9d0ca-2f10-4f3d-b910-052714a96f7d) and [**query**](#gt_37fbc661-f744-48fa-9d8e-f34513cab9c2) strings of the [data connection](#Section_28d3d364de644cada48144231aed4d7b) elements in the file. However, usernames and passwords are never stored in a VDW file.
* The location of [**Office data connection (ODC) files**](#gt_e6fb952a-b6b6-40af-ad1b-ea6cf97da509), which contain additional sensitive information.

Data from external [**data sources**](#gt_e091613c-6901-4874-b9b2-27273ead1075) are linked to [shapes](#Section_6f93675ffa84476c92dd9d3f32d9b22b) and stored in [shape data](#Section_2c2d05f1d69d45f8bfc6ace1d0bc6374) elements. The data is queried with the credentials of the user making the request and can return sensitive data that is specific to the user. This data is stored in the VDW file when the file is created.

## Index of Security Fields

None.

# Appendix A: Full XML Schemas

For ease of implementation, this section provides the full XML schemas for the portions of the document described in [XML Part Structure](#Section_e353d699f4bc454eaf7b20193355ee32).

## app Schema

1. <xsd:schema id="app" targetNamespace="http://schemas.openxmlformats.org/officeDocument/2006/extended-properties" attributeFormDefault="qualified" elementFormDefault="qualified">
2. <xsd:complexType name="CT\_PageMetaData">
3. <xsd:attribute name="ID" type="xsd:unsignedLong" use="required" form="unqualified"/>
4. <xsd:attribute name="Name" type="xsd:string" use="required" form="unqualified"/>
5. </xsd:complexType>
6. <xsd:complexType name="CT\_PagesMetaData">
7. <xsd:sequence>
8. <xsd:element name="PageMetaData" minOccurs="1" maxOccurs="unbounded" type="CT\_PageMetaData"/>
9. </xsd:sequence>
10. </xsd:complexType>
11. <xsd:complexType name="CT\_Properties">
12. <xsd:sequence>
13. <xsd:element name="Application" type="xsd:string" minOccurs="1" maxOccurs="1"/>
14. <xsd:element name="Company" type="xsd:string" minOccurs="1" maxOccurs="1"/>
15. <xsd:element name="AppVersion" type="xsd:string" minOccurs="1" maxOccurs="1"/>
16. <xsd:element name="DocumentVersion" type="xsd:string" minOccurs="1" maxOccurs="1"/>
17. <xsd:element name="DocumentLanguage" type="xsd:integer" minOccurs="1" maxOccurs="1"/>
18. <xsd:element name="DocumentMeasurementSystem" type="xsd:string" minOccurs="1" maxOccurs="1"/>
19. <xsd:element name="DocumentCurrencyID" type="xsd:integer" minOccurs="1" maxOccurs="1"/>
20. <xsd:element name="PagesMetaData" type="CT\_PagesMetaData" minOccurs="1" maxOccurs="1"/>
21. </xsd:sequence>
22. </xsd:complexType>
23. <xsd:element name="Properties" type="CT\_Properties"/>
24. </xsd:schema>

## core Schema

1. <xsd:schema id="core" targetNamespace="http://schemas.openxmlformats.org/package/2006/metadata/core-properties" xmlns="http://schemas.openxmlformats.org/package/2006/metadata/core-properties" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:msdata="urn:schemas-microsoft-com:xml-msdata" attributeFormDefault="qualified" elementFormDefault="qualified"
2. >
3. <xsd:complexType name="CT\_coreProperties">
4. <xsd:sequence>
5. <xsd:element name="description" msdata:Prefix="dc" type="xsd:string" minOccurs="0" maxOccurs="1"/>
6. </xsd:sequence>
7. </xsd:complexType>
8. <xsd:element name="coreProperties" msdata:Prefix="cp" type="CT\_coreProperties"></xsd:element>
9. </xsd:schema>

## DataBinding Schema

1. <xsd:schema id="DataBinding">
2. <xsd:complexType name="CT\_PrimaryKeyValue">
3. <xsd:attribute name="Value" type="xsd:string"/>
4. </xsd:complexType>
5. <xsd:complexType name="CT\_PrimaryKeyValues">
6. <xsd:sequence>
7. <xsd:element name="PrimaryKey" minOccurs="1" maxOccurs="unbounded" type="CT\_PrimaryKeyValue"/>
8. </xsd:sequence>
9. </xsd:complexType>
10. <xsd:complexType name="CT\_Binding">
11. <xsd:sequence>
12. <xsd:element name="PrimaryKeys" minOccurs="0" maxOccurs="1" type="CT\_PrimaryKeyValues"/>
13. </xsd:sequence>
14. <xsd:attribute name="BindingID" type="xsd:integer"/>
15. <xsd:attribute name="Row" type="xsd:integer" use="optional"/>
16. </xsd:complexType>
17. <xsd:complexType name="CT\_Bindings">
18. <xsd:sequence>
19. <xsd:element name="Binding" minOccurs="1" maxOccurs="unbounded" type="CT\_Binding"/>
20. </xsd:sequence>
21. </xsd:complexType>
22. <xsd:complexType name="CT\_BindingConnection">
23. <xsd:sequence>
24. <xsd:element name="Bindings" minOccurs="1" maxOccurs="1" type="CT\_Bindings"/>
25. </xsd:sequence>
26. <xsd:attribute name="RecordsetID" type="xsd:unsignedLong"/>
27. </xsd:complexType>
28. <xsd:complexType name="CT\_BindingConnections">
29. <xsd:sequence>
30. <xsd:element name="BindingConnection" minOccurs="0" maxOccurs="unbounded" type="CT\_BindingConnection"/>
31. </xsd:sequence>
32. </xsd:complexType>
33. <xsd:element name="BindingConnections" type="CT\_BindingConnections"/>
34. </xsd:schema>

## DataConnection Schema

1. <xsd:schema id="DataConnection">
2. <xsd:complexType name="CT\_DataConnection">
3. <xsd:attribute name="ID" type="xsd:unsignedLong" use="required"/>
4. <xsd:attribute name="ConnectionString" type="xsd:string" use="optional"/>
5. <xsd:attribute name="Filename" type="xsd:string" use="optional"/>
6. </xsd:complexType>
7. <xsd:complexType name="CT\_DataConnections">
8. <xsd:sequence>
9. <xsd:element name="DataConnection" minOccurs="0" maxOccurs="unbounded" type="CT\_DataConnection"/>
10. </xsd:sequence>
11. </xsd:complexType>
12. <xsd:complexType name="CT\_PrimaryKey">
13. <xsd:attribute name="Key" type="xsd:string" use="required"/>
14. </xsd:complexType>
15. <xsd:complexType name="CT\_PrimaryKeys">
16. <xsd:sequence>
17. <xsd:element name="PrimaryKey" minOccurs="1" maxOccurs="unbounded" type="CT\_PrimaryKey"/>
18. </xsd:sequence>
19. </xsd:complexType>
20. <xsd:complexType name="CT\_DataColumn">
21. <xsd:attribute name="Name" type="xsd:string" use="required"/>
22. <xsd:attribute name="DisplayName" type="xsd:string" use="required"/>
23. <xsd:attribute name="Type" type="xsd:integer" use="required"/>
24. </xsd:complexType>
25. <xsd:complexType name="CT\_DataColumns">
26. <xsd:sequence>
27. <xsd:element name="DataColumn" minOccurs="1" maxOccurs="unbounded" type="CT\_DataColumn"/>
28. </xsd:sequence>
29. </xsd:complexType>
30. <xsd:complexType name="CT\_DataRecordset">
31. <xsd:sequence>
32. <xsd:element name="PrimaryKeys" minOccurs="0" maxOccurs="1" type="CT\_PrimaryKeys"/>
33. <xsd:element name="DataColumns" minOccurs="1" maxOccurs="1" type="CT\_DataColumns"/>
34. </xsd:sequence>
35. <xsd:attribute name="ID" type="xsd:unsignedLong" use="required"/>
36. <xsd:attribute name="ConnectionID" type="xsd:unsignedLong" use="required"/>
37. <xsd:attribute name="Command" type="xsd:string" use="required"/>
38. <xsd:attribute name="Pages" type="xsd:string" use="required"/>
39. <xsd:attribute name="RowOrder" type="xsd:integer" use="optional"/>
40. </xsd:complexType>
41. <xsd:complexType name="CT\_DataRecordsets">
42. <xsd:sequence>
43. <xsd:element name="DataRecordset" minOccurs="0" maxOccurs="unbounded" type="CT\_DataRecordset"/>
44. </xsd:sequence>
45. </xsd:complexType>
46. <xsd:complexType name="CT\_Connections">
47. <xsd:sequence>
48. <xsd:element name="DataConnections" minOccurs="1" maxOccurs="1" type="CT\_DataConnections"/>
49. <xsd:element name="DataRecordsets" minOccurs="1" maxOccurs="1" type="CT\_DataRecordsets"/>
50. </xsd:sequence>
51. </xsd:complexType>
52. <xsd:element name="Connections" type="CT\_Connections"/>
53. </xsd:schema>

