

# [MS-OGRAPH]:

## Office Graph Binary 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](#) or the [Microsoft Community Promise](#). If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting [iplg@microsoft.com](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](#).
- **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](http://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).

**Preliminary Documentation.** This particular Open Specifications document provides documentation for past and current releases and/or for the pre-release version of this technology. This document provides final documentation for past and current releases and preliminary documentation, as applicable and specifically noted in this document, for the pre-release version. Microsoft will release final documentation in connection with the commercial release of the updated or new version of this technology. Because this documentation might change between the pre-release version and the final

version of this technology, there are risks in relying on this preliminary documentation. To the extent that you incur additional development obligations or any other costs as a result of relying on this preliminary documentation, you do so at your own risk.

Preliminary

## Revision Summary

Date	Revision History	Revision Class	Comments
6/27/2008	1.0	New	First release
10/6/2008	1.02	Minor	Revised and edited the technical content
1/16/2009	1.03	Minor	Updated the Intellectual Property Rights Notice
7/13/2009	1.04	Major	Changes made for template compliance
8/28/2009	1.05	Editorial	Revised and edited the technical content
11/6/2009	1.06	Editorial	Revised and edited the technical content
2/19/2010	2.0	Editorial	Revised and edited the technical content
3/31/2010	2.01	Editorial	Revised and edited the technical content
4/30/2010	2.02	Editorial	Revised and edited the technical content
6/7/2010	2.03	Minor	Updated the technical content
6/29/2010	2.04	Editorial	Changed language and formatting in the technical content.
7/23/2010	2.04	None	No changes to the meaning, language, or formatting of the technical content.
9/27/2010	2.04	None	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	2.04	None	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	2.04	None	No changes to the meaning, language, or formatting of the technical content.
3/18/2011	2.04	None	No changes to the meaning, language, or formatting of the technical content.
6/10/2011	2.04	None	No changes to the meaning, language, or formatting of the technical content.
1/20/2012	2.5	Minor	Clarified the meaning of the technical content.
4/11/2012	2.5	None	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	2.6	Minor	Clarified the meaning of the technical content.
10/8/2012	2.6	None	No changes to the meaning, language, or formatting of the technical content.
2/11/2013	2.6	None	No changes to the meaning, language, or formatting of the technical content.
7/30/2013	2.6	None	No changes to the meaning, language, or formatting of the technical content.
11/18/2013	2.6	None	No changes to the meaning, language, or formatting of the technical content.
2/10/2014	2.6	None	No changes to the meaning, language, or formatting of the

<b>Date</b>	<b>Revision History</b>	<b>Revision Class</b>	<b>Comments</b>
			technical content.
4/30/2014	2.7	Minor	Clarified the meaning of the technical content.
7/31/2014	2.7	None	No changes to the meaning, language, or formatting of the technical content.
10/30/2014	2.8	Minor	Clarified the meaning of the technical content.
3/16/2015	3.0	Major	Significantly changed the technical content.
9/4/2015	3.0	None	No changes to the meaning, language, or formatting of the technical content.
7/15/2016	3.0	None	No changes to the meaning, language, or formatting of the technical content.
9/14/2016	3.0	None	No changes to the meaning, language, or formatting of the technical content.
10/17/2016	3.0	None	No changes to the meaning, language, or formatting of the technical content.
4/27/2018	4.0	Major	Significantly changed the technical content.
8/28/2018	5.0	Major	Significantly changed the technical content.
4/22/2021	6.0	Major	Significantly changed the technical content.

# Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>10</b>
1.1	Glossary .....	10
1.2	References .....	14
1.2.1	Normative References .....	14
1.2.2	Informative References .....	15
1.3	Structure Overview (Synopsis) .....	15
1.3.1	Byte Ordering .....	15
1.3.2	Organization of This Documentation .....	16
1.4	Relationship to Protocols and Other Structures .....	16
1.5	Applicability Statement .....	16
1.6	Versioning and Localization .....	16
1.7	Vendor-Extensible Fields .....	17
<b>2</b>	<b>Structures .....</b>	<b>18</b>
2.1	File Structure.....	18
2.1.1	Compound File .....	18
2.1.2	Stream.....	18
2.1.3	Substream.....	18
2.1.4	Record .....	18
2.1.5	Collection of Records.....	19
2.1.6	Future Record .....	19
2.1.6.1	Chart.....	19
2.1.7	Storages and Streams.....	20
2.1.7.1	Component Object Stream .....	20
2.1.7.2	OLE Stream.....	20
2.1.7.3	Workbook Stream .....	20
2.1.7.3.1	Globals Substream .....	20
2.1.7.3.2	Chart Sheet Substream .....	21
2.2	Conceptual Overview .....	22
2.2.1	Charts .....	23
2.2.1.1	Chart Sheet.....	23
2.2.1.2	Data Sheet.....	24
2.2.1.3	Chart.....	24
2.2.1.4	Axis Group .....	25
2.2.1.5	Axis.....	27
2.2.1.6	Chart Group .....	28
2.2.1.7	Legend .....	30
2.2.1.8	Series.....	31
2.2.1.9	Data Point.....	32
2.2.1.10	Data Label .....	33
2.2.1.11	Trendline .....	39
2.2.1.12	Error Bar.....	40
2.2.1.13	Data Table .....	40
2.2.1.14	Attached Label.....	41
2.2.1.15	SPRC.....	41
2.3	Record Enumeration .....	42
2.3.1	By Name .....	42
2.3.2	By Number .....	45
2.4	Records .....	49
2.4.1	AlRuns .....	49
2.4.2	Area.....	49
2.4.3	AreaFormat.....	49
2.4.4	AttachedLabel .....	55
2.4.5	AxcExt .....	56
2.4.6	AxesUsed .....	58

2.4.7	Axis	59
2.4.8	AxisLine	59
2.4.9	AxisParent	60
2.4.10	Bar	60
2.4.11	Begin	61
2.4.12	Blank	61
2.4.13	BOF	62
2.4.14	BOFDatasheet	64
2.4.15	BopPop	64
2.4.16	BopPopCustom	66
2.4.17	BoundSheet8	67
2.4.18	BRAI	68
2.4.19	CatLab	69
2.4.20	CatSerRange	70
2.4.21	Chart	71
2.4.22	Chart3d	71
2.4.23	Chart3DBarShape	73
2.4.24	ChartColors	74
2.4.25	ChartFormat	74
2.4.26	ChartFrtInfo	75
2.4.27	ClrtClient	76
2.4.28	CodePage	77
2.4.29	ColumnWidth	77
2.4.30	Continue	77
2.4.31	Country	78
2.4.32	CrtLine	78
2.4.33	CrtLink	79
2.4.34	Dat	79
2.4.35	DataFormat	79
2.4.36	DataLabExt	80
2.4.37	DataLabExtContents	81
2.4.38	Date1904	82
2.4.39	DefaultText	82
2.4.40	Dimensions	83
2.4.41	DropBar	83
2.4.42	End	84
2.4.43	EndBlock	84
2.4.44	EndObject	85
2.4.45	EOF	85
2.4.46	ExcludeColumns	86
2.4.47	ExcludeRows	86
2.4.48	Fbi	86
2.4.49	Fbi2	87
2.4.50	Font	87
2.4.51	FontX	89
2.4.52	Format	89
2.4.53	Frame	90
2.4.54	FrtFontList	91
2.4.55	FrtWrapper	91
2.4.56	GelFrame	92
2.4.57	IFmtRecord	94
2.4.58	Label	94
2.4.59	Legend	95
2.4.60	LegendException	96
2.4.61	Line	96
2.4.62	LineFormat	97
2.4.63	LinkedSelection	98
2.4.64	MainWindow	99

2.4.65	MarkerFormat .....	99
2.4.66	MaxStatus .....	101
2.4.67	MsoDrawing .....	101
2.4.68	MsoDrawingGroup .....	101
2.4.69	MsoDrawingSelection .....	102
2.4.70	Number .....	102
2.4.71	Obj .....	102
2.4.72	ObjectLink .....	103
2.4.73	Orient .....	104
2.4.74	Palette .....	105
2.4.75	PicF .....	105
2.4.76	Pie .....	106
2.4.77	PieFormat .....	107
2.4.78	PlotArea .....	107
2.4.79	PlotGrowth .....	107
2.4.80	Pos .....	108
2.4.81	Radar .....	110
2.4.82	RadarArea .....	110
2.4.83	Scatter .....	110
2.4.84	Scl .....	111
2.4.85	Selection .....	112
2.4.86	SerAuxErrBar .....	112
2.4.87	SerAuxTrend .....	113
2.4.88	SerFmt .....	115
2.4.89	Series .....	115
2.4.90	SeriesList .....	116
2.4.91	SeriesText .....	116
2.4.92	SerParent .....	116
2.4.93	SerToCrt .....	117
2.4.94	ShtProps .....	117
2.4.95	StartBlock .....	118
2.4.96	StartObject .....	120
2.4.97	Surf .....	122
2.4.98	Text .....	122
2.4.99	Tick .....	127
2.4.100	TxO .....	130
2.4.101	Units .....	132
2.4.102	ValueRange .....	132
2.4.103	WinDoc .....	135
2.4.104	Window1 .....	135
2.4.105	Window1_10 .....	136
2.4.106	Window2Graph .....	136
2.4.107	YMult .....	137
2.5	Structures .....	138
2.5.1	Boolean .....	138
2.5.2	CFrtId .....	138
2.5.3	ChartNumNillable .....	139
2.5.4	DateUnit .....	139
2.5.5	FontIndex .....	139
2.5.6	FontInfo .....	140
2.5.7	FormatRun .....	140
2.5.8	FrtFlags .....	140
2.5.9	FrtHeader .....	141
2.5.10	FrtHeaderOld .....	141
2.5.11	FtCf .....	142
2.5.12	FtCmo .....	142
2.5.13	FtGmo .....	144
2.5.14	Graph_Col .....	144

2.5.15	Graph_Rw .....	144
2.5.16	Icv .....	144
2.5.17	IcvChart .....	147
2.5.18	IcvFont.....	148
2.5.19	IFmt .....	148
2.5.20	LongRGB .....	148
2.5.21	NilChartNum .....	149
2.5.22	OfficeArtClientAnchorChart .....	149
2.5.23	OfficeArtClientData .....	150
2.5.24	OfficeArtClientTextbox.....	151
2.5.25	PositionMode.....	151
2.5.26	Run.....	151
2.5.27	ShortXLUnicodeString .....	152
2.5.28	TxOLastRun .....	152
2.5.29	TxORuns.....	153
2.5.30	XLUnicodeString .....	153
2.5.31	XLUnicodeStringMin2 .....	154
2.5.32	XLUnicodeStringNoCch .....	154
2.5.33	Xnum .....	154
<b>3</b>	<b>Structure Examples .....</b>	<b>155</b>
3.1	Graph - Column Chart .....	155
3.1.1	Graph - Column Chart: Chart .....	156
3.1.2	Graph - Column Chart: Scl.....	156
3.1.3	Graph - Column Chart: PlotGrowth .....	156
3.1.4	Graph - Column Chart: Series .....	157
3.1.5	Graph - Column Chart: Begin .....	157
3.1.6	Graph - Column Chart: BRAI .....	157
3.1.7	Graph - Column Chart: SeriesText .....	158
3.1.8	Graph - Column Chart: BRAI{1}.....	158
3.1.9	Graph - Column Chart: BRAI{2}.....	159
3.1.10	Graph - Column Chart: DataFormat .....	159
3.1.11	Graph - Column Chart: SerToCrt .....	160
3.1.12	Graph - Column Chart: ShtProps .....	160
3.1.13	Graph - Column Chart: DefaultText.....	161
3.1.14	Graph - Column Chart: Text.....	161
3.1.15	Graph - Column Chart: FontX.....	163
3.1.16	Graph - Column Chart: AxesUsed .....	163
3.1.17	Graph - Column Chart: AxisParent .....	164
3.1.18	Graph - Column Chart: Pos .....	164
3.1.19	Graph - Column Chart: Axis .....	165
3.1.20	Graph - Column Chart: CatSerRange.....	165
3.1.21	Graph - Column Chart: Tick .....	166
3.1.22	Graph - Column Chart: Axis{1}.....	167
3.1.23	Graph - Column Chart: ValueRange .....	167
3.1.24	Graph - Column Chart: Tick{1}.....	168
3.1.25	Graph - Column Chart: AxisLine .....	170
3.1.26	Graph - Column Chart: LineFormat .....	170
3.1.27	Graph - Column Chart: Frame .....	171
3.1.28	Graph - Column Chart: LineFormat{1}.....	171
3.1.29	Graph - Column Chart: AreaFormat .....	172
3.1.30	Graph - Column Chart: ChartFormat .....	174
3.1.31	Graph - Column Chart: Bar .....	174
3.2	Graph - Pie Chart .....	175
3.2.1	Graph - Pie Chart: Orient.....	175
3.2.2	Graph - Pie Chart: Chart.....	176
3.2.3	Graph - Pie Chart: ShtProps .....	176
3.2.4	Graph - Pie Chart: AxesUsed .....	177