## DataGraphicDefinition Schema

1. <xsd:schema id="DataGraphicDefinition">
2. <xsd:complexType name="CT\_IconSetRule">
3. <xsd:attribute name="Name" type="xsd:string" use="required"/>
4. <xsd:attribute name="Formula" type="xsd:string" use="required"/>
5. <xsd:attribute name="Image" type="xsd:string" use="required"/>
6. </xsd:complexType>
7. <xsd:complexType name="CT\_IconSetDef">
8. <xsd:sequence>
9. <xsd:element name="IconSetRule" minOccurs="1" maxOccurs="unbounded" type="CT\_IconSetRule"/>
10. </xsd:sequence>
11. <xsd:attribute name="DefID" type="xsd:string" use="required"/>
12. </xsd:complexType>
13. <xsd:complexType name="CT\_DataGraphicDefs">
14. <xsd:sequence>
15. <xsd:element name="IconSetDef" minOccurs="0" maxOccurs="unbounded" type="CT\_IconSetDef"/>
16. </xsd:sequence>
17. </xsd:complexType>
18. <xsd:element name="DataGraphicDefs" type="CT\_DataGraphicDefs"/>
19. </xsd:schema>

## DataGraphic Schema

1. <xsd:schema id="DataGraphic">
2. <xsd:complexType name="CT\_DataGraphicDef">
3. <xsd:attribute name="DefID" type="xsd:string" use="required"/>
4. <xsd:attribute name="DefShapeID" type="xsd:string" use="required"/>
5. </xsd:complexType>
6. <xsd:complexType name="CT\_DataGraphicDefinitions">
7. <xsd:sequence>
8. <xsd:element name="DataGraphicDef" minOccurs="1" maxOccurs="unbounded" type="CT\_DataGraphicDef"/>
9. </xsd:sequence>
10. </xsd:complexType>
11. <xsd:complexType name="CT\_Geometry">
12. <xsd:attribute name="Index" type="xsd:integer" use="required"/>
13. <xsd:attribute name="BoundPts" type="xsd:string" use="optional"/>
14. <xsd:attribute name="Attribute" type="xsd:string" use="required"/>
15. <xsd:attribute name="Formula" type="xsd:string" use="required"/>
16. </xsd:complexType>
17. <xsd:complexType name="CT\_Sheet">
18. <xsd:sequence>
19. <xsd:element name="Geometry" minOccurs="0" maxOccurs="unbounded" type="CT\_Geometry"/>
20. </xsd:sequence>
21. <xsd:attribute name="Name" type="xsd:string"/>
22. <xsd:attribute name="Angle" type="xsd:string" use="optional"/>
23. <xsd:attribute name="FlipX" type="xsd:string" use="optional"/>
24. <xsd:attribute name="FlipY" type="xsd:string" use="optional"/>
25. <xsd:attribute name="LocPinX" type="xsd:string" use="optional"/>
26. <xsd:attribute name="LocPinY" type="xsd:string" use="optional"/>
27. <xsd:attribute name="PinX" type="xsd:string" use="optional"/>
28. <xsd:attribute name="PinY" type="xsd:string" use="optional"/>
29. <xsd:attribute name="HideText" type="xsd:string" use="optional"/>
30. <xsd:attribute name="FillBackground" type="xsd:string" use="optional"/>
31. <xsd:attribute name="FillBackgroundTrans" type="xsd:string" use="optional"/>
32. <xsd:attribute name="FillForeground" type="xsd:string" use="optional"/>
33. <xsd:attribute name="FillForegroundTrans" type="xsd:string" use="optional"/>
34. <xsd:attribute name="FillPattern" type="xsd:string" use="optional"/>
35. <xsd:attribute name="LineColor" type="xsd:string" use="optional"/>
36. <xsd:attribute name="LineColorTrans" type="xsd:string" use="optional"/>
37. <xsd:attribute name="LineWeight" type="xsd:string" use="optional"/>
38. <xsd:attribute name="TextWidth" type="xsd:string" use="optional"/>
39. <xsd:attribute name="TextHeight" type="xsd:string" use="optional"/>
40. <xsd:attribute name="TextAngle" type="xsd:string" use="optional"/>
41. <xsd:attribute name="TextLocPinX" type="xsd:string" use="optional"/>
42. <xsd:attribute name="TextLocPinY" type="xsd:string" use="optional"/>
43. <xsd:attribute name="TextPinX" type="xsd:string" use="optional"/>
44. <xsd:attribute name="TextPinY" type="xsd:string" use="optional"/>
45. <xsd:attribute name="Rounding" type="xsd:string" use="optional"/>
46. </xsd:complexType>
47. <xsd:complexType name="CT\_Formulas">
48. <xsd:sequence>
49. <xsd:element name="Sheet" minOccurs="1" maxOccurs="unbounded" type="CT\_Sheet"/>
50. </xsd:sequence>
51. </xsd:complexType>
52. <xsd:complexType name="CT\_FormulaReferences">
53. <xsd:anyAttribute namespace="##local" processContents="skip"/>
54. </xsd:complexType>
55. <xsd:complexType name="CT\_DataGraphics\_Shape">
56. <xsd:sequence>
57. <xsd:element name="DataGraphicDefinitions" minOccurs="0" maxOccurs="1" type="CT\_DataGraphicDefinitions"/>
58. <xsd:element name="Formulas" minOccurs="0" maxOccurs="1" type="CT\_Formulas"/>
59. <xsd:element name="FormulaReferences" minOccurs="0" maxOccurs="1" type="CT\_FormulaReferences"/>
60. </xsd:sequence>
61. <xsd:attribute name="ID" type="xsd:string" use="required"/>
62. <xsd:attribute name="ShapeName" type="xsd:string" use="required"/>
63. <xsd:attribute name="DataRefs" type="xsd:string" use="optional"/>
64. </xsd:complexType>
65. <xsd:complexType name="CT\_DataGraphics\_Page">
66. <xsd:sequence>
67. <xsd:element name="Shape" minOccurs="0" maxOccurs="unbounded" type="CT\_DataGraphics\_Shape"/>
68. </xsd:sequence>
69. <xsd:attribute name="PageName" type="xsd:string"/>
70. <xsd:attribute name="ViewTransform" type="xsd:string"/>
71. </xsd:complexType>
72. <xsd:complexType name="CT\_DataGraphics">
73. <xsd:sequence>
74. <xsd:element name="Page" minOccurs="1" maxOccurs="unbounded" type="CT\_DataGraphics\_Page"/>
75. </xsd:sequence>
76. <xsd:attribute name="ColorTable" type="xsd:string"/>
77. </xsd:complexType>
78. <xsd:element name="DataGraphics" type="CT\_DataGraphics"/>
79. </xsd:schema>