3.2.5	Graph - Pie Chart: AxisParent .....	177
3.2.6	Graph - Pie Chart: ChartFormat.....	177
3.2.7	Graph - Pie Chart: Pie .....	178
3.2.8	Graph - Pie Chart: Legend .....	178
3.2.9	Graph - Pie Chart: Begin .....	179
3.2.10	Graph - Pie Chart: Pos .....	180
3.2.11	Graph - Pie Chart: Text .....	180
3.2.12	Graph - Pie Chart: Begin{1} .....	182
3.2.13	Graph - Pie Chart: BRAI .....	183
3.2.14	Graph - Pie Chart: End .....	183
<b>4</b>	<b>Security Considerations.....</b>	<b>184</b>
<b>5</b>	<b>Appendix A: Product Behavior .....</b>	<b>185</b>
<b>6</b>	<b>Change Tracking.....</b>	<b>190</b>
<b>7</b>	<b>Index.....</b>	<b>191</b>

Preliminary

# 1 Introduction

This document specifies the Office Graph Binary File Format. This file format is a collection of records and structures that specify a single chart or graph and the tabular data that the chart or graph represents.

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

## 1.1 Glossary

This document uses the following terms:

**3-D Phong shading:** A shading algorithm that is used to apply shading to 3-D charts. The algorithm interpolates color between points on the surface of a chart to give a smooth 3-D appearance.

**active cell:** The cell that is currently selected in a worksheet.

**active selection:** The current selection of cells, rows, or columns that includes the **active cell** in a datasheet.

**anchor:** A set of qualifiers and quantifiers that specifies the location of an element or object within a document. These values are typically relative to another element or known location in the document, such as the edge of a page or margin.

**Augmented Backus-Naur Form (ABNF):** A modified version of Backus-Naur Form (BNF), commonly used by Internet specifications. ABNF notation balances compactness and simplicity with reasonable representational power. ABNF differs from standard BNF in its definitions and uses of naming rules, repetition, alternatives, order-independence, and value ranges. For more information, see [\[RFC5234\]](#).

**background color:** A color against which characters, patterns, and graphics are displayed. See also **foreground color**.

**beta:** A pre-released version of a product that is sent to customers and partners for evaluation and feedback.

**big-endian:** Multiple-byte values that are byte-ordered with the most significant byte stored in the memory location with the lowest address.

**Binary Interchange File Format (BIFF):** The binary file formats that are used to save Excel workbooks.

**border:** A line that can be applied to the outer edge of a cell, shape, object, or chart element. A border can be variously formatted for style, color, and thickness.

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

**bubble size:** A value that represents the diameter of a bubble in a bubble chart.

**build identifier:** An integer that identifies a build.

**category:** A subdivision of items into useful groups such as geographical regions. For example, categories that represent geographical regions could be North, South, East, and West.

**cell:** A box that is formed by the intersection of a row and a column in a worksheet or a table. A cell can contain numbers, strings, and formulas, and various formats can be applied to that data.

**character set:** A mapping between the characters of a written language and the values that are used to represent those characters to a computer.

**chart area:** A region in a chart object that is used to position chart elements, render axes, and plot data.

**chart sheet:** A single logical container that is used to create and store charts in a workbook.

**chart window:** A separate window that is used to display a chart.

**code page:** An ordered set of characters of a specific script in which a numerical index (code-point value) is associated with each character. Code pages are a means of providing support for **character sets** and keyboard layouts used in different countries. Devices such as the display and keyboard can be configured to use a specific code page and to switch from one code page (such as the United States) to another (such as Portugal) at the user's request.

**color palette:** A collection of colors that is available to format text, shapes, cells, and chart elements.

**data label:** A label that provides detailed information about a data point on a chart.

**data marker:** A customizable symbol or shape that identifies a data point on a line, scatter, or radar chart. A data marker can be formatted with various sizes and colors.

**datasheet:** A worksheet window that contains the source data for a Microsoft Graph chart object.

**date system:** A method of calculating calendar dates and times.

**display units:** An axis-formatting option that determines how numeric units are displayed on a value axis.

**distributed alignment:** A formatting setting that spreads text evenly, both vertically and horizontally, between the edges of a cell, object, or page. Distributed alignment is used primarily with East Asian languages. See also **justify distributed**.

**down bar:** See **up-down bar**.

**drawing object:** A shape, curve, line, WordArt, or other type of graphical object that can be inserted into a document.

**drop lines:** A set of supplemental lines on an area chart or a line chart. Drop lines increase the legibility of a chart by connecting each data point in a series to the category axis.

**fill:** A color, pattern, or other attribute that is used to format the background of a cell, shape, or chart element. See also fill color and **fill pattern**.

**fill pattern:** A repetitive design that users can add to the background of a cell, shape, or chart element.

**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.

**floor:** An extension of the horizontal axis, or the area created by the inclusion of the z axis, in a 3-D chart to create a three-dimensional effect. See also **wall**.

**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 strikethrough), size, family (a typeface such as Times New Roman), and other qualities to describe how the collection is drawn.

**font scaling:** A process of resizing a proportionally-spaced font.

**forecast:** The process of projecting values forward or backward in a series, based on trends in existing data.

**foreground color:** A color that is used to display text, patterns, and other objects that appear in front of or on top of the **background color**.

**format string:** A string that contains the number formatting information to apply to data, such as decimal position, percentage indicator, or currency symbol.

**formatting run:** A set of formatting properties that are applied to a **text run**.

**formula:** A logical equation or function that produces a result in a spreadsheet application.

**graph object:** An object that represents a chart and the datasheet that contains the data for that chart.

**high-low lines:** Supplemental lines that are added to a line chart and connect the maximum data points of one series with the minimum data points of another series for each **category**. High-low lines are typically used on stock charts.

**horizontal alignment:** A formatting setting that specifies how content is positioned within the horizontal space of a cell, object, or page. Content can be aligned along the left or right edge, or distributed evenly across the horizontal space.

**justify distributed:** A special, distributed-alignment setting that evenly distributes the last line of text in a cell. The setting is primarily used for East Asian languages. See also **distributed alignment**.

**leader line:** A line that connects a data label to its corresponding data point. The primary purpose of a leader line is to increase legibility.

**left-to-right:** A reading order in which characters in words are read from left to right, and words are read from left to right in sentences.

**legend entry:** An item in a chart legend that identifies a single series or **category**.

**legend key:** A symbol that is associated with a legend entry.

**little-endian:** Multiple-byte values that are byte-ordered with the least significant byte stored in the memory location with the lowest address.

**major gridline:** A horizontal or vertical line that is in the plot area of a chart and corresponds to the major scaling unit on an axis.

**major tick mark:** A **tick mark** that corresponds to a major scaling unit on an axis.

**metafile:** A file that stores an image as graphical objects, such as lines, circles, and polygons, instead of pixels. A metafile preserves an image more accurately than pixels when an image is resized.

**minor gridline:** A horizontal or vertical line that is in the plot area of a chart and corresponds to the minor scaling unit on an axis.

**minor tick mark:** A **tick mark** that corresponds to a minor scaling unit on an axis.

**moving average:** A type of trendline that is calculated based on the most recent period of data points in a series.

**number format:** A property of a cell or other type of object that determines how numerical data is displayed or interpreted. For example, a currency number format affixes the proper currency symbol to the number.

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

**OLE server:** An application or DLL that supplies a linked or embedded OLE object to another application.

**outline:** A nested grouping of rows or columns that are in a worksheet.

**outline effect:** A formatting effect in which a line is placed around the edge of a shape or around each character in a text string.

**out-of-memory:** A state of a computer or application when it halts because all of the available volatile memory has been allocated and none is currently available for reallocation.

**plot area:** A portion of a chart area that contains the plotted data and axes.

**point:** A unit of measurement for fonts and spacing. A point is equal to 1/72 of an inch.

**primary pie:** The main chart in a bar of pie or pie of pie chart. A primary pie chart has one pie slice (data point) that is a grouping of data points.

**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.

**reading order:** The positioning of characters in words and the positioning of words in sentences. This can be left-to-right or right-to-left.

**right-to-left:** A reading and display order that is optimized for right-to-left languages.

**secondary bar/pie:** A secondary chart in a bar of pie or pie of pie chart that displays the detailed data of the grouped data point in the primary pie chart. The secondary bar/pie chart takes the form of a stacked bar chart or a pie chart that is connected to the primary pie chart with series lines.

**selected:** The condition of a set of items that has focus in a workbook.

**selection:** An item or set of items, such as cells, shapes, objects, and chart elements, that has focus in a document.

**series line:** A supplemental line on a stacked column, stacked bar, pie of pie, or bar of pie chart that connects each data point in a series with the next data point to increase legibility.

**shadow effect:** A formatting effect that makes a font or object appear to be elevated from the page or screen surface, and therefore casts a shadow.

**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.

**strikethrough formatting:** A formatting option in which characters are crossed out by horizontal line.

**system palette:** An itemization of all of the colors that can be displayed by the operating system for a device.

**text run:** A string of characters that represents a discrete span of text with the same formatting properties.

**tick mark:** A small line of measurement, similar to a division line on a ruler, that intersects an axis in a chart.

**trendline:** A line that is added to a chart to show the trend of multiple data points in a series. A trendline is used to facilitate regression analysis.

**twip:** A unit of measurement that is used in typesetting and desktop publishing. It equals one-twentieth of a printer's point, or 1/1440 of an inch.

**Unicode:** A character encoding standard developed by the Unicode Consortium that represents almost all of the written languages of the world. The **Unicode** standard [\[UNICODE5.0.0/2007\]](#) 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).

**up bar:** See **up-down bar**.

**up-down bar:** A vertical bar that highlights the difference between data points in a line chart that contains more than one data series.

**vertical alignment:** A formatting setting that specifies how content is positioned within the vertical space of a cell, object, or page. Content can be aligned along the top or bottom edge, or distributed evenly across the vertical space.

**wall:** An extension of the background of a 3-D chart to create a three-dimensional effect. See also **floor**.

**workbook:** A container for a collection of **sheets**.

**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.

**z-order:** The rendering order of an object on a z axis.

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

## 1.2 References

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

### 1.2.1 Normative References

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

[CODEPG] Microsoft Corporation, "Code Pages", <https://docs.microsoft.com/en-us/globalization/encoding/code-pages>

[ECMA-376] ECMA International, "Office Open XML File Formats", 1st Edition, ECMA-376, December 2006, <http://www.ecma-international.org/publications/standards/Ecma-376.htm>

[IEEE754] IEEE, "IEEE Standard for Binary Floating-Point Arithmetic", IEEE 754-1985, October 1985, <http://ieeexplore.ieee.org/servlet/opac?punumber=2355>

[MS-CFB] Microsoft Corporation, "[Compound File Binary File Format](#)".