## ShapeInfo Schema

1. <xsd:schema id="ShapeInfo">
2. <xsd:complexType name="CT\_PageInfo">
3. <xsd:attribute name="Name" type="xsd:string"/>
4. </xsd:complexType>
5. <xsd:complexType name="CT\_Pages">
6. <xsd:sequence>
7. <xsd:element name="PageInfo" minOccurs="1" maxOccurs="unbounded" type="CT\_PageInfo"/>
8. </xsd:sequence>
9. </xsd:complexType>
10. <xsd:complexType name="CT\_ShapeData">
11. <xsd:attribute name="Name" type="xsd:string" use="required"/>
12. <xsd:attribute name="FormattedValue" type="xsd:string" use="required"/>
13. <xsd:attribute name="Value" type="xsd:string" use="required"/>
14. <xsd:attribute name="Format" type="xsd:string" use="required"/>
15. <xsd:attribute name="Type" use="required">
16. <xsd:simpleType>
17. <xsd:restriction base="xsd:integer">
18. <xsd:enumeration value="0"/>
19. <xsd:enumeration value="1"/>
20. <xsd:enumeration value="2"/>
21. <xsd:enumeration value="3"/>
22. <xsd:enumeration value="4"/>
23. <xsd:enumeration value="5"/>
24. <xsd:enumeration value="6"/>
25. <xsd:enumeration value="7"/>
26. </xsd:restriction>
27. </xsd:simpleType>
28. </xsd:attribute>
29. <xsd:attribute name="LangID" type="xsd:integer" use="required"/>
30. <xsd:attribute name="UnitLabel" type="xsd:integer" use="required"/>
31. <xsd:attribute name="CalendarID" type="xsd:integer" use="optional"/>
32. <xsd:attribute name="CurrencyID" type="xsd:integer" use="optional"/>
33. <xsd:attribute name="ServerAction" type="xsd:integer" use="optional"/>
34. <xsd:attribute name="Unit" type="xsd:integer" use="required"/>
35. <xsd:attribute name="DisplayUnit" type="xsd:integer" use="optional"/>
36. <xsd:attribute name="BindingID" type="xsd:integer" use="optional"/>
37. <xsd:attribute name="HyperlinkID" type="xsd:string" use="optional"/>
38. </xsd:complexType>
39. <xsd:complexType name="CT\_ShapeDataItems">
40. <xsd:sequence>
41. <xsd:element name="ShapeData" minOccurs="1" maxOccurs="unbounded" type="CT\_ShapeData"/>
42. </xsd:sequence>
43. </xsd:complexType>
44. <xsd:complexType name="CT\_Hyperlink">
45. <xsd:attribute name="Name" type="xsd:string" use="required"/>
46. <xsd:attribute name="Value" type="xsd:string" use="optional"/>
47. <xsd:attribute name="Description" type="xsd:string" use="optional"/>
48. <xsd:attribute name="SubAddress" type="xsd:string" use="optional"/>
49. <xsd:attribute name="SubAddressShape" type="xsd:string" use="optional"/>
50. <xsd:attribute name="Zoom" use="optional">
51. <xsd:simpleType>
52. <xsd:union>
53. <xsd:simpleType>
54. <xsd:restriction base="xsd:integer">
55. <xsd:minExclusive value="0"/>
56. </xsd:restriction>
57. </xsd:simpleType>
58. <xsd:simpleType>
59. <xsd:restriction base="xsd:integer">
60. <xsd:enumeration value="-2"/>
61. <xsd:enumeration value="-1"/>
62. </xsd:restriction>
63. </xsd:simpleType>
64. </xsd:union>
65. </xsd:simpleType>
66. </xsd:attribute>
67. <xsd:attribute name="Default" type="xsd:boolean" use="optional"/>
68. </xsd:complexType>
69. <xsd:complexType name="CT\_Hyperlinks">
70. <xsd:sequence>
71. <xsd:element name="Hyperlink" minOccurs="1" maxOccurs="unbounded" type="CT\_Hyperlink"/>
72. </xsd:sequence>
73. </xsd:complexType>
74. <xsd:complexType name="CT\_ShapeInfo">
75. <xsd:sequence>
76. <xsd:element name="ShapeDataItems" minOccurs="0" maxOccurs="1" type="CT\_ShapeDataItems"/>
77. <xsd:element name="Hyperlinks" minOccurs="0" maxOccurs="1" type="CT\_Hyperlinks"/>
78. </xsd:sequence>
79. <xsd:attribute name="Name" type="xsd:string" use="required"/>
80. <xsd:attribute name="DisplayName" type="xsd:string"/>
81. <xsd:attribute name="Guid" type="xsd:string" use="optional"/>
82. <xsd:attribute name="Layout" type="xsd:string" use="required"/>
83. </xsd:complexType>
84. <xsd:complexType name="CT\_Page">
85. <xsd:sequence>
86. <xsd:element name="Pages" minOccurs="1" maxOccurs="1" type="CT\_Pages"/>
87. <xsd:element name="ShapeInfo" minOccurs="0" maxOccurs="unbounded" type="CT\_ShapeInfo"/>
88. </xsd:sequence>
89. <xsd:attribute name="Name" type="xsd:string" use="required"/>
90. <xsd:attribute name="Zoom" use="required">
91. <xsd:simpleType>
92. <xsd:union>
93. <xsd:simpleType>
94. <xsd:restriction base="xsd:double">
95. <xsd:enumeration value="-2"/>
96. <xsd:enumeration value="-1"/>
97. </xsd:restriction>
98. </xsd:simpleType>
99. <xsd:simpleType>
100. <xsd:restriction base="xsd:double">
101. <xsd:minInclusive value="0"/>
102. </xsd:restriction>
103. </xsd:simpleType>
104. </xsd:union>
105. </xsd:simpleType>
106. </xsd:attribute>
107. <xsd:attribute name="OffsetX" type="xsd:integer" use="required"/>
108. <xsd:attribute name="OffsetY" type="xsd:integer" use="required"/>
109. <xsd:attribute name="DefaultUnitsText" use="required">
110. <xsd:simpleType>
111. <xsd:restriction base="xsd:integer">
112. <xsd:enumeration value="48"/>
113. <xsd:enumeration value="49"/>
114. <xsd:enumeration value="50"/>
115. <xsd:enumeration value="51"/>
116. <xsd:enumeration value="52"/>
117. <xsd:enumeration value="53"/>
118. <xsd:enumeration value="54"/>
119. </xsd:restriction>
120. </xsd:simpleType>
121. </xsd:attribute>
122. <xsd:attribute name="DefaultUnitsAngle" use="required">
123. <xsd:simpleType>
124. <xsd:restriction base="xsd:integer">
125. <xsd:enumeration value="80"/>
126. <xsd:enumeration value="81"/>
127. <xsd:enumeration value="82"/>
128. <xsd:enumeration value="83"/>
129. <xsd:enumeration value="84"/>
130. <xsd:enumeration value="85"/>
131. </xsd:restriction>
132. </xsd:simpleType>
133. </xsd:attribute>
134. <xsd:attribute name="DefaultUnitsDuration" use="required">
135. <xsd:simpleType>
136. <xsd:restriction base="xsd:integer">
137. <xsd:enumeration value="42"/>
138. <xsd:enumeration value="43"/>
139. <xsd:enumeration value="44"/>
140. <xsd:enumeration value="45"/>
141. <xsd:enumeration value="46"/>
142. <xsd:enumeration value="47"/>
143. </xsd:restriction>
144. </xsd:simpleType>
145. </xsd:attribute>
146. <xsd:attribute name="DefaultUnitsPage" use="required">
147. <xsd:simpleType>
148. <xsd:restriction base="xsd:integer">
149. <xsd:enumeration value="43"/>
150. <xsd:enumeration value="44"/>
151. <xsd:enumeration value="45"/>
152. <xsd:enumeration value="46"/>
153. <xsd:enumeration value="47"/>
154. <xsd:enumeration value="50"/>
155. <xsd:enumeration value="51"/>
156. <xsd:enumeration value="53"/>
157. <xsd:enumeration value="54"/>
158. <xsd:enumeration value="63"/>
159. <xsd:enumeration value="64"/>
160. <xsd:enumeration value="65"/>
161. <xsd:enumeration value="66"/>
162. <xsd:enumeration value="67"/>
163. <xsd:enumeration value="68"/>
164. <xsd:enumeration value="69"/>
165. <xsd:enumeration value="70"/>
166. <xsd:enumeration value="71"/>
167. <xsd:enumeration value="72"/>
168. <xsd:enumeration value="73"/>
169. <xsd:enumeration value="74"/>
170. <xsd:enumeration value="75"/>
171. <xsd:enumeration value="76"/>
172. </xsd:restriction>
173. </xsd:simpleType>
174. </xsd:attribute>
175. </xsd:complexType>
176. <xsd:element name="Page" type="CT\_Page"/>
177. </xsd:schema>

## ShapeTextBinding Schema

1. <xsd:schema id="ShapeTextBinding">
2. <xsd:complexType name="CT\_TextRun">
3. <xsd:attribute name="Name" type="xsd:string" use="required"/>
4. <xsd:attribute name="Formula" type="xsd:string" use="optional"/>
5. <xsd:attribute name="Format" type="xsd:string" use="optional"/>
6. <xsd:attribute name="Color" type="xsd:string" use="optional"/>
7. <xsd:attribute name="DataRefs" type="xsd:string" use="optional"/>
8. <xsd:attribute name="DataRefsColor" type="xsd:string" use="optional"/>
9. </xsd:complexType>
10. <xsd:complexType name="CT\_ShapeText">
11. <xsd:sequence>
12. <xsd:element name="TextRun" minOccurs="1" maxOccurs="1" type="CT\_TextRun"/>
13. </xsd:sequence>
14. <xsd:attribute name="Name" type="xsd:string" use="required"/>
15. </xsd:complexType>
16. <xsd:complexType name="CT\_ShapeTextBinding\_Page">
17. <xsd:sequence>
18. <xsd:element name="ShapeText" minOccurs="0" maxOccurs="unbounded" type="CT\_ShapeText"/>
19. </xsd:sequence>
20. <xsd:attribute name="Name" type="xsd:string"/>
21. </xsd:complexType>
22. <xsd:element name="Page" type="CT\_ShapeTextBinding\_Page"/>
23. </xsd:schema>