[MS-ODRAW] Microsoft Corporation, "[Office Drawing Binary File Format](#)".

[MS-OLEDS] Microsoft Corporation, "[Object Linking and Embedding \(OLE\) Data Structures](#)".

[MS-OSHARED] Microsoft Corporation, "[Office Common Data Types and Objects Structures](#)".

[MS-XLS] Microsoft Corporation, "[Excel Binary File Format \(.xls\) Structure](#)".

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

[RFC5234] Crocker, D., Ed., and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008, <http://www.rfc-editor.org/rfc/rfc5234.txt>

## 1.2.2 Informative References

[MSDN-FONTS] Microsoft Corporation, "About Fonts", [http://msdn.microsoft.com/en-us/library/dd162470\(VS.85\).aspx](http://msdn.microsoft.com/en-us/library/dd162470(VS.85).aspx)

## 1.3 Structure Overview (Synopsis)

This file format specifies chart data contained in a compound file, as described in [\[MS-CFB\]](#). The data is stored in the compound file by using storages, streams, and substreams that contain information about the content and structure of a chart. Some storages, streams, and substreams store information by using binary records. The binary record structure and content are specified in this document. Each binary record contains its record type, information about the record size, and zero or more type-specific fields, depending on the record type, which is specified in **Record Enumeration** (section [2.3](#)). Type-specific fields contain information that further specifies the chart data. Specifications of the fields for a record type can be found in **Records** (section [2.4](#)).

For a more detailed overview of file architecture and content, see **File Structure** (section [2.1](#)).

### 1.3.1 Byte Ordering

Some computer architectures number bytes in a binary word from left to right, which is referred to as **big-endian**. This documentation uses big-endian bit diagrams. Other architectures number the bytes in a binary word from right to left, which is referred to as **little-endian**. The underlying file format enumerations, objects, and records are little-endian.

Using big-endian and little-endian methods, the number 0x12345678 would be stored as shown in the following table.

Byte order	Byte 0	Byte 1	Byte 2	Byte 3
Big-endian	0x12	0x34	0x56	0x78
Little-endian	0x78	0x56	0x34	0x12

Unless otherwise specified, all data in files of the type specified by this document are stored in little-endian format.

### 1.3.2 Organization of This Documentation

Section 2 of this document is arranged with overviews of high-level concepts, followed by more detailed concepts. Sections 2.1 and 2.2, in particular, specify high-level concepts that are required to understand the remainder of the document and should be read before reading the remainder of Section 2.

Section 2.1 specifies the structures and concepts that are used to organize and structure the file itself. Subsection 2.1.7 further specifies the valid storages, streams, and substreams that are enabled for files of the type specified by this document.

Section 2.2 specifies higher-level chart concepts for use in later sections of this document.

Section 2.3 specifies the record name associated with a record type. For more information about record types, see **Records** (section [2.4](#)). These associations are listed by record name, as well as by record type.

Section 2.4 specifies the details of individual records.

Section 2.5 specifies the details of structures that are used by records and other structures.

Section 3 provides specific examples that illustrate the concepts, records, and structures of this file format.

Section 4 discusses security issues that relate to files of the type specified by this document.

Section 5 is a list of version-specific behaviors. It is not intended to be read alone, but rather to be understood in the context of the specifications in Section 2. Specifications in Section 2 provide links to the relevant items in Section 5.

### 1.4 Relationship to Protocols and Other Structures

This file format is an embedded object, as described in [\[MS-OLEDS\]](#). It is dependent on the structures defined in the following references:

- [\[ECMA-376\]](#) for number formatting.
- [\[MS-ODRAW\]](#) for the persistence format for shapes.
- [\[MS-OSHARED\]](#) for the persistence format for additional common structures.

### 1.5 Applicability Statement

This document specifies a persistence format that represents a single chart or graph and the tabular data that the chart or graph represents.

This persistence format can be contained in other documents as an embedded object, as described in [\[MS-OLEDS\]](#). It is not applicable for use in stand-alone documents.

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

### 1.6 Versioning and Localization

This document covers versioning issues in the following areas:

**Structure Versions:** There is only one version of this specification.



**Localization:** The following records and structures contain fields that specify locale-dependent meaning:

- Font
- Format
- Tick
- Text

For more information, see each record, structure, and section description.

### **1.7 Vendor-Extensible Fields**

This persistence format can be extended by storing information in streams and storages not specified in section 2. Implementations are not required to preserve or remove additional streams or storages when modifying an existing document.

Preliminary

## 2 Structures

### 2.1 File Structure

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

A file of the type specified by this document is an **OLE compound file**. A compound file contains storages, streams, and substreams. Each stream or substream contains a series of binary records. Each binary record contains zero or more structured fields that contain the chart data.

#### 2.1.1 Compound File

A file of the type specified by this document MUST be an **OLE compound file**, as specified in [\[MS-CFB\]](#).

#### 2.1.2 Stream

A file of the type specified by this document consists of storages and streams, as specified in [\[MS-CFB\]](#). Each binary stream or substream that contains chart data MUST be written as a series of binary records as specified in **Storages and Streams** (section [2.1.7](#)).

The valid storages, streams, and substreams in a file of the type specified by this document are specified in **Storages and Streams** (section [2.1.7](#)).

This document uses **Augmented Backus-Naur Form (ABNF)**, as specified in [\[RFC5234\]](#), to specify the record sequence for streams and substreams that contain binary records.

#### 2.1.3 Substream

The workbook stream contains substreams that specify properties and data for a chart and specify the **chart sheet** that constitute a file of the type specified by this document. The beginning of each substream is marked by a BOF record that has a **docType** field that specifies the type of the substream. The end of each substream is marked by an EOF record.

For more information, see **Workbook Stream** (section [2.1.7.3](#)).

#### 2.1.4 Record

A record is the basic building block that stores information about features in a chart. Each binary record is a variable-length sequence of bytes. A binary record consists of three components: a record type, a record size, and the record data specific to that record type.

The record type is a 2-byte unsigned integer that specifies what type of information is specified by the record and how the structure of the record data of this record is ordered and structured. Each valid record type is listed in **Record Enumeration** (section [2.3](#)).

The record size is a 2-byte unsigned integer that specifies the count of bytes that specifies the total size of the record data.

The record data component contains fields that correspond to a record type and comprise the remainder of the record. The order and structure of the fields for a record type listed in **Record Enumeration** (section [2.3](#)) are specified in the corresponding section for that record type in **Records** (section [2.4](#)). The size of the record data component MUST be equal to the record size. Fields in the record data component can contain simple values, arrays of values, structures of several fields, arrays of fields, and arrays of structures.

If the total number of bytes to be written for a record data component is greater than 8224, a collection of **Continue** records, as specified in the **ABNF** for that record, contains the remaining record data. In general, the first 8224 bytes of data is contained in the record data component of the record, and the remaining data is divided into blocks of 8224 bytes and contained in the subsequent collection of **Continue** records, until all the data is written. See specific record definitions for exceptions.

### 2.1.5 Collection of Records

A collection of records is a series of related records that are treated as a set. Records in a collection do not have to share the same record type. A collection of records can contain other collections of records. The beginning of a collection of records is sometimes indicated by a different type of record, as specified in the record sequence **ABNF** that specifies the rule that contains that record. Similarly, the end of a collection of records is sometimes indicated by a different type of record, as specified by the record sequence **ABNF** that specifies the rule that contains that record. Each record or collection within the collection can be referred to by index, and the index count starts over for each collection.

In this document, record A precedes record B when record A is the last record of that type to appear before record B in the stream or substream that contains those records.

In this document, record B follows record A when record A is the last record of that type to appear before record B in the stream or substream that contains those records.

### 2.1.6 Future Record

The future record architecture enables an application that does not support certain records to open and save the file while ignoring but preserving those records in the file. Records saved by using an **FrthHeader** (section [2.5.9](#)), **FrthHeaderOld** (section [2.5.10](#)), or **FrthFlags** (section [2.5.8](#)) structure are future records and can be treated as such by an application that implements the future record architecture. Future records that are not supported by an application can be cached on load and persisted on save, enabling the data to be retained in the file for an application that does support those records. When loading and caching these unsupported records, an application can process **range** references associated with the unsupported records, as specified in the **FrthHeader**, **FrthHeaderOld**, and **FrthFlags** structures. The application can adjust these range references as **cells** are inserted, deleted, or moved to preserve the correctness and validity of these range references associated with the unsupported records when saving the file along with the unsupported records.

For an in-depth discussion of how these structures relate to each other, see the specification of each structure in **Structures** (section [2.5](#)).

#### 2.1.6.1 Chart

There are three types of future records that can appear in the chart sheet substream.

The first type of future records is chart-specific future records that are stored using the future record architecture, as specified in **Future Record** (section [2.1.6](#)). Specifically, these chart-specific future records have a record type that **MUST** be greater than or equal to 2048 and less than or equal to 2303, as specified in **Record Enumeration** (section [2.3](#)). These future records **MUST** exist in the collection of records specified by the **StartBlock** and **EndBlock** records, unless they exist in the collection of records specified by the **StartObject** and **EndObject** records. The **StartBlock** and **EndBlock** records themselves **MUST NOT** appear in the collection of records specified by the **StartObject** and **EndObject** records. Furthermore, **ChartFrthInfo** **MUST** precede the first chart-specific future record in a **chart sheet** substream.

The second type of future records is stored by wrapping non-future records in the **FrthWrapper** record, as specified in **FrthWrapper** (section [2.4.55](#)). These records **MUST** be written in their entirety—the record type, record size, and record data components, as specified in **Records** (section [2.4](#))—in the

**wrappedRecord** field of the **FrWrapper** record. These records MUST exist in the collection of records specified by the **StartObject** and **EndObject** records.

The third type of future records is all other future records that are stored using the future record architecture, as specified in **Future Record** (section 2.1.6).

See the record definitions for details on how these records are written relative to other future records in the chart sheet substream.

## 2.1.7 Storages and Streams

This section specifies the storages, streams, and substreams of this file format. See **File Structure** (section 2.1) for a description of storages, streams, and substreams.

If the stream is in **Binary Interchange File Format (BIFF)** format, the record sequence is specified using **ABNF** grammar.

### 2.1.7.1 Component Object Stream

This stream is specified in [\[MS-OLEDS\]](#), section [2.3.8](#).

The name of this stream MUST be "\001CompObj", where \001 is the character with the value 0x01, not the string literal "\001".

A file MUST contain at most one component object stream.

### 2.1.7.2 OLE Stream

This stream is specified in [\[MS-OLEDS\]](#), section [2.3.3](#).

The name of this stream MUST be "\001Ole", where \001 is the character with the value 0x01, not the string literal "\001".

A file MUST contain at most one OLE stream.

### 2.1.7.3 Workbook Stream

The workbook stream specifies a chart, as well as associated properties and data for the chart.

The name of this stream MUST be "Workbook". A file MUST contain exactly one workbook stream, which consists of two substreams. There MUST be exactly one globals substream. The globals substream MUST be the first substream to appear in the workbook stream and MUST be followed by exactly one **chart sheet** substream.

#### 2.1.7.3.1 Globals Substream

This substream specifies global properties and data for a chart.

There MUST be exactly one globals substream in a workbook stream, and the globals substream MUST be the first substream in the workbook stream.

Record sequence [<1>](#) **ABNF**:

```
WORKBOOKCONTENT = CodePage Date1904 *Format *Font Window1 BoundSheet8 * (MsoDrawingGroup
  *Continue) EOF
```

```
WORKBOOK = BOF WORKBOOKCONTENT
```

### 2.1.7.3.2 Chart Sheet Substream

This substream specifies a collection of records that contains a chart.

If a record in the **ABNF** grammar for this substream is one of the following and is in a collection specified by **StartObject** (section [2.4.96](#)) and **EndObject** (section [2.4.44](#)), that record MUST be replaced by an **FrWrapper** record (section [2.4.55](#)), and the **wrappedRecord** field of that **FrWrapper** record MUST specify the record that is replaced:

- **AlRuns**
- **AreaFormat**
- **Begin**
- **BRAI**
- **Continue**
- **DefaultText**
- **End**
- **Fbi**
- **Font**
- **FontX**
- **Frame**
- **GelFrame**
- **LineFormat**
- **ObjectLink**
- **PicF**
- **Pos**
- **SeriesText**
- **Text**

**StartBlock/EndBlock** (section [2.4.95](#) and section [2.4.43](#)) record pairs and **ChartFrInfo** (section [2.4.26](#)) are omitted from the following ABNF grammar. See **Chart** (section [2.1.6.1](#)) and the respective record specifications for more information.

Record sequence <2> ABNF:

```
CHARTSHEETCONTENT = *Fbi *Fbi2 [ChartColors] [Palette] [ClrtClient] DATASHEET OBJECTS Units  
                    CHARTFORMATS Dimensions EOF
```

```
CHARTSHEET = BOF CHARTSHEETCONTENT
```

```
DATASHEET = BOF Datasheet Begin Country FontX *ColumnWidth ExcludeRows ExcludeColumns Orient  
            *(Blank / Number / Label) WinDoc MainWindow Window1 Window2Graph MaxStatus Selection  
            [LinkedSelection] End
```

```
OBJECTS = *(MsoDrawing *Continue [OBJ / TEXTOBJECT]) [MsoDrawingSelection]
```

```

OBJ = Obj *Continue

TEXTOBJECT = TxO *Continue

CHARTFOMATS = Chart Begin *2FONTLIST Scl PlotGrowth [FRAME] *SERIESFORMAT *SS ShtProps *2DFTTEXT
AxesUsed 1*2AXISPARENT [DAT] *ATTACHEDLABEL *([DataLabExt StartObject] ATTACHEDLABEL
[EndObject]) End

FONTLIST = FrtFontList StartObject *(Font [Fbi]) EndObject

AXISPARENT = AxisParent Begin Pos [AXES] 1*4CRT End

AXES = [IVAXIS DVAXIS [SERIESAXIS] / DVAXIS DVAXIS] *3ATTACHEDLABEL [PlotArea FRAME]

IVAXIS = Axis Begin [CatSerRange] AxcExt [CatLab] AXS End

DVAXIS = Axis Begin [ValueRange] [AXM] AXS End

SERIESAXIS = Axis Begin [CatSerRange] AXS End

CRT = ChartFormat Begin (Bar / Line / (BopPop [BopPopCustom]) / Pie / Area / Scatter / Radar /
RadarArea / Surf) CrtLink [SeriesList] [Chart3d] [LD] [2DROPBAR] *4(CrtLine LineFormat)
*2DFTTEXT [DataLabExtContents] [SS] End

LD = Legend Begin Pos ATTACHEDLABEL [FRAME] End

SERIESFORMAT = Series Begin 4AI *SS (SerToCrt / (SerParent (SerAuxTrend / SerAuxErrBar)))
*(LegendException [Begin ATTACHEDLABEL End]) End

FRAME = Frame Begin LineFormat AreaFormat [GELFRAME] End

AI = BRAI [SeriesText]

ATTACHEDLABEL = Text Begin Pos [FontX] [AlRuns] AI [FRAME] [ObjectLink] [DataLabExtContents] End

SS = DataFormat Begin [Chart3DBarShape] [LineFormat AreaFormat PieFormat] [SerFmt] [LineFormat]
[AreaFormat] [GELFRAME] [MarkerFormat] [AttachedLabel] End

AXS = [IFmtRecord] [Tick] [FontX] *4(AxisLine LineFormat) [AreaFormat] [GELFRAME]

DFTTEXT = [DataLabExt StartObject] DefaultText ATTACHEDLABEL [EndObject]

DROPBAR = DropBar Begin LineFormat AreaFormat [GELFRAME] End

AXM = YMult StartObject ATTACHEDLABEL EndObject

DAT = Dat Begin LD End

GELFRAME = 1*2GelFrame *Continue [PICF]

PICF = Begin PicF End

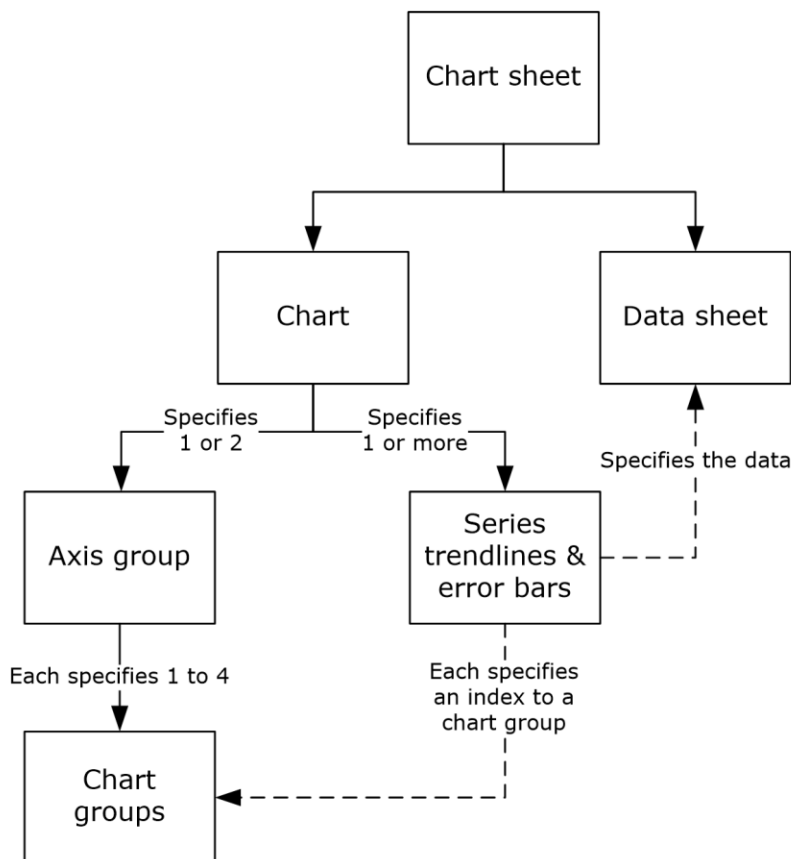
```

## 2.2 Conceptual Overview

This section specifies how high-level features of the file format are represented by combinations of records.

## 2.2.1 Charts

The following diagram identifies the major aspects of the file format representation of a chart.



**Figure 1: High-level structure of a chart**

The chart sheet specifies a chart, a graphic that displays data or the relationships between sets of data in a visual form, and a data sheet, a set of data for the chart. The chart specifies one or two axis groups, a set of axes against which the chart data is plotted, and the set of series, trendlines, and error bars specified in the chart. Each axis group specifies one to four chart groups that specify the type of visualization used to display the data. Each series, trendline, and error bar specifies a chart group with which it is associated.

### 2.2.1.1 Chart Sheet

A chart sheet is a set of data and the chart that displays the data.

A chart sheet is specified by the sequence of records that conforms to the CHARTSHEET rule, as specified by the **chart sheet** substream **ABNF**.

The following records and rules define the significant parts of a chart sheet:

- The **Fbi** and **Fbi2** records specify properties used for **font** scaling on the chart.
- The **ChartColors**, **Palette**, and **ClrtClient** records specify properties of the **color palettes** used in the chart sheet.

- The sequence of records that conforms to the DATASHEET rule specifies the data sheet that is contained in the chart sheet.
- The sequence of records that conforms to the OBJECTS rule specifies the **drawing objects** on the chart.
- The sequence of records that conforms to the CHARTFORMATS rule specifies the chart that is contained in the chart sheet.
- The **Dimensions** record specifies properties about the size of the data in the data sheet.

### 2.2.1.2 Data Sheet

A data sheet is a set of data that is displayed in row and column format and that specifies the data for a chart. In addition to specifying the chart data, the data sheet specifies properties for the visual display of the data and the grid for viewing or editing data.

A data sheet is specified by the sequence of records that conforms to the DATASHEET rule, as specified by the chart sheet substream **ABNF**.

The following records and rules define the significant parts of a data sheet:

- The **FontX** record (section [2.4.51](#)) specifies the font properties of the data sheet.
- The **ColumnWidth** records (section [2.4.29](#)) specify the width of the columns in the data sheet that have non-default width.
- The **ExcludeRows** and **ExcludeColumns** records specify the rows and columns of data that are included in or excluded from the chart.
- The **Orient** record specifies whether the series are arranged in the data sheet by rows or columns and specifies which row or column is used for the **categories** or horizontal values.
- The **Blank**, **Number**, and **Label** records specify the contents of individual **cells** in the data sheet.
- The **WinDoc**, **MainWindow**, **Window1**, **Window2Graph**, and **Selection** records specify properties of the data sheet, including location, size, **selection** on last save, and the visible range of data.
- The **LinkedSelection** record specifies the location in the data sheet to which to paste the selection from the OLE stream.

### 2.2.1.3 Chart

A chart is a graphic that displays data or the relationships between sets of data in a visual form. A chart element is an item within the chart such as an axis, legend, series, data point, data label, trendline, error bar, or data table.

A chart is specified by the sequence of records that conforms to the CHARTFORMATS rule, as specified by the chart sheet substream **ABNF**.

A chart can contain the following records and rules that specify the parts of the chart:

- The **Chart** record specifies the position and size of the **chart area**.
- The sequence of records that conforms to the FONTLIST rule specifies **font** information for the chart.
- The **Scl** record specifies the **zoom level** of the current view in the window used to display the chart.



- The **PlotGrowth** record specifies **font scaling** for the text on the chart.
- The sequence of records that conforms to the FRAME rule specifies formatting of the chart area.
- The sequence of records that conforms to the SERIESFORMAT rule specifies the series, trendlines, and error bars on the chart.
- The sequence of records that conforms to the DFTTEXT rule specifies default properties of the text in the chart. The **DataLabExt**, **StartObject**, and **EndObject** records MUST NOT exist in the sequence of records that conform to the DFTTEXT rule.
- The **AxesUsed** record and the sequence of records that conforms to the AXISPARENT rule specifies the axis groups of the chart.
- The sequence of records that conforms to the DAT rule specifies the data table for the chart.
- The sequence of records that conforms to the ATTACHEDLABEL rule specifies the chart title. This sequence of records MUST have an **ObjectLink** record in which the **wLinkObj** field has a value of 0x0001.
- The sequence of records that conforms to the ATTACHEDLABEL rule and is preceded by the optional **DataLabExt** record that specifies properties of series and data labels. This sequence of records MUST have an **ObjectLink** record in which the **wLinkObj** field has a value of 0x0004. See **Data Label** (section [2.2.1.10](#)) for additional information and restrictions on this collection of records.

#### 2.2.1.4 Axis Group

An axis group is a set of axes that specify a coordinate system, a set of chart groups that are plotted using these axes, and the **plot area** that defines where the axes are rendered on the chart.

An axis group is specified by a collection of records that conforms to the AXISPARENT rule, as specified by the chart sheet substream **ABNF**.

The following records and rules define the significant parts of an axis group:

- The **AxisParent** record specifies whether the axis group is the primary axis group or the secondary axis group on a chart. Typically, the axes of the primary axis group are displayed to the left and at the bottom of the plot area, whereas axes of the secondary axis group are displayed on the right and at the top of the plot area.
- The **Pos** record specifies the position and size of the outer plot area. The outer plot area is the bounding rectangle that includes the axis labels, the axis titles, and the data table of the chart. This record MUST be ignored on a secondary axis group.
- The sequences of records that conform to the IVAXIS, DVAXIS, and SERIESAXIS rules in the sequence of records that conform to the AXES rule specify the axes of the axis group.
- The sequence of records that conform to the ATTACHEDLABEL rule in the sequence of records that conform to the AXES rule specifies the axis titles of the axis group. Each attached label MUST contain an **ObjectLink** record that conforms to the following requirements:
  - The **wLinkObj** field MUST be equal to 0x0002, 0x0003, or 0x0007, indicating with which axis the axis title is associated.
  - The **wLinkObj** field MUST specify an axis defined in the current axis group.
  - The **wLinkObj** field MUST be unique among the other attached labels that represent axis titles in the same axis group.