## ShapeOutline Schema

1. <xsd:schema id="ShapeOutline">
2. <xsd:complexType name="CT\_Path">
3. <xsd:attribute name="Shape" type="xsd:string" fixed="poly"/>
4. <xsd:attribute name="Data" use="required">
5. <xsd:simpleType>
6. <xsd:restriction base="xsd:string">
7. <xsd:pattern value="\d\*(,\d\*){3,}"/>
8. </xsd:restriction>
9. </xsd:simpleType>
10. </xsd:attribute>
11. </xsd:complexType>
12. <xsd:complexType name="CT\_Shape">
13. <xsd:sequence>
14. <xsd:element name="Path" minOccurs="1" maxOccurs="unbounded" type="CT\_Path"/>
15. </xsd:sequence>
16. <xsd:attribute name="Name" type="xsd:string" use="required"/>
17. <xsd:attribute name="Layout" type="xsd:string"/>
18. </xsd:complexType>
19. <xsd:complexType name="CT\_Shapes">
20. <xsd:sequence>
21. <xsd:element name="Shape" minOccurs="0" maxOccurs="unbounded" type="CT\_Shape"/>
22. </xsd:sequence>
23. </xsd:complexType>
24. <xsd:complexType name="CT\_ShapeOutline\_Page">
25. <xsd:sequence>
26. <xsd:element name="Shapes" type="CT\_Shapes"/>
27. </xsd:sequence>
28. </xsd:complexType>
29. <xsd:element name="Page" type="CT\_ShapeOutline\_Page"/>
30. </xsd:schema>

# Appendix B: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include updates to those products.

* Microsoft SharePoint Server 2010
* Microsoft Visio 2010
* Microsoft SharePoint Server 2013
* Microsoft Visio 2013
* Microsoft SharePoint Server 2016
* Microsoft Visio 2016
* Microsoft SharePoint Server 2019
* Microsoft Visio 2019
* Microsoft Visio 2021
* Microsoft SharePoint Server Subscription Edition Preview

Exceptions, if any, are noted in this section. If an update version, service pack or Knowledge Base (KB) number appears with a product name, the behavior changed in that update. The new behavior also applies to subsequent updates 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.

# Change Tracking

This section identifies changes that were made to this document since the last release. Changes are classified as Major, Minor, or None.

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.
* A document revision that captures changes to protocol functionality.

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 **None** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the relevant technical content is identical to the last released version.

The changes made to this document are listed in the following table. For more information, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com).

| Section | Description | Revision class |
| --- | --- | --- |
| [6](#Section_bd6d77a1e42d482b88d6c213964a63a8) Appendix B: Product Behavior | Updated list of supported products. | major |
| 6 Appendix B: Product Behavior | Updated list of supported products. | Major |

# Index

A

[ABNF and full grammar definition](#section_9dd6fede2e2a4b4f9193df3626a781bd) 76

[app schema](#section_ebbad19a77c24cf0974c080cc65c0407) 197

[App XML part example](#section_7efc636f7c194cf983a0509fd772e736) 177

[Applicability](#section_fc2d21c58d1140898d856cebfddbba58) 17

C

[Change tracking](#section_976d155eb9364b9e84433355602fe17c) 206

[Common data types and fields](#section_8dcfbbd43c0a4166bc50a44293ec38e8) 18

Complex types

[CT\_Binding](#section_5ad7ddf78438400abe979591e0cca24a) 42

[CT\_BindingConnection](#section_7692f89f4a9a4267b1b5ba319675ca62) 43

[CT\_BindingConnections](#section_4fc00ba8f24243b69cc791796bc6c238) 43

[CT\_Bindings](#section_096f1897249c40c487ba6023f438c74e) 42

[CT\_Connections](#section_812ac2f1a2e94ace94f7e6fe2837009c) 50

[CT\_DataColumn](#section_f5259dad7ba3465092d7ef1d69cc6a70) 46

[CT\_DataColumns](#section_2084702f494442eb8bf757fc33425d26) 48

[CT\_DataConnection](#section_7e5607b6320948fa9f6dc172500e1dd7) 44

[CT\_DataConnections](#section_918bfb0865234923a664b44f66011e43) 45

[CT\_DataGraphicDef](#section_52cb1fef11f541c5a70f89aec95917a0) 51

[CT\_DataGraphicDefinitions](#section_abcf02bac3304ecdb03652568f71f0f9) 51

[CT\_DataGraphicDefs](#section_c4e5d982bba44fc19128daa4f175f249) 61

[CT\_DataGraphics](#section_81e7a61453b0487eacebb684a078cab8) 59

[CT\_DataGraphics\_Page](#section_1656fc08de3a4c088c7a5ed89568c0b6) 58

[CT\_DataGraphics\_Shape](#section_24cbea2f9ad1441a81f41db90fb3dd52) 57

[CT\_DataRecordset](#section_09771dd2110c41e9ad2b609514a2c79c) 48

[CT\_DataRecordsets](#section_7097a58e6c8d428a9c088aadbaf427a1) 50

[CT\_FormulaReferences](#section_c376a1af36d2469686af16d31ea7afeb) 57

[CT\_Formulas](#section_7f2ff83ae58b49b2a8f45e273ba8a371) 56

[CT\_Geometry](#section_984113db7f1a430fb5b90f70e3593b50) 51

[CT\_Hyperlink](#section_0cd8a3b16777487ea09dc7f31460d3e0) 64

[CT\_Hyperlinks](#section_155b1e53317a4af1b82ca10f8e346210) 65

[CT\_IconSetDef](#section_fc407930f282426b9d378c8a09a296ad) 60

[CT\_IconSetRule](#section_3d0b2cb1726b4aa6908159af8bfe6a3b) 60

[CT\_Page](#section_020589d88a1f420c97805dd19dac1cf7) 67

[CT\_PageInfo](#section_3eddc14246c1484497128d5790bdb49d) 61

[CT\_Pages](#section_45b420de6cd2463e95d3b7bc083a0b18) 62

[CT\_Path](#section_37b1961441894431bf9967fa8ad6e2f7) 71

[CT\_PrimaryKey](#section_80c4c5601faa4ad48c8d3aab7b8b3188) 45

[CT\_PrimaryKeys](#section_514c708d371c4dd3907ca4875ec1fb91) 46

[CT\_PrimaryKeyValue](#section_72fa7d7e26164b8bbb1b3b72453e0462) 41

[CT\_PrimaryKeyValues](#section_37d1d095aebc4480b1cf30acc6990811) 41

[CT\_Shape](#section_6defd531916c4391bbb80bc71d0c0c5c) 72

[CT\_ShapeData](#section_a90cbb53af5347ae9da20afddd00cf2f) 62

[CT\_ShapeDataItems](#section_67db1c6345344b4e86ae8bc52c53deec) 64

[CT\_ShapeInfo](#section_95489a96487e434db23af35fd0f877a0) 66

[CT\_ShapeOutline\_Page](#section_dec60ada5c5f4622b28b7c486527b022) 73

[CT\_Shapes](#section_f63911d856514018837174226fb2b6ab) 73

[CT\_ShapeText](#section_80e46fba919845399e9b8c1147fccc25) 75

[CT\_ShapeTextBinding\_Page](#section_c4f3d4ae0c3d4088a40d9e3e1471dd99) 76

[CT\_Sheet](#section_50dec7c1701c454c90ca55fbec1e724a) 53

[CT\_TextRun](#section_9b45f00ef7384cfba633da8bca9b0af4) 74

[Compound file structure overview](#section_71416c996b32472eb1b85f46497e2df2) 18

Conceptual overview

[data binding to web drawing elements](#section_30d7029b15b049e5a5ad1cee2e87a2c8) 27

[data connectivity and refresh](#section_8aae518c4bca49968a84541ff996d01a) 26

[drawing page](#section_ebba8748c5a54ac7a4191c6a70307640) 24

[recalculating shape properties](#section_5f97357e86884d1186fca41ee7546314) 28

[shape](#section_6f93675ffa84476c92dd9d3f32d9b22b) 25

[web drawing](#section_124a373ef5ca47bfbde823c9db618c25) 24

[Custom input type definitions](#section_0f6e4ab0c4c44182b2d1ca077f19afd5) 166

[vBoolean](#section_8a0254e2d0bc4714be02f7104ff17c1f) 166

[vColor](#section_9b24cbcbe0544d9ea92dccd6dde23908) 167

[vDouble](#section_6b18e891cafc4cea9e6f2daaf85c5f2d) 167

[vDoubleEx](#section_17ab59629f9e4b75a84af0cc80400b88) 168

[vFloat](#section_1d9b3dd5ba94418ba9853237ae1cf224) 168

[vSignedInt](#section_6e6498757ff54acfa20b54eb9d8cd472) 168

[vSignedLong](#section_52e79e59b9eb4e8a8a7e92d0cdbdecf8) 168

[vString](#section_fe954fee44b243f1b639ae8ad580be3b) 169

[vUnsignedInt](#section_44ec4229ed484f498abffdb0490d6f1a) 169

[vUnsignedLong](#section_1bf83d6915fa4b6dad66865028333cae) 169

[Custom internal unit types](#section_1ef4d381d8284ad094e4227751b7b6d2) 170

[angleInternalUnitNumber](#section_51d7e07cd65741efa808ebb426864c91) 171

[durationInternalUnitNumber](#section_79862c6e7c6b4af5b33e935edf99db5b) 171

[lengthInternalUnitNumber](#section_e88a0a895198463e833075e8f4f1d981) 171

[typographicInternalUnitNumber](#section_9ba74244fcbe42229c3a78fcbd9db05e) 171

[Custom structures](#section_c1c8b5108b6d43dc8ef4bd4ab3e22fb6) 171

[vCalendar](#section_4c2dd2c033874a6c99af9b277c89b411) 171

[vCurrencyID](#section_b13bb6692bc04291b31c4532158fa675) 172

[vFormatString](#section_3d8a401ea64247d082ad4cfbdd2331b9) 175

[vLanguageID](#section_eee4cf5d442246928202e581f8be4d24) 175

[vServerAction](#section_094c818d877844b881c7dd957e2aaf7f) 175

[vUnitLabel](#section_b8166c01556f4257b5bcc081923ee37d) 176

[Custom token groupings](#section_abd3b6779b2348f9b4a9774346798171) 170

[vAngle](#section_8609b9afc5aa4a0e9c37b1f7929ac9b4) 170

[vAny](#section_3a9fd4fb2af1480b9cd1c40c8640d1c0) 170

[vNum](#section_ad426b3dbea2407c8687202b9d13adad) 170

[vNumAny](#section_1d4fd8bc32844cb5915e14424a11a789) 170

[vUnitType](#section_c9e68c3c1b3e4dd6af70f30cf7f15a0b) 170

D

[Data binding to web drawing elements](#section_30d7029b15b049e5a5ad1cee2e87a2c8) 27

[data binding](#section_b9b194892da24d77a8a51c99bb866fe1) 27

[data graphics](#section_037c6f2c08934c85aaf588143a8d9075) 28

[diagram update](#section_b30a1cbd988f4d6bbf2e7219c9423972) 28

[Data connectivity and refresh](#section_8aae518c4bca49968a84541ff996d01a) 26

[data connections](#section_28d3d364de644cada48144231aed4d7b) 26

[recordset](#section_d7cde52f3fed4fd6a692ad6ecb672ed2) 27

[recordset refresh](#section_07b8ea46bb6a4b2b92853c88ad3f1f86) 27

[recordset row addressing](#section_2921c4da365349d9beef15dfa624c959) 27

[Data types and fields - common](#section_8dcfbbd43c0a4166bc50a44293ec38e8) 18

[DataBinding schema](#section_2b98a7986f9b4b4883716e4eef70a84d) 197

[DataBinding XML part example](#section_761b227542d1472b82b48fab2747c78f) 182

[DataConnection schema](#section_cbc46b5891b0460fa2a8fba03da818e4) 198

[DataConnection XML part example](#section_8d3c08d504b245cf850e803ef56081cc) 183

[DataGraphic schema](#section_b480bb2ee41e4d65a1f95fb7bc7f0b98) 199

[DataGraphic XML part example](#section_0c399b5a77434ff78c2d6e88b3c3414f) 184

[DataGraphicDefinition schema](#section_37eb6952e60c4400a606434e1dba0933) 199

Details

[common data types and fields](#section_8dcfbbd43c0a4166bc50a44293ec38e8) 18

[Document with a Shape on a Page example](#section_8eb6359ce84042a088a74da4f114a220) 177

[Document with Recalculated Visual Properties example](#section_d500fef7c1c7466dbaf12916e6a48aad) 181

[Drawing page](#section_ebba8748c5a54ac7a4191c6a70307640) 24

E

Elements

formatting

[fill attributes](#section_19df61cc63694108ba84ac716f720fde) 37

[shadow canvas](#section_d15c7fe1a64d48c9942eb1189ac6a6a9) 37

[stroke attributes](#section_31f519dff14b4b538582b0bf9358c066) 37

geometry

[geometry path specifications](#section_78f60135434b407984bdd1062fe933f6) 38

global

[BindingConnections](#section_78da47b061264b4388ca71683d5dd702) 41

[Connections](#section_0b609eff79274eb5abc93b7c5c307d29) 44

[DataGraphicDefs](#section_c32b94b9d8cd4f96825939486e776195) 60

[DataGraphics](#section_9fde0d0eb7a949bba229ce9a7b000d87) 50

[Page - ShapeInfo](#section_8add6b5b5d1747ec84418fd27c5b7f77) 61

[Page - ShapeOutline](#section_fffd3fb79dba430dbe611c200624a659) 71

[Page - ShapeTextBinding](#section_665d689445e34b03bf469a8c2002e37d) 73

model

[model ellipses](#section_9fcca8d6c94b4989bbfb6dea992d1737) 38

[model paths](#section_c5a0c47a44fe44948a8bd746bb232275) 38

sheet

[shape transform](#section_c36eb41e53144ecb808acfc9ccd917be) 36

text

[glyph canvas](#section_4588ece693c04e5b8c803e0870294bd2) 39

[text model](#section_81df0650e4aa403abaa976490f1f9ab5) 39

[text run](#section_8a573079d4b14123825dcb299f45d6b0) 40

[Examples](#section_eebab70c52c94ee1b97634154553bcdc) 177

[App XML part](#section_7efc636f7c194cf983a0509fd772e736) 177

[DataBinding XML part](#section_761b227542d1472b82b48fab2747c78f) 182

[DataConnection XML part](#section_8d3c08d504b245cf850e803ef56081cc) 183

[DataGraphic XML part](#section_0c399b5a77434ff78c2d6e88b3c3414f) 184

[Document with a Shape on a Page](#section_8eb6359ce84042a088a74da4f114a220) 177

[Document with Recalculated Visual Properties](#section_d500fef7c1c7466dbaf12916e6a48aad) 181

[Formula Evaluation](#section_93f81c51a16a4c02b7ab59741cf3e656) 193

[overview](#section_eebab70c52c94ee1b97634154553bcdc) 177

ShapeInfo XML part ([section 3.1.2](#section_43827d6356eb4fbba21373b5ad80dbae) 178, [section 3.2.4](#section_a7e5792b994b49f0aeaf15f5868af5f6) 187)

[ShapeOutline XML part](#section_3e8b7cc383834658ae381c4034f8a94e) 179

[ShapeTextBinding XML part](#section_eba55c872520433f8a67a3ce856430dc) 189

XAML ([section 3.1.4](#section_29f6f7436dd641eda0e82a52c01a5e8c) 180, [section 3.2.6](#section_516be87c8cb0410fb82637f6d5eb73f6) 190)

F

[Fields - security index](#section_90ae08555d0a4fe287946d80722f236a) 196

[Fields - vendor-extensible](#section_5aa8158b920141acbd91401bfe9ce871) 17

File structure overview

[compound file](#section_71416c996b32472eb1b85f46497e2df2) 18

[overview](#section_3328b64021d548c9a79702e72c6b0f13) 18

[package](#section_ff4eda2d09694c3c8a4e8afa1e16d499) 19

[part enumeration](#section_24caf49c97804bd0827c61d9cc647ab3) 19

[parts](#section_66eb42ec5ca8459696291233020c701f) 19

[relationships](#section_d7daa0e06e22406f96d9655bc5f29e78) 19

[streams](#section_1fc77f54c9d74549aca5a7bed511c016) 19

Fonts

[XAML images](#section_2183bf65abb049839cc9c5a99409c721) 35

[XAML resources](#section_7d86d940fa714c5284fc572f44845f1e) 32

[Formatting elements](#section_b08c0179146b4b2dbdce35c55c7f7737) 37

Formula evaluation

[ABNF and full grammar definitions](#section_9dd6fede2e2a4b4f9193df3626a781bd) 76