- The **PlotArea** record and the sequence of records that conform to the FRAME rule in the sequence of records that conform to the AXES rule specify the properties of the inner plot area. The inner plot area is the rectangle bounded by the **chart area**. The **PlotArea** record MUST NOT exist on a secondary axis group.
- The sequence of records that conform to the CRT rule specifies the chart groups of the axis group.

Because there are many different ways to represent data visually, each representation has specific requirements about the layout of the data and the way it is plotted. This places restrictions on the combinations of chart group types that can be plotted on the same axis group and the combinations of chart group types that can be plotted in the same chart.

A chart MUST contain one of the following:

- A single axis group that contains a single chart group that contains a **Chart3d** record.
- One or two axis groups that each contain a single bubble chart group.
- One or two axis groups that each contain a single scatter chart group.
- One or two axis groups that each conform to one of the following restrictions on chart group type combinations:
  - Zero or one of each of the following chart group types: area, column, and line.
  - A single bar chart group.
  - A single bar of pie chart group.
  - A single doughnut chart group.
  - A single filled radar chart group.
  - A single pie chart group.
  - A single pie of pie chart group.
  - A single radar chart group.

In addition to the restrictions on the combinations of chart group types that can be plotted on the same axis group or chart, there are additional restrictions on the axes of the axis group based on the chart groups of the axis group.

The following restrictions apply to the axes of an axis group:

- The axis group MUST NOT contain any axes if the axis group contains a bar of pie, doughnut, pie, or pie of pie chart group.
- The axis group MUST contain a **category** or date axis if the axis group contains an area, bar, column, filled radar, line, radar, or surface chart group.
- The axis group MUST contain an area, bar, column, filled radar, line, radar, or surface chart group if the axis group contains a category or date axis.
- The axis group MUST contain two value axes if, and only if, all chart groups are of type bubble or scatter.
- The axis group MUST contain a series axis if, and only if, the chart group attached to the axis group is one of the following:
  - An area chart group with the **fStacked** field of the **Area** record set to zero.

- A column chart group with the **fStacked** field of the **Bar** record set to zero and the **fClustered** field of the **Chart3d** record set to zero.
- A line chart group with field **fStacked** of the **Line** record set to zero.
- A surface chart group.
- The chart group on the axis group MUST contain a **Chart3d** record if the axis group contains a series axis.

### 2.2.1.5 Axis

An axis is a line that borders the chart **plot area** and provides a frame of reference for measurement. In addition to the axis line and its properties, the axis also specifies all parts of the chart that are associated with the axis line, such as the axis labels, **major gridlines**, **minor gridlines**, and the **walls** and **floor** of the chart.

An axis is specified by a sequence of records that conforms to the IVAXIS, DVAXIS, or SERIESAXIS rules, as specified by the chart sheet substream **ABNF**.

An axis has a type, as described in the following table.

Type	Specified by	Description
Category	A sequence of records that conforms to the IVAXIS rule that contains an <b>AxcExt</b> record with field <b>fDateAxis</b> set to zero.	A category axis displays a set of category labels that are evenly distributed along the axis in a specified order. A category axis displays arbitrary text values such as "Qtr1", "Qtr2", and "Qtr3", and cannot display scaled numerical values.
Date	A sequence of records that conforms to the IVAXIS rule that contains an <b>AxcExt</b> record with field <b>fDateAxis</b> set to 1.	A date axis displays scaled date or time values and can display data points located at uneven intervals.
Series	A sequence of records that conforms to the SERIESAXIS rule.	A series axis displays a set of series names that are evenly distributed along the axis. When this axis is used, the data points of each series are plotted in three-dimensional space. The data points of a single series are plotted on a plane identified by the corresponding series name on this axis.
Value	A sequence of records that conforms to the DVAXIS rule.	A value axis displays scaled numeric values. The bubble and scatter chart groups, which can contain two value axes, distinguish the value axes by specifying the axis orientation, either horizontal or vertical.

The following records and rules define the significant parts of an axis:

- The **CatSerRange** and **ValueRange** records specify the scaling properties of the axis, the crossing location of the other axis in the axis group, and the direction of the axis.
- The **AxcExt** record specifies whether an axis is of type category or date and specifies properties of a date axis.
- The **CatLab** record specifies additional properties of the axis labels.
- The **IFmtRecord** record in the sequence of records that conforms to the AXS rule specifies the **number format** of the axis labels.

- The **Tick** record in the sequence of records that conforms to the AXS rule specifies properties of the axis labels and the **major tick marks** and **minor tick marks** of the axis.
- The **FontX** record in the sequence of records that conforms to the AXS rule specifies the font properties of the axis labels.
- The **AxisLine** and **LineFormat** record pairs in the sequence of records that conforms to the AXS rule specify the axis line, major gridlines and minor gridlines of the axis, and the border lines of the walls and floor of the chart. If the **AxisLine** and **LineFormat** record pair specifying the axis line are omitted, the axis line has default line format properties. If other **AxisLine** and **LineFormat** record pairs are omitted, the corresponding chart elements are omitted from the chart.
- The **AreaFormat** record and the sequence of records that conforms to the GELFRAME rule in the sequence of records that conforms to the AXS rule specify the fill format for the walls and floor of the chart. If the **wType** field of the **Axis** record in the axis equals 0x0000, these records apply to the walls of the chart. If the **wType** field of the **Axis** record in the axis is set to 0x0001, these records apply to the floor of the chart. If the **wType** field of the **Axis** record in the axis is set to 0x0002, these records MUST NOT exist. If the chart sheet substream does not contain a **Chart3d** record, these records MUST NOT exist.
- The sequence of records that conforms to the AXM rule specify the **display units** and the display units label of a value axis.

### 2.2.1.6 Chart Group

A chart group is a set of one or more series that visually represent data in a similar manner and are plotted using the same coordinate system. A chart group also includes all parts of the chart that are associated with the set of series, and the chart group can specify default properties for the data points and data labels associated with the series.

Basic charts have a single chart group because they contain one or more series of a single type and all data points are plotted using the same coordinates. Complex charts, such as combination charts that contain multiple series of different types, or multiple series of similar type that are plotted using different sets of axes, contain multiple chart groups.

A chart group specifies a collection of series of a common type that share an axis group and specifies the chart elements that are common to the collection of series.

A chart group is specified by the sequence of records that conforms to the CRT rule, as specified by the chart sheet substream **ABNF**.

A chart group has a type as described in the following table.

Type	Specified by	Description
Area	A chart group that contains an <b>Area</b> record.	A chart group type in which the data points of a series are plotted in a line, and the region between the line and the horizontal axis is filled.
Bar	A chart group that contains a <b>Bar</b> record with field <b>fTranspose</b> set to 1.	A chart group type in which the data points in a series are represented as horizontal bars.
Bar of pie	A chart group that contains a <b>BopPop</b> record with field <b>pst</b> set to 0x02.	A chart group type that plots data points as segments in a circle or bar, where the bar displays the details of the data called out from a single segment of the circle.
Bubble	A chart group that contains a <b>Scatter</b> record with field <b>fBubbles</b> set to 1.	A chart group type that is a variation on a scatter chart group type in which the data points are represented as bubbles. A bubble chart

Type	Specified by	Description
		group type compares three variables. The third variable determines the size of the data point.
Column	A chart group that contains a <b>Bar</b> record with field <b>fTranspose</b> set to zero.	A chart group type in which the data points in a series are represented as vertical bars.
Doughnut	A chart group that contains a <b>Pie</b> record with field <b>pcDonut</b> not set to 0x0000.	A chart group type in which multiple series are represented as concentric rings and the data points are represented as segments of the ring.
Filled radar	A chart group that contains a <b>RadarArea</b> record.	A chart group type that is a variation on a radar group type in which the data points in a series are connected by a line, and the area enclosed by the line is filled.
Line	A chart group that contains a <b>Line</b> record.	A chart group type in which data points in a series are connected by a line.
Pie	A chart group that contains a <b>Pie</b> record with field <b>pcDonut</b> set to 0x0000.	A chart group type that plots data points as segments (or slices) of a circle.
Pie of pie	A chart group that contains a <b>BopPop</b> record with field <b>pst</b> set to 0x01.	A chart group type that plots data points as segments (or slices) of two circles, where the secondary circle displays the details of the data called out from a single segment of the primary circle.
Radar	A chart group that contains a <b>Radar</b> record.	A chart group type in which each data point in a series is plotted along a separate axis that starts at the center of the chart and extends outward.
Scatter	A chart group that contains a <b>Scatter</b> record with field <b>fBubbles</b> set to zero.	A chart group type that displays quantitative values on both horizontal and vertical axes to represent two variables as a single data point.
Surface	A chart group that contains a <b>Surf</b> record.	A chart group type that shows a three-dimensional surface that connects a set of data points.

The following records and rules define the significant parts of a chart group:

- The **SeriesList** record specifies the series of the chart. This record MUST NOT exist in the first chart group in the chart sheet substream. This record MUST exist in subsequent chart groups in the chart sheet substream.
- The **Chart3d** record specifies that the **plot area**, axis group, and chart group are rendered in a 3-dimensional (3-D) scene, rather than a 2-dimensional scene, and specifies properties of the 3-D scene. If this record exists in the chart sheet substream, the chart sheet substream MUST have exactly one chart group. This record MUST NOT exist in a bar of pie, bubble, doughnut, filled radar, pie of pie, radar, or scatter chart group.
- The sequence of records that conforms to the LD rule specifies the legend on the chart. The sequence of records that conforms to the LD rule MUST NOT exist in a chart group that is not the first chart group in the chart sheet substream.
- The sequence of records that conform to the DROPBAR rule specifies the **up-down bars** on the chart group.
- The **CrtLine/LineFormat** record pairs specify the **drop lines**, **high-low lines**, **series lines**, and **leader lines** for the chart.

- The sequences of records that conform to the DFTTEXT rule, the **DataLabExtContents** record, and the sequence of records that conforms to the SS rule specify the data label and data point formatting for the chart group. See **Data Label** (section [2.2.1.10](#)) for details on the chart group data label.

### 2.2.1.7 Legend

A legend identifies groupings of information about the chart. A legend consists of a set of **legend entries**. Each legend entry, which consists of a **legend key** and a text label, identifies either the data points in the chart, the series and **trendlines** in the chart, or the bands on a surface chart group.

A legend is specified by a sequence of records that conforms to an LD rule in a chart group, as specified by the chart sheet substream **ABNF**.

A legend on a chart can contain three types of content:

- A legend can contain legend entries for each of the series and trendlines in the chart. In this type of legend:
  - Each legend entry represents a single series or trendline.
  - The legend keys contain the same formatting as the corresponding series or trendline.
  - The legend text labels contain the name of the corresponding series or trendline.
- This type of legend is used when the chart group type is not Surface and one of the following conditions is satisfied:
  - The chart has more than one included series.
  - The chart has a single included series, no data point formatting exceptions on the included series, and either contains a trendline or a **ChartFormat** record associated with the included series that has the **fVaried** field equal to zero or ignored.
- A legend can contain legend entries for each data point in the chart. In this type of legend:
  - Each legend entry represents a single data point on the chart.
  - The legend keys contain the same formatting as the corresponding data point.
  - The legend text labels contain the **category** name or horizontal value of the corresponding data point.

This type of legend is used when the chart group type is not Surface and the conditions for a legend that contains legend entries for each of the series and trendlines in the chart are not satisfied.

- A legend can contain legend entries for each band on a surface chart group. A surface chart group is formatted into different bands based on the value of the surface at the data point in space. In this type of legend:
  - Each legend entry represents a single band that represents a range of values on a surface chart group.
  - The legend keys contain the same formatting as the corresponding band.
  - The legend text labels contain the value range of the corresponding band.

This type of legend is used when the chart contains a surface chart group.

The following records and rules define the significant parts of a legend:

- The **Legend** record specifies the layout of the legend and whether the legend is automatically positioned.
- The **Pos** record specifies the position of the legend.
- The sequence of records that conforms to the ATTACHEDLABEL rule specifies the default text formatting for the legend entries. The **Pos** record of the attached label MUST be ignored. The **ObjectLink** record of the attached label MUST NOT exist. A series can specify formatting exceptions for individual legend entries.
- The sequence of records that conforms to the FRAME rule specifies the fill and border formatting properties of the legend.

### 2.2.1.8 Series

A series is a set of related data points that are plotted in a chart. In addition to specifying the data points of the series and the formatting properties of the data points, a series can also specify a series name and properties of the data labels and **legend entries** that are associated with the series.

A series is specified by a sequence of records that conforms to the SERIESFORMAT rule, as specified by the chart sheet substream **ABNF** that contains a **SerToCrt** record.

A series can be of type Included or Excluded. Included series are series that are shown in the chart. Excluded series are not shown in the chart, but exist as series in the file. A series is an excluded series if one of the following conditions is satisfied:

- Both of the following two conditions are satisfied:
  - The **colSeriesX** field of the **Orient** record in the chart sheet substream is greater than zero.
  - The one-based index of the **Series** record associated with this series in the collection of all **Series** records in the current chart sheet substream equals the value of the **colSeriesX** field of the **Orient** record in the chart sheet substream.
 

This case occurs when the chart group type is scatter and the first row or column of the data sheet is not used as the horizontal values of the data points. In this case, the series is a placeholder for the row or column of data in the data sheet that will become an included series on the chart if the chart group type is changed to a type other than scatter or bubble.
- The second sequence of records that conforms to the AI rule in the series contains a **BRAI** record with the **rowCol** field equal to a row or column that is excluded from the data sheet.
 

This case occurs when the series uses data that is located in a row or column that has been excluded from the data sheet.
- Both of the following two conditions are satisfied:
  - The chart group type of the series is bar of pie, pie, or pie of pie.
  - The series is not the first series in the chart sheet substream to be on the chart group and contains in the second sequence of records that conforms to the AI rule a **BRAI** record that contains a **rowCol** field equal to a row or column that is not excluded from the data sheet.
 

This case occurs when the chart group is of type bar of pie, pie, or pie of pie, and the chart sheet contains multiple series, because these chart group types only display the first series of data on the chart.

All other series are included series.

The following records and rules define the significant parts of a series:

- The **Series** record specifies the type and size of the data in the series.
- The four sequences of records that conform to the AI rule specify a reference to a row or column in a data sheet. The meaning of the row or column is specified as follows:
  - In the first collection, the value of the first **cell** in the row or column specifies the name of the series. The **SeriesText** record, when in this collection, specifies a cache of the series name.
  - In the second collection, the row or column, excluding the first cell and all excluded cells of the data sheet, specifies a set of coordinates along the vertical value axis if the series is in a bubble or scatter chart group, or a set of coordinates along the value axis if the series is in any other chart group type.
  - In the third collection, the row or column, excluding the first cell and all excluded cells of the data sheet, specifies a set of coordinates along a horizontal value axis if the series is in a bubble or scatter chart group, a set of category (3) axis labels that are distributed evenly along the category (3) axis based on their order in the set, or a set of dates along a date axis in any other chart group type.
 

The **rowCol** field of the **BRAI** record in this third collection of records MUST equal the **colSeriesX** field of the **Orient** record in the chart sheet substream.
  - In the fourth collection, the row or column, excluding the first cell and all excluded cells, specifies a set of scalar values used in a bubble chart group to define the size of the bubbles. The **rowCol** field of the **BRAI** record in this fourth collection of records MUST be set to 0x0000 if the chart group type is not bubble because these values are not required in the representation of the series in non-bubble chart group types.
- The sequence of records that conform to the SS rule specifies the data point and data label properties for a series or individual data points of the series. If formatting is not specified for an individual data point, the data point inherits the formatting of the series. If formatting is not specified for the series, the series inherits the formatting of the chart group that contains the series. The **yi** field of the **DataFormat** record MUST specify the zero-based index of the **Series** record associated with this series in the collection of all **Series** records in the chart sheet substream that contains the series. See **Data Label** (section [2.2.1.10](#)) for an explanation of the data label and the conditions on this collection of records.
- The **SerToCrt** record specifies the chart group that contains this series.
- The **LegendException** record specifies a legend entry in the legend that corresponds to the series. If the **LegendException** record specifies that the legend entry has non-default formatting, the attached label that follows the **LegendException** and **Begin** records specifies the custom formatting of the legend entry. If the **fLabel** field of **LegendException** equals zero, the attached label MUST NOT exist.

### 2.2.1.9 Data Point

A data point is a value plotted in a chart and visually displayed as a shape, such as a bar, column, or marker, as specified by the chart group type of the chart.

A data point consists of a set of three values located on the same index of the second to fourth sequence of records that conform to the AI rule contained in the series of the data point. For example, the set of three values for a data point in a bubble chart group consists of the coordinate of the data point along the vertical value axis, the coordinate of the data point along the horizontal value axis, and the scalar value that defines the **bubble size** of the data point.



### 2.2.1.10 Data Label

A data label is a label on a chart that is associated with a data point, or with a series on an area or filled radar chart group. A data label contains information about the associated data point, such as the description of the data point, a **legend key**, or custom text.

#### Inheritance

The following is the order of inheritance that determines the contents of a data label associated with the data point:

- Data labels can be specified for a chart group, specifying the default setting for the data labels associated with the data points on the chart group.
- Data labels can be specified for a series, specifying the default setting for the data labels associated with the data points of the series. This type of data label overrides the data label properties specified on the chart group for the data labels associated with the data points in a series.
- Data labels can be specified for a data point, specifying the settings for a data label associated with that data point. This type of data label overrides the data label properties specified on the chart group and series for the data labels associated with that data point.

#### Records

The set of records that specifies a data label, and the requirements that exist on these records, differ depending on whether the data label is specified for a chart group, series, or data point. The set of records that represent a data label are as follows:

- For a chart group, properties of a data label are specified by the following collections of records that are specified in the chart group:
  - The sequence of records that conforms to the DFTTEXT rule.
  - The **DataLabExtContents** record that is not contained in the sequence of records that conforms to the LD or the sequence of records that conforms to the DFTTEXT rule.
  - The **AttachedLabel** record that is contained in the sequence of records that conforms to the SS rule.
- For a series or data point, properties of a data label are specified by the following collections of records that are specified in the chart:
  - The attached label that corresponds to the series or data point. The attached label appears in the chart sheet substream after the axis group and is specified by the following properties of the attached label:
    - The **wLinkObj** field of the **ObjectLink** record is set to 0x0004.
    - The **wLinkVar1** field of the **ObjectLink** record equals the index to the corresponding series.
    - The **wLinkVar2** field of the **ObjectLink** record equals the index to the corresponding data point or is set to 0xFFFF for a corresponding series.
  - The **AttachedLabel** record that corresponds to the series or data point. The **AttachedLabel** record appears in the sequence of records that conforms to the SS rule that has the following properties:
    - The **yi** field of the **DataFormat** record equals the index to the corresponding series.

- The **xi** field of the **DataFormat** record equals the index to the corresponding data point or is set to 0xFFFF for a corresponding series.

## Overrides

Some properties of the records that specify the contents of a data label can overlap and conflict. For the information that overlaps between these records, there is a set of rules that specifies the relationships between these records and the fields to be ignored in conflict situations.

In general, properties of the **DataLabExtContents** record of the data label override properties of the **Text** record of the of the data label, which overrides the properties of the **AttachedLabel** record of the data label. Details of the relationships between individual fields are specified in the records.

The following section provides an explanation of how to interpret the data label and when the different records that specify the data label are relevant to the data label.

On a data label associated with a chart group:

- When the chart group has a data label, the following algorithm determines the data label contents. Once the set of records that represents the data label has been established, the data label information that overlaps across records is respected on the **DataLabExtContents** record and ignored on the **Text** and **AttachedLabel** records of the data label. In this algorithm, **ABNF** rules are used to specify the sequence of records that conform to the rule. In this algorithm, "DFTTEXT with id" specifies the sequence of records that conforms to the DFTTEXT rule that contains a **DefaultText** record with an **id** field equal to the value specified.

```
//Define variables
X equals a DataLabExtContents record
Y equals a DataLabExtContents record
SS equals an to AttachedLabel record
SWAP equals a Boolean

//Initialize SWAP
SET SWAP equal to FALSE

//Initialize SS
SET SS equal to AttachedLabel record in the sequence of records that conforms to
the SS rule

//Initialize X
IF DFTTEXT with id field equal to 0 exists
  IF DFTTEXT with id field equals to 0 has a DataLabExtContents record
    SET X equal to DataLabExtContents record in DFTTEXT with id field equal to
    0
  ELSE
    IF chart group type equals area or filled radar
      SET field fSerName of X equal to field fShowLabel of Text record in
      DFTTEXT with id field equal to 0
      SET field fCatName of X equal to 0
    ELSE
      SET field fCatName of X equal to field fShowLabel of Text record in
      DFTTEXT with id field equal to 0
      SET field fSerName of X equal to 0
    END IF
    SET field fValue of X equal to field fShowValue of Text record in DFTTEXT
    with id field equal to 0
    SET field fPercent of X equal to field (fShowPercent OR fShowLabelAndPerc)
    of Text record in DFTTEXT with id field equal to 0
    SET field fBubSizes of X equal to field fShowBubbleSizes of Text record in
    DFTTEXT with id field equal to 0
    SET field RgchSet of X equal to NULL
  END IF
ELSE
  SET field fCatName of X equal 1
  SET fields fSerName, fValue, fPercent, fBubSizes of X equal to 0
```

```

        SET field RgchSet of X equal to NULL
    END IF

//Initialize Y
IF DFTTEXT with id field equal to 1 exists
    IF DFTTEXT with id field equals to 1 has a DataLabExtContents record
        SET Y equal to DataLabExtContents record in DFTTEXT with id field equal to
        1
    ELSE
        IF chart group type equals area or filled radar
            SET field fSerName of Y equal to field (fShowLabel OR
                fShowLabelAndPerc) of Text record in DFTTEXT with id field equal to
                1
            SET field fCatName of Y equal to 0
        ELSE
            SET field fCatName of Y equal to field (fShowLabel OR
                fShowLabelAndPerc) of Text record in DFTTEXT with id field equal to
                1
            SET field fSerName of Y equal to 0
        END IF
        SET field fValue of Y equal to field fShowValue of Text record in DFTTEXT
        with id field equal to 1
        SET field fPercent of Y equal to field (fShowPercent OR fShowLabelAndPerc)
        of Text record in DFTTEXT with id field equal to 1
        SET field fBubSizes of Y equal to field fShowBubbleSizes of Text record in
        DFTTEXT with id field equal to 1
        SET field RgchSet of X equal to NULL
    END IF
ELSE
    SET field fValue of Y equal to 1
    SET fields fCatName, fSerName, fPercent, fBubSizes of Y equal to 0
    SET field RgchSet of X equal to NULL
END IF

//Modify X, Y, and SS
IF (field fCatName of X equals 1) AND (fields fSerName, fValue, fPercent, AND
    fBubSizes of X equal 0)
    IF (field fShowValue of SS equals 1) AND (fields fShowPercent,
        fShowLabelAndPerc, fShowLabel, fShowBubbleSizes, AND fShowSeriesName of
        SS equal 0)
        IF (field fValue of Y equals 1) AND (fields fSerName, fCatName,
            fPercent, OR fBubSizes of Y equal 1)
            SET SWAP equals to TRUE
            SET field fValue of X equal to 1
            SET fields fSerName, fCatName, fPercent, AND fBubSizes of X equal
            to 0
            SET field fShowValue of SS equal to field fValue of Y
            SET field fShowPercent of SS equal to field fSPercent of Y
            SET field fShowLabel of SS equal to field fCatName of Y
            SET field fShowBubbleSizes of SS equal to field fBubSizes of Y
            SET field fShowSeriesName of SS equal to field fSerName of Y
        END IF
    ELSE
        IF ((field fShowPercent of SS equals 1) AND (field fShowBubbleSizes of
            SS equals 0)) OR ((field fShowPercent of SS equals 0) AND (field
            fShowBubbleSizes of SS equals 1)) AND (fields fShowLabelAndPerc,
            fShowLabel, fShowValue, AND fShowSeriesName of SS equal 0)
            SET SWAP equals to TRUE
            SET field fValue of X equal to 1
            SET fields fSerName, fCatName, fPercent, AND fBubSizes of X equal
            to 0
            SET field fShowValue of SS equal to field fValue of Y
            SET field fShowPercent of SS equal to field fSPercent of Y
            SET field fShowLabel of SS equal to field fCatName of Y
            SET field fShowBubbleSizes of SS equal to field fBubSizes of Y
            SET field fShowSeriesName of SS equal to field fSerName of Y
        ELSE
            IF (field fShowLabel of SS equals 1) AND (fields fShowPercent,