[custom input type definitions](#section_0f6e4ab0c4c44182b2d1ca077f19afd5) 166

[custom internal unit types](#section_1ef4d381d8284ad094e4227751b7b6d2) 170

[custom structures](#section_c1c8b5108b6d43dc8ef4bd4ab3e22fb6) 171

[custom token groupings](#section_abd3b6779b2348f9b4a9774346798171) 170

[function token definitions](#section_33090309eb7443d1be514dd4617fe997) 82

[function token table](#section_a5e026e746284da18c6f1e6823e72c02) 78

[introduction](#section_51452e133e544c73808f90ae80320fc0) 76

[parse token definitions](#section_ae5070d120d946c7ba3c1ad07ed812c4) 154

[parse token table](#section_1c7c6787214545db854e2a1e1ca24d69) 152

[Formula Evaluation example](#section_93f81c51a16a4c02b7ab59741cf3e656) 193

[Full XML schema](#section_1a038b564bbb4bde998a78233de627c6) 197

[Function token definitions](#section_33090309eb7443d1be514dd4617fe997) 82

[Abs](#section_ac15962f890348f99b787d74a264ff70) 82

[ACos](#section_7762f008b9294ca4bfbbcc6fda5857a2) 83

[Add](#section_665760e6818241ccbb499eb6784a17a6) 83

[And](#section_2ed042522c4e4127a430bc711205b624) 85

[Ang360](#section_e60c124063804a61a949983cd68c2faa) 85

[ASin](#section_8d23b8911c7548e0b1aa375b4d35752a) 85

[ATan](#section_8fd8722125b34d8684a2e4ec2a8232fe) 86

[ATan2](#section_d736ae1d0195473480298025561413ae) 86

[BitAnd](#section_7256bb6117d4492980b790555ed5641a) 87

[BitNot](#section_5d8adde5567f436ab018aa972a1ca3b4) 87

[BitOr](#section_7bb67b2740794448bd12439129b78927) 88

[BitXor](#section_72d57fa3d3d643eaab25ac53c0f385ae) 88

[Blend](#section_ae98bf6e773b4822ace6c7f408c5d72e) 89

[Bound](#section_bbcae47f0b9045e7b977266ed8ff95f9) 89

[Cat](#section_96f779a649c541879d0698b956760265) 91

[Ceiling](#section_8f189e0061184600837579b7c614a8ad) 91

[CellIsThemed](#section_a2a02f04068e44b290d7cdc8643f2023) 92

[Char](#section_6c2ccd755ee4427c9cb24f414e43995e) 92

[Cos](#section_5dd3ee1bd91441b4a0d0d6f9c20020d8) 93

[CosH](#section_f378fdf119d04ecf80fc73e806342f3d) 93

[CY](#section_75f0d1d2078744afa3bcdce85676591c) 94

[Date](#section_3112fa9d82044dcfa53fc58f2b2bbb99) 94

[DateTime](#section_7e46243bfaaa4fff8e620dea9bf123d1) 95

[DateValue](#section_b446491e250d45d19cf1e0366b99ab76) 96

[Day](#section_fb6a38429d3a46b2a2382da4f0280b58) 96

[DayOfYear](#section_46b2e2f7565b493299c9a57846a7e32c) 97

[Deg](#section_3624b42b2a034f7dadc5dfaac36ee3a9) 97

[Div](#section_22f5e4257463428c8b879257310dce28) 98

[EEQ](#section_e05c6f8efaa944948905032e95b2d7e7) 99

[EGE](#section_fe352fc6b52a4bcda51bfe343e63ed0e) 100

[EGT](#section_417a0e1d182f43f39090e0e2bb6cb4e9) 100

[ELE](#section_5be29110bd0f4c889977bb5acadd06e8) 101

[ELT](#section_82ef1ff2f70f40b5a645846d3f3158f4) 101

[ENE](#section_743eb63035bf4b27a9a66455c5582c1b) 102

[FEQ](#section_09cc3a4c55b541f8b7e7587543f09f6b) 102

[FGE](#section_ecc5f04605fe4bc8b55527b8af39d6e2) 103

[FGT](#section_cdd9657a9451406da2c8ce4476c5767f) 103

[Find](#section_6d05d919d6504b72a245cfc1208aa564) 104

[FLE](#section_7c57da47a59a482189c4c641a88dd2c2) 105

[Floor](#section_dab6b8dc3bcf4c858cad5ee336228a6a) 105

[FLT](#section_8343e93a1bee4813aa469079707f8773) 106

[FNE](#section_933a601244c940ddb0a0fa936d8a48ca) 106

[FormatEx](#section_5043dc904a694be0a82ee78817288f13) 107

[Hour](#section_ae1d796b6ace40ceb40365e42e6f053a) 109

[HSL](#section_34029dd83de24dccb178577569bfae12) 109

[Hue](#section_5c026f3c342d4c4fbb6b7788a9120a7f) 110

[HueDiff](#section_4661b142b56e44aeab179b5380bf1c5a) 110

[Index](#section_387f76986c5a48b9ab7bd2ae1d7a8e56) 111

[Int](#section_8fc9b021fa0141d1866e224f8bd0719f) 112

[IntersectX](#section_19e8ea77229a4f2a89f3f5c837ce9d35) 112

[IntersectY](#section_756cf723a35d4f16b9e622cba43c97d1) 113

[Intup](#section_35a7b694448342c3b476c74eaa02f40a) 114

[IsErr](#section_daad281ef1d249c092d54c023d4a6679) 114

[IsErrNA](#section_d1c99e1d3db14b9084eb3bada4f9a4b6) 115

[IsError](#section_447254752e8446008be8460fda2f67d0) 115

[IsErrValue](#section_afc2ec8cc6ab47c79fadb6151a48f7ca) 115

[Left](#section_ad6229bc1c014ce79aba8dd70c559dab) 116

[Len](#section_622fcc92f34b46fca1e9963431a8053d) 116

[Ln](#section_4cf5e3dcee9647e698169a0dd75a0883) 117

[Log10](#section_3e5556f668894bd289204faeb44fa4b3) 117

[Lookup](#section_73b5f09a88274eba932ba7c6deac33b9) 118

[Lower](#section_c81fcfe8b9e440ddbd72e0efb23103f9) 118

[Lum](#section_aa06018e8235479db01da2d02aa0b9ba) 119

[LumDiff](#section_ef9c2489cf8e489383e58757c5b2b57a) 119

[Magnitude](#section_8013d759a53c44078499a4b7f6a505e7) 120

[Max](#section_ce8b0939dcef4c6196f733b6165a7739) 121

[Mid](#section_d16240afddeb48dda574b287c66340ed) 121

[Min](#section_6b30c62f628542a5bbc43a660cca4a3e) 122

[Minute](#section_aadc518bedfa4c5cb9153bdb01b39d3a) 122

[Modulus](#section_05d2a9ec70254d6bab0b3564a4644544) 123

[Month](#section_94f622168a994fd59202e6dfb43fe4ce) 123

[MsoShade](#section_5c930f5672654b7eb2c921016b7784b5) 124

[MsoTint](#section_edb72710046d423283c3e38f94628e00) 124

[Mul](#section_9ea4b2faacb44423a91f850be85a98fc) 125

[Not](#section_9f1f0c0af6f34965ba55af7aa0c60c93) 126

[Now](#section_98767867233649c2a290958683bc9a0d) 127

[Or](#section_282f7e8edffc4c2f9dc442fbbec44446) 127

[Pct](#section_b0b58c93e77841779a32c20890802739) 127

[Pi](#section_01fcf3f9c3f345b2b5cc5ff9f00afc81) 128

[Pnt](#section_bc0addccf70a4e5c981d0ff88d23ba09) 128

[Pow](#section_aecdce6ab67b4f4bbfcb9398b6fec21d) 129

[Rad](#section_75a45a7ca53e43d3947f8ad7fa30157d) 130

[Rand](#section_79d086a25da64ee29829be5af12835bf) 130

[Replace](#section_570ef2505bcd40388afa1a875bf50770) 130

[RGB](#section_db09d78046eb4ad19dc749ebc187a383) 131

[Right](#section_d2e4d8ac78d34641811157faec3c6b98) 132

[Round](#section_6ec2ac090e1a4fb9b212ff9233a2adb4) 132

[Sat](#section_283a1d24a19b48cc8f3eae152ed12e33) 133

[SatDiff](#section_9bf7e00cb38441debc494d356e5a29dc) 133

[Second](#section_4bff656d866140d4a27b3896e503b624) 134

[SetAtRef](#section_e4b35b9aebca46ea8f74c6441aa09f68) 134

[SetAtRefEval](#section_422d1c250b3640efbd8591a3b7b53a26) 135

[SetAtRefExpr](#section_120b16678db140ae9d93fed35414aea8) 135

[Shade](#section_9f30bd848ad346d8aae90e2a7b73188f) 136

[ShapeText](#section_cacc5273bcda480b94a9e8433103819a) 136

[Sign](#section_d09e0bd732f54f288c734b9bd31f4f19) 137

[Sin](#section_3431c35c5fbe4d979c2d11b38b10c8f2) 137

[SinH](#section_e47a000ce6d24637942a15d0c4a37332) 138

[Sqrt](#section_21f56dff97624702a8d653f31bdf3161) 138

[StrSame](#section_d78370656e7e4d50ac6258dd77c7b7cf) 139

[StrSameEx](#section_c0434b6c5f5148ed8cff1fe499c73262) 139

[Sub](#section_1bba6fe5ccae4beca707af111c3798e8) 140

[Substitute](#section_fbbbca96110b49098720b105389044de) 142

[Sum](#section_20d1ad033898478fb5b9f2fb17b49590) 143

[Tan](#section_9ee23ba8f95d4c0ea8502374ec465e38) 143

[TanH](#section_661cb296b35c4c14bd69d27132354595) 144

[TextHeight](#section_d7d7e4d35ac8493bbf43940f49ddea6e) 144

[TextWidth](#section_afe4155702d94267a3f029b60fa3b527) 145

[Time](#section_291011bc78944de5aed906836a6332b9) 145

[TimeValue](#section_1e82ec257009418bab7d70b41fed1fa2) 146

[Tint](#section_04d6a79942d04ee08b9126a049a0efa7) 147

[Tone](#section_997f3432602d425eb6f804bd1ff70253) 147

[Trim](#section_732cea39bbc848cf8194b56bebbe99ee) 148

[Trunc](#section_cb4984ba303d46949d72b6579222c282) 149

[UMinus](#section_c3d40d6127aa466086661d4778c2cdc4) 150

[UPlus](#section_1fec68737b9d45309309cc81970b42c5) 150

[Upper](#section_46f69d676fc24c849f066749c3ccd5f6) 150

[WeekDay](#section_a3daca1f74f7441c85a32e55f8ef6f17) 151

[Year](#section_ccd3fce2e5944c2b96f8a9821a70f105) 151

[Function token table](#section_a5e026e746284da18c6f1e6823e72c02) 78

G

[Geometry elements](#section_23c25866532c499aa07c9dc558139a9b) 37

[Glossary](#section_f114f9ab0fa047ec8e94ff586d31981d) 12

I

[Image elements](#section_385a6cb3903347b3acca493e5b363057) 40

[Implementer - security considerations](#section_011784d84d1a4bb4a49fdb8fbe0435ae) 196

[Index of security fields](#section_90ae08555d0a4fe287946d80722f236a) 196

[Informative references](#section_31ef2b971e4445f08efcebdf70c622a5) 16

[Introduction](#section_23088f718f134815af4feccaaece7198) 12

L

[Localization](#section_178ba59a88d54611858e016d7c11ba39) 17

M

[Model elements](#section_1c468e4074744dd09853b7f29ed4c462) 38

N

[Normative references](#section_4f7f022a107f491ebcd2f01806f22e00) 15

O

[Overview (synopsis)](#section_3cdf637453924eab9092dbd44cbd6354) 16

P

[Package structure overview](#section_ff4eda2d09694c3c8a4e8afa1e16d499) 19

[Parse token definitions](#section_ae5070d120d946c7ba3c1ad07ed812c4) 154

[PtgAcre](#section_7f6eff5e434a407b8565fe966e0b066c) 154

[PtgAngDD](#section_222188be55a34b90b2c39eb9f5878bab) 154

[PtgAngDft](#section_57e444144344437fb32a6cf2d3b20d2f) 154

[PtgAngDMS](#section_459da555a74b48439f4cbc516a043d30) 155

[PtgAngRad](#section_a09cdad2084647eb97edef0ef25726e0) 155

[PtgBool](#section_f4171c65cc2f43458b8b16adb7361508) 155

[PtgColorRGB](#section_824daa8a2c0e4a46912624580eab2882) 155

[PtgCy](#section_a0035c3665bf4a2aa677a9d3810e85c3) 156

[PtgDataBinding](#section_7442cb26af7f46ce9bd348656835396c) 156

[PtgDate](#section_661d6eb308e34587b15343cf0ad0a1f6) 156

[PtgEDay](#section_f6690609c80a49ed9fffe0dbcfb262a6) 156

[PtgEHour](#section_e31838fa14684d478f193c033df45e4c) 157

[PtgEMin](#section_f58cc7ec8a1246d89edb4b3e00a08ac6) 157

[PtgErr](#section_a2cbc6aecf6f4d0eac103b71a11cf6d2) 157

[PtgESec](#section_01c1da7178fc485c8ee6b48674eeb35b) 157

[PtgEWeek](#section_4aae0088510448abaf3f8ec63f95fd5f) 158

[PtgFunc](#section_8206928cdaaf400daa071a566fe1c88b) 158

[PtgFuncVar](#section_e483d78ecd494ba7aa042f843a09f8df) 158

[PtgHectare](#section_7b405e52f9c34451a7f90ad394edd4aa) 158

[PtgJmp](#section_a2fcbad2d4cd4d0bb4338ec960e51755) 159

[PtgJmpF](#section_81250f6f47d04c3d93cd60e2d8798784) 159

[PtgJmpLabel](#section_5f545b4b47f44ed38932659537323ff9) 159

[PtgJmpT](#section_f605c7795ea249bca765efaf9ae6e1f0) 159

[PtgMissArg](#section_c613ca4769e54f2aa8a84f0c9c8530cd) 159

[PtgNoOp](#section_5d39bf2968f3460e8ae0a46d6262dd4d) 160

[PtgNum](#section_d895c0dbaac64ac3b9390507f58282d0) 160

[PtgNumCM](#section_bc2d88fb83a94d9588803acf4c0cdc27) 160

[PtgNumDft](#section_de132a3cdab04751844609cf15ec3756) 160

[PtgNumF](#section_09bdcb1c922444c8b9c3ecdaa03e0c6d) 160

[PtgNumFI](#section_a3917f0c42e94c71b55c70ea3aed6819) 161

[PtgNumI](#section_ac87036f348249b58cba49659d7598a8) 161

[PtgNumKM](#section_6dc5f00e5e7945678ac890f98cd40706) 161

[PtgNumM](#section_000148cf28984a299ade5cc03f395c6c) 161

[PtgNumMI](#section_e1f97f6e268c4429868c74e9323a7b9e) 161

[PtgNumMM](#section_5fc97e8093e041d7bcd6ddd2c66aa589) 162

[PtgNumMultiDim](#section_e93d3e0048ec4f74a46336021c9e5f84) 162

[PtgNumNM](#section_bf29f1f677664e9e856811466228244f) 162

[PtgNumPct](#section_ef65a2675fd84ba3a1448fb7698f6551) 162

[PtgNumYards](#section_7039749a686144d9bcd9da3946769b16) 163

[PtgPageDft](#section_fe0d479f9b534257a3d13618b7314aaa) 163

[PtgPnt](#section_45fae81bb4e6413394e32daf5c98d790) 163

[PtgPop](#section_a5016a0a9e794b6dbd38231239a97710) 163

[PtgPushTop](#section_46e71a47bb0d441cb00756eb6182c94a) 164

[PtgRecalcRef](#section_1877c57f45234887a93a70db1b9d46e5) 164

[PtgStr1](#section_acd019e2c6c245e68cbd229441afdb69) 164

[PtgTDurDft](#section_1200eb57709c4fcc9d20e2bac20fa493) 164

[PtgTypCD](#section_bf50076676d84a8d900468c81b728b8e) 164

[PtgTypCi](#section_c5f1913417a7448287105b162359a1fa) 165

[PtgTypDft](#section_d66b059e323d45739c91ec57ed148b16) 165

[PtgTypDi](#section_a56c1d526fe9452b9857c97fd84c3825) 165

[PtgTypPi](#section_f3c4aabfb3bb4c68b5b9f175f9620048) 165

[PtgTypPP](#section_997b3916e4734908b63936fd05317e60) 165

[PtgTypPt](#section_b30a64caae964c76ac0728982f26bb31) 166

[PtgUnsWord](#section_9651517171494cd9a74403c2fcca6d5b) 166

[Parse token table](#section_1c7c6787214545db854e2a1e1ca24d69) 152

Part enumeration

[App](#section_a9ba94e7522542ce9a4b8552b7d3f6e7) 21

[ContentType](#section_6f4e2abfa782498d99d07b262e3f53f5) 24

[Core](#section_ef5dbca1b87c456cbc1673fafb319612) 21

[DataBinding](#section_ddb7bd92bfe64998916e8dfb99140798) 22

[DataConnection](#section_83bf3bed01f44b4fb096b7f19c2aced8) 22

[DataGraphic](#section_1fb58579229c4f8b9a7b7b606ee56810) 22

[DataGraphicDefinition](#section_52fd15daf90d485f9e6123ed67a3b063) 22

[Fonts](#section_6b505bc0615e455f9af5cbf71b0e401e) 20

[Images](#section_4cb2e2621f8d44a6b20f0be3a3fb0179) 21

[Rels](#section_f996c29c681640c58fd62e2ab94bac40) 24

[ShapeGraphic](#section_91b39933d4194cff88431368a42a2258) 20

[ShapeInfo](#section_7ab8617e1a9f4fcd909c52b0d983de4b) 23

[ShapeOutline](#section_95cae30c8856445bb58a1de22f34c6b3) 23

[ShapeTextBinding](#section_f40c633a99df45dcb12a9049fca1fb6b) 23

[Part enumeration structure overview](#section_24caf49c97804bd0827c61d9cc647ab3) 19