```

```

fShowLabelAndPerc, fShowValue, fShowBubbleSizes, AND
fShowSeriesName of SS equal 0)
IF chart group type equals area or filled radar
    SET field fSerName of X equal to 1
    SET fields fCatName, fValue, fPercent, AND fBubSizes equal
    to 0
    SET field fShowValue of SS to 1
    SET fields fShowPercent, fShowLabelAndPerc, fShowLabel,
    fShowBubbleSize, and fShowSeriesName of SS to 0
END IF
SET field fValue of Y equals to 1
SET fields fSerName, fCatName, fPercent, AND fBubSizes of Y
equal to 0
ELSE
    IF fields fShowLabel, fShowPercent, fShowLabelAndPerc,
    fShowValue, fShowBubbleSizes, AND fShowSeriesName of SS
    equal 0
    IF chart group type equals area or filled radar
        SET fields fShowLabel of SS to 1
        SET fields fShowPercent, fShowLabelAndPerc, fShowValue,
        fShowBubbleSize, and fShowSeriesName of SS to 0
    END IF
    SET field fValue of Y equals to 1
    SET fields fSerName, fCatName, fPercent, AND fBubSizes of Y
    equal to 0
    END IF
END IF
END IF
END IF
ELSE
    IF (field fValue of Y equals 1) AND (fields fSerName, fCatName, fPercent,
    AND fBubSizes of Y equal 0)
    IF fields fShowLabel, fShowPercent, fShowLabelAndPerc, fShowValue,
    fShowBubbleSizes, AND fShowSeriesName of SS equal 0
    IF (chart group type equals area or filled radar) AND (field
    fSerName of X equals 1) AND (fields fCatName, fValue, fPercent,
    AND fBubSizes of X equal 0)
    SET field fShowSeriesName of SS to 1
    SET fields fShowPercent, fShowLabelAndPerc, fShowValue,
    fShowBubbleSize, and fShowLabel of SS to 0
    ELSE
    SET field fCatName of X equals to 1
    SET fields fSerName, fValue, fPercent, AND fBubSizes of X equal
    to 0
    END IF
    ELSE
    IF (field fShowValue of SS equals 1) AND (fields fShowLabel,
    fShowPercent, fShowLabelAndPerc, fShowBubbleSizes, AND
    fShowSeriesName of SS equal 0)
    SET field fCatName of X equals to 1
    SET fields fSerName, fValue, fPercent, AND fBubSizes of X equal
    to 0
    ELSE
    SET field fShowValue of SS equal to field fValue of X
    SET field fShowPercent of SS equal to field fSPercent of X
    SET field fShowLabel of SS equal to field fCatName of X
    SET field fShowBubbleSizes of SS equal to field fBubSizes of X
    SET field fShowSeriesName of SS equal to field fSerName of X
    END IF
    END IF
    END IF
    END IF
    //Determine if X or Y is used
    IF (field fShowValue of SS equals 1) AND (fields fShowLabel, fShowPercent,
    fShowLabelAndPerc, fShowBubbleSizes, AND fShowSeriesName of SS equal 0)
    IF (SWAP equals FALSE)
    IF DFTTEXT with id field equal to 1 exists
    SET the data label equal to DFTTEXT with id field equal to 1

```

```

ELSE
    SET the data label equal to the default formatting properties
END IF
SET the DataLabExtContents record of the data label equal to Y
ELSE
    IF DFTTEXT with id field equal to 0 exists
        SET the data label equal to DFTTEXT with id field equal to 0
    ELSE
        SET the data label equal to the default formatting properties
    END IF
    SET the DataLabExtContents record of the data label equal to X
END IF
ELSE
    IF (SWAP equals FALSE)
        IF DFTTEXT with id field equal to 0 exists
            SET the data label equal to DFTTEXT with id field equal to 0
        ELSE
            SET the data label equal to the default formatting properties
        END IF
        SET the DataLabExtContents record of the data label equal to X
    ELSE
        IF DFTTEXT with id field equal to 1 exists
            SET the data label equal to DFTTEXT with id field equal to 1
        ELSE
            SET the data label equal to the default formatting properties
        END IF
        SET the DataLabExtContents record of the data label equal to Y
    END IF
END IF

```

On a data label associated with a series or data point:

- The attached label specifies an exception on the data label and indicates that the current data label differs from the data label inherited from the chart group for a series, or from the chart group and series for a data point. The attached label of the data label contains a **Text** record and can contain a **DataLabExtContents** record.
- On a data label associated with a data point, when the attached label exists and contains a **SeriesText** record and a **BRAI** record with field **formula** set to 0x0000, the **SeriesText** record specifies the contents of the data label.
- When the attached label exists and contains a **DataLabExtContents** record, the data label information specified by the **DataLabExtContents** record that overlaps across records is respected, and the information contained in the **Text** and **AttachedLabel** records of the data label that overlaps is ignored.
- When the attached label exists and does not contain a **DataLabExtContents** record, the data label specified by the **Text** record that overlaps across records is respected, and the information contained in the **AttachedLabel** record of the data label that overlaps is ignored.
- When the data label does not specify an attached label, the **AttachedLabel** record of the data label specifies the content properties of the data label.

### Restrictions

The following requirements and restrictions exist on parts of the collections of records that represent the data label.

If the chart group contains a data label:

- The chart group SHOULD<3> contain two sequences of records that conform to the DFTTEXT rule that have the same records with the same field values.

- If the chart group contains a **DataLabExtContents** record, the **DataLabExtContents** record in the sequences of records that conform to the DFTTEXT rule on the chart group and the **DataLabExtContents** record that is not specified in the sequences of records that conform to the DFTTEXT rule on the chart group SHOULD<4> contain the same field values.
- If the chart group contains a sequence of records that conforms to the DFTTEXT rule, the following restrictions apply:
  - The **fShowValue** field of the **AttachedLabel** record SHOULD<5> equal the **fShowValue** field of the **Text** record in the sequence of records that conforms to the DFTTEXT rule.
  - The **fShowPercent** field of the **AttachedLabel** record SHOULD<6> equal the **fShowPercent** field of the **Text** record in the sequence of records that conforms to the DFTTEXT rule.
  - The **fShowLabelAndPerc** field of the **AttachedLabel** record SHOULD<7> equal the **fShowLabelAndPerc** field of the **Text** record in the sequence of records that conforms to the DFTTEXT rule.
  - The **fShowLabel** field of the **AttachedLabel** record SHOULD<8> equal the **fShowLabel** field of the **Text** record in the sequence of records that conforms to the DFTTEXT rule.
  - The **fShowBubbleSizes** field of the **AttachedLabel** record SHOULD<9> equal the **fShowBubbleSizes** field of the **Text** record in the sequence of records that conforms to the DFTTEXT rule.
  - If the chart group contains a **DataLabExtContents** record and the chart group type is not area or filled radar, the **fShowSeriesName** field of the **AttachedLabel** record SHOULD<10> equal the **fSerName** field of the **DataLabExtContents** record. Otherwise, the **fShowSeriesName** field of the **AttachedLabel** record SHOULD<11> be set to zero.
  - The **fAutoText** field of the **Text** record in the sequence of records that conforms to the DFTTEXT rule MUST be set to 1.

The **DataLabExtContents** record MUST exist on a data label if, and only if, at least one of the following conditions is satisfied:

- **Non-Area Series Only:** All of the following conditions are satisfied:
  - The chart group type of the associated chart group, series, or data point is not area or filled radar.
  - The data label shows only the series name, and the **DataLabExtContents** record has the following state:
    - **fSerName** is set to 1.
    - **fBubSizes**, **fCatName**, **fPercent**, and **fValue** are set to zero.
- **Area Category Only:** All of the following conditions are satisfied:
  - The chart group type of the associated chart group, series, or data point is area or filled radar.
  - The data label only shows the **category** name, and the **DataLabExtContents** record has the following state:
    - **fCatName** is set to 1.
    - **fBubSizes**, **fPercent**, **fSerName**, and **fValue** is set to zero.
- **Label and Percent with Non-Null Separator:** All of the following conditions are satisfied:

- The data label shows only the label and percent, and the **DataLabExtContents** record has the following state:
  - **fCatName** and **fPercent** is set to 1.
  - **fBubSizes**, **fSerName**, and **fValue** is set to zero.
- The data label has a non-NULL separator, and the **rgchSep** field of the **DataLabExtContents** record is not equal to NULL.
- **Combo Other Than Label and Percent**: All of the following conditions are satisfied:
  - The data label shows more than one type of content, other than label and percent, and the fields of the **DataLabExtContents** record satisfy at least one of the following conditions:
    - **fCatName** or **fPercent** is set to 1, and **fBubSizes**, **fSerName**, or **fValue** is set to 1.
    - **fBubSize** equals 1, and **fSerName** or **fValue** is set to 1.
    - **fSerName** and **fValue** is set to 1.

The optional **DataLabExt**, **StartObject**, and **EndObject** records MUST exist in the chart sheet substream if, and only if, the attached label contained between the **StartObject** and **EndObject** records satisfies one of the following conditions:

- The Non-Area Series Only case specified previously.
- The Area Category Only case specified previously.

### 2.2.1.11 Trendline

A trendline is a straight or curved line that graphically represents the general trend of the data points of a series. In addition to specifying the data and formatting properties of the line, the trendline name and the trendline label can also be specified in the trendline.

A trendline is specified by a sequence of records that conforms to the SERIESFORMAT rule, as specified by the chart sheet substream **ABNF** that contains a **SerAuxTrend** record.

The following records and rules define the significant parts of a trendline:

- The **SeriesText** record in the first sequence of records that conforms to the AI rule specifies a custom trendline name to display in the legend.
- The sequence of records that conforms to the SS rule specifies the formatting properties of the trendline. The **yi** field of the **DataFormat** record MUST specify the zero-based index of the **Series** record associated with this trendline in the collection of all **Series** records in the current chart sheet substream. Each trendline MUST contain zero or one sequences of records that conform to the SS rule.
- The **SerParent** record specifies the series associated with the trendline.
- The **SerAuxTrend** record specifies properties of the trendline, such as the trendline type and the number of periods to **forecast** forward and backward, and specifies properties of the content of the trendline label.
- The **LegendException** record specifies the formatting of the **legend entry** in the legend that corresponds to the trendline. This record is specified if the legend entry has been deleted or does not use the default formatting of the legend. When the legend entry has non-default formatting, the attached label that follows the **LegendException** and **Begin** records specifies the custom formatting of the legend entry.