[Parts structure overview](#section_66eb42ec5ca8459696291233020c701f) 19

[Product behavior](#section_bd6d77a1e42d482b88d6c213964a63a8) 205

R

[Recalculating shape properties](#section_5f97357e86884d1186fca41ee7546314) 28

[color table](#section_40036cbbe7fd48c5b634ca12f18a8adb) 29

[formulas](#section_176ebf0750704600bc590ed4c9d7406a) 29

[unit number](#section_b455d99896fc43d3822a35f666f2b2d7) 32

[References](#section_9d5689fa9e4a4be79753706b10960d12) 15

[informative](#section_31ef2b971e4445f08efcebdf70c622a5) 16

[normative](#section_4f7f022a107f491ebcd2f01806f22e00) 15

[Relationship to protocols and other structures](#section_b90c6cb9048c4685b6a355ad7f251801) 17

[Relationships structure overview](#section_d7daa0e06e22406f96d9655bc5f29e78) 19

S

Schemas

[app schema](#section_ebbad19a77c24cf0974c080cc65c0407) 197

[DataBinding schema](#section_2b98a7986f9b4b4883716e4eef70a84d) 197

[DataConnection schema](#section_cbc46b5891b0460fa2a8fba03da818e4) 198

[DataGraphic schema](#section_b480bb2ee41e4d65a1f95fb7bc7f0b98) 199

[DataGraphicDefinition schema](#section_37eb6952e60c4400a606434e1dba0933) 199

[ShapeInfo schema](#section_94cef3364dea4390b5f7cbce9ed4f55a) 201

[ShapeOutline schema](#section_2557c83ab94c4495b8f69bdb01853a41) 204

[ShapeTextBinding schema](#section_029d6fc6a40f4e03a52931da213a7105) 203

Security

[field index](#section_90ae08555d0a4fe287946d80722f236a) 196

[implementer considerations](#section_011784d84d1a4bb4a49fdb8fbe0435ae) 196

[Shape](#section_6f93675ffa84476c92dd9d3f32d9b22b) 25

[shape data](#section_2c2d05f1d69d45f8bfc6ace1d0bc6374) 26

[shape hyperlinks](#section_1fc5829872d4478d9d86b282342940cc) 26

[shape identification](#section_0355ae7f62234794ba0d918ddf9067e8) 25

[shape selection](#section_ec83e5f78bc341a49c30a9d58f2f5d92) 26

[shape visualization](#section_9e453f130d754a97b54ef4631c4aba2e) 25

ShapeGraphic

[XAML recalculated shapes](#section_0407e80e0a0f4cd7b0d1de23a6a7b3ad) 36

[XAML resources](#section_6b1e6568312a42bb8ab856b28d4c9078) 32

[XAML shapes](#section_9cb62f8d7fc2459180559f852f9b7db9) 36

[XAML terminology](#section_2650bb0ff70f434a934dabbf8b5e5d14) 32

[ShapeInfo schema](#section_94cef3364dea4390b5f7cbce9ed4f55a) 201

ShapeInfo XML part example ([section 3.1.2](#section_43827d6356eb4fbba21373b5ad80dbae) 178, [section 3.2.4](#section_a7e5792b994b49f0aeaf15f5868af5f6) 187)

[ShapeOutline schema](#section_2557c83ab94c4495b8f69bdb01853a41) 204

[ShapeOutline XML part example](#section_3e8b7cc383834658ae381c4034f8a94e) 179

[ShapeTextBinding schema](#section_029d6fc6a40f4e03a52931da213a7105) 203

[ShapeTextBinding XML part example](#section_eba55c872520433f8a67a3ce856430dc) 189

[Sheet elements](#section_94a21e97d2934aee9999f20030a1ee56) 36

[Streams structure overview](#section_1fc77f54c9d74549aca5a7bed511c016) 19

Structures

[conceptual overview](#section_025eb2de7816498e9dceb4eff561d6c1) 24

[file structure](#section_3328b64021d548c9a79702e72c6b0f13) 18

[formula evaluation and shape property recalculation](#section_51452e133e544c73808f90ae80320fc0) 76

[overview](#section_8dcfbbd43c0a4166bc50a44293ec38e8) 18

[ShapeGraphic XML part](#section_d86834c499d248e2b13ab2c8f016eca0) 32

[XML parts](#section_e353d699f4bc454eaf7b20193355ee32) 40

T

[Text elements](#section_b061c2bde75d48edb51047479b51c6ec) 39

[Tracking changes](#section_976d155eb9364b9e84433355602fe17c) 206

V

[Vendor-extensible fields](#section_5aa8158b920141acbd91401bfe9ce871) 17

[Versioning](#section_178ba59a88d54611858e016d7c11ba39) 17

W

[Web drawing](#section_124a373ef5ca47bfbde823c9db618c25) 24

X

XAML example ([section 3.1.4](#section_29f6f7436dd641eda0e82a52c01a5e8c) 180, [section 3.2.6](#section_516be87c8cb0410fb82637f6d5eb73f6) 190)

[XAML recalculated shapes](#section_0407e80e0a0f4cd7b0d1de23a6a7b3ad) 36

[XAML resources](#section_6b1e6568312a42bb8ab856b28d4c9078) 32

[fonts](#section_7d86d940fa714c5284fc572f44845f1e) 32

[images](#section_2183bf65abb049839cc9c5a99409c721) 35

[XAML shapes](#section_9cb62f8d7fc2459180559f852f9b7db9) 36

[XAML terminology](#section_2650bb0ff70f434a934dabbf8b5e5d14) 32

[XML parts](#section_e353d699f4bc454eaf7b20193355ee32) 40

[app](#section_2c93e85a56504f499b49db4ecd19f55e) 41

[core](#section_f1f650910bf743349e3327acea61c645) 41

[DataBinding](#section_311514cd8ef444c9ab42562671155333) 41

[DataConnection](#section_c489eefb7a014941bc37a4d332e66dc8) 44

[DataGraphic](#section_23f5bf65461648cfbeda817f02b9d087) 50

[DataGraphicDefinition](#section_43b985472c2f40eaafc98da28ba27fbb) 60

[introduction](#section_057509dc724345f7ae1b46a2e25f522f) 41

[ShapeInfo](#section_bf230dbc228e4108bd8f93c9cbd4fcac) 61

[ShapeOutline](#section_e81d9654a5d846cfbba59b380b3a8ade) 71

[ShapeTextGinding](#section_2e21d026e11d46ee860a75c6b75bd84a) 73

[XML schema](#section_1a038b564bbb4bde998a78233de627c6) 197