The following restrictions apply to a trendline:

- All fields except the **cValx** and **cValy** fields of the **Series** record MUST be ignored. The **cValy** and **cValx** fields of the **Series** record MUST be greater than zero.
- The **BRAI** records in the sequences of records that conform to the AI rule MUST be ignored, and the **rowCol** field of the **BRAI** records MUST be set to 0x0000.

#### 2.2.1.12 Error Bar

An error bar is a set of lines displayed on a chart that indicates a range of uncertainty in the measurement of each data point in a series.

An error bar is specified by a sequence of records that conforms to the SERIESFORMAT rule, as specified by the chart sheet substream **ABNF** that contains a **SerAuxErrBar** record.

The following records and rules define the significant parts of an error bar:

- The sequence of records that conforms to the SS rule specifies the formatting properties of the error bar. The **yi** field of the **DataFormat** record MUST specify the zero-based index of the **Series** record associated with this error bar in the collection of all **Series** records in the current chart sheet substream. Each error bar MUST contain zero or one sequence of records that conform to the SS rule.
- The **SerParent** record specifies the series associated with the error bar.
- The **SerAuxErrBar** record specifies properties of the error bar, including the direction and type of the error bar.

The following restrictions apply to error bars:

- All fields except the **cValx** and **cValy** fields in the **Series** record MUST be ignored. The **cValx** and **cValy** fields MUST equal the **cValx** and **cValy** fields of the of the **Series** record in the associated series specified by the **SerParent** record.
- The **BRAI** records in the sequences of records that conform to the AI rule MUST be ignored, and the **rowCol** field of the **BRAI** records MUST be set to 0x0000.
- The **LegendException** record in the chart sheet substream MUST NOT exist.

#### 2.2.1.13 Data Table

A data table is a table on a chart that contains a row for each series and lists the values of each data point on the chart.

A data table is specified by a sequence of records that conforms to a DAT rule, as specified by the chart sheet substream **ABNF**.

The following records and rules define the significant parts of a data table:

- The **Dat** record specifies whether the data table shows **legend keys** next to the name of the series and specifies which data table borders are displayed.
- The sequence of records that conforms to the ATTACHEDLABEL rule in the sequence of records that conforms to the LD rule specifies the text formatting for the data table. The **Pos** record of the attached label MUST be ignored. The **ObjectLink** record of the attached label MUST NOT exist.
- The sequence of records that conforms to the FRAME rule in the sequence of records that conforms to the LD rule specifies the formatting properties of the data table.



The following restrictions apply to the collection of records that represents a data table:

- The **fWasDataTable** field of the **Legend** record in the sequence of records that conforms to the LD rule MUST be set to 1.
- The **Pos** record in the sequence of records that conforms to the LD rule MUST be ignored. A data table is positioned automatically.

A data table is not displayed on a chart if the chart contains a bar of pie, bubble, doughnut, filled radar, pie, pie of pie, radar, or scatter chart group.

#### 2.2.1.14 Attached Label

An attached label is a generic text element of a chart. An attached label can specify properties of an axis title, chart title, data label, data table, **display units** label, legend, **legend entry**, trendline label, or the default text properties of the entire chart.

The location of the attached label in the chart sheet substream **ABNF** and the properties of the **ObjectLink** record of the attached label specify the chart element to which the attached label applies. See the conceptual overviews for explanation of the meaning of the attached label in each context that it is used.

An attached label is specified by a sequence of records that conforms to an ATTACHEDLABEL rule, as specified by the chart sheet substream ABNF.

- The **Text** record specifies properties of the text in the attached label. These properties include text options, such as text rotation and reading order. On attached labels that represent data labels, the **Text** record also specifies content and layout properties of the data labels.
- The **Pos** record specifies the position of the attached label on the chart. The **Pos** record MUST be ignored if the attached label represents the data labels on a series or chart group, a data table, the default text of the chart, or a **legend key**.
- The **FontX** and **AlRuns** records specify the font and rich text formatting properties of the attached label.
- The sequence of records that conforms to the AI rule specifies custom text of the attached label.
- The sequence of records that conforms to the FRAME rule specifies the fill and border properties of the attached label.
- The **ObjectLink** record specifies the chart element to which the attached label applies.
- The **DataLabExtContents** record specifies additional data label properties on attached labels that represent a data label. See **Data Label** (section [2.2.1.10](#)) for the conditions on when this record can be written. This record MUST NOT exist unless the attached label specifies properties of a data label. An attached label specifies properties of a data label if the **wLinkObj** field of the **ObjectLink** record is set to 0x0004 or the attached label is in the sequence of records that conforms to the DFTTEXT rule in a chart group.

#### 2.2.1.15 SPRC

A SPRC is a unit of measurement that is 1/4000 of the height or width of the chart. If the field is being used to specify a width or horizontal distance, the SPRC is 1/4000 of the width of the chart. If the field is being used to specify a height or vertical distance, the SPRC is 1/4000 of the height of the chart.

## 2.3 Record Enumeration

This section specifies the record name associated with a record type value. For more information about record types, see **Record** (section [2.1.4](#)). These associations between record name and record type are listed by record name as well as by record type.

The type-specific meaning and fields for each record type are specified in **Records** (section [2.4](#)).

### 2.3.1 By Name

Name	Record type (number)
AlRuns	4176
Area	4122
AreaFormat	4106
AttachedLabel	4108
AxcExt	4194
AxesUsed	4166
Axis	4125
AxisLine	4129
AxisParent	4161
BOF	2057
BOFDatasheet	4178
BRAI	4177
Bar	4119
Begin	4147
Blank	1
BopPop	4193
BopPopCustom	4199
BoundSheet8	133
CatLab	2134
CatSerRange	4128
Chart	4098
Chart3DBarShape	4191
Chart3d	4154
ChartColors	684
ChartFormat	4116
ChartFrtInfo	2128

<b>Name</b>	<b>Record type (number)</b>
ClrtClient	4188
CodePage	66
ColumnWidth	36
Continue	60
Country	140
CrtLine	4124
CrtLink	4130
Dat	4195
DataFormat	4102
DataLabExt	2154
DataLabExtContents	2155
Date1904	34
DefaultText	4132
Dimensions	512
DropBar	4157
EOF	10
End	4148
EndBlock	2131
EndObject	2133
ExcludeColumns	4180
ExcludeRows	4179
Fbi	4192
Fbi2	4200
Font	49
FontX	4134
Format	1054
Frame	4146
FrtFontList	2138
GelFrame	4198
IFmtRecord	4174
Label	516

<b>Name</b>	<b>Record type (number)</b>
Legend	4117
LegendException	4163
Line	4120
LineFormat	4103
LinkedSelection	4190
MainWindow	4185
MarkerFormat	4105
MaxStatus	4184
MsoDrawing	235
MsoDrawingGroup	236
MsoDrawingSelection	237
Number	3
ObjectLink	4135
Orient	4181
Palette	146
PicF	4156
Pie	4121
PieFormat	4107
PlotArea	4149
PlotGrowth	4196
Pos	4175
Radar	4158
RadarArea	4160
Scatter	4123
Scl	160
Selection	29
SerAuxErrBar	4187
SerAuxTrend	4171
SerFmt	4189
SerParent	4170
SerToCrt	4165

<b>Name</b>	<b>Record type (number)</b>
Series	4099
SeriesList	4118
SeriesText	4109
ShtProps	4164
StartBlock	2130
StartObject	2132
Surf	4159
Text	4133
Tick	4126
TxO	438
Units	4097
ValueRange	4127
WinDoc	4183
Window1	61
Window1_10	61
Window2Graph	62
YMult	2135

### 2.3.2 By Number

<b>Name</b>	<b>Record type (number)</b>
Blank	1
Number	3
EOF	10
Selection	29
Date1904	34
ColumnWidth	36
Font	49
Continue	60
Window1	61
Window1_10	61

<b>Name</b>	<b>Record type (number)</b>
Window2Graph	62
CodePage	66
BoundSheet8	133
Country	140
Palette	146
Scl	160
MsoDrawing	235
MsoDrawingGroup	236
MsoDrawingSelection	237
TxO	438
Dimensions	512
Label	516
ChartColors	684
Format	1054
BOF	2057
ChartFrtInfo	2128
StartBlock	2130
EndBlock	2131
StartObject	2132
EndObject	2133
CatLab	2134
YMult	2135
FrtFontList	2138
DataLabExt	2154
DataLabExtContents	2155
Units	4097
Chart	4098
Series	4099
DataFormat	4102
LineFormat	4103
MarkerFormat	4105

<b>Name</b>	<b>Record type (number)</b>
AreaFormat	4106
PieFormat	4107
AttachedLabel	4108
SeriesText	4109
ChartFormat	4116
Legend	4117
SeriesList	4118
Bar	4119
Line	4120
Pie	4121
Area	4122
Scatter	4123
CrtLine	4124
Axis	4125
Tick	4126
ValueRange	4127
CatSerRange	4128
AxisLine	4129
CrtLink	4130
DefaultText	4132
Text	4133
FontX	4134
ObjectLink	4135
Frame	4146
Begin	4147
End	4148
PlotArea	4149
Chart3d	4154
PicF	4156
DropBar	4157
Radar	4158

<b>Name</b>	<b>Record type (number)</b>
Surf	4159
RadarArea	4160
AxisParent	4161
LegendException	4163
ShtProps	4164
SerToCrt	4165
AxesUsed	4166
SerParent	4170
SerAuxTrend	4171
IFmtRecord	4174
Pos	4175
AlRuns	4176
BRAI	4177
BOFDatasheet	4178
ExcludeRows	4179
ExcludeColumns	4180
Orient	4181
WinDoc	4183
MaxStatus	4184
MainWindow	4185
SerAuxErrBar	4187
ClrtClient	4188
SerFmt	4189
LinkedSelection	4190
Chart3DBarShape	4191
Fbi	4192
BopPop	4193
AxcExt	4194
Dat	4195
PlotGrowth	4196
GelFrame	4198



Name	Record type (number)
BopPopCustom	4199
Fbi2	4200

## 2.4 Records

### 2.4.1 AIRuns

This record specifies rich text formatting in chart titles, trendline, and data labels.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
cRuns										rgRuns (variable)																					
...																															

**cRuns (2 bytes):** An unsigned integer that specifies the number of rich text runs. MUST be greater than or equal to 0x0003 and less than or equal to 0x0100.

**rgRuns (variable):** An array of **FormatRun** structures (section 2.5.7) that specifies the rich text runs. The number of elements in the array MUST be equal to **cRuns**.

### 2.4.2 Area

This record specifies that the chart group is an area chart group and specifies the chart group attributes.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
A	B	C	reserved																												

**A - fStacked (1 bit):** A bit that specifies whether the data points in the chart group that share the same **category** are stacked.

**B - f100 (1 bit):** A bit that specifies whether the data points in the chart group are displayed as a percentage of the sum of all data points in the chart group that share the same category. MUST be 0 if **fStacked** is 0.

**C - fHasShadow (1 bit):** A bit that specifies whether one or more data points in the chart group has shadows.

**reserved (13 bits):** MUST be zero, and MUST be ignored.

### 2.4.3 AreaFormat


This record specifies the patterns and colors used in a filled region of a chart. If this record is not present in the SS rule of the chart sheet substream **ABNF**, all fields have default values. Otherwise, all fields MUST contain a value.











0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rgbFore																															
rgbBack																															
fls																A	B	reserved													
icvFore																icvBack															













**rgbFore (4 bytes):** A **LongRGB** structure (section 2.5.20) that specifies the **foreground color** of the **fill pattern**.<12> The default value of this field is selected automatically from the next available color in the Chart color table.














**rgbBack (4 bytes):** A **LongRGB** structure (section 2.5.20) that specifies the **background color** of the fill pattern.<13> The default value of this field is 0x00FFFFFF.









**fls (2 bytes):** An unsigned integer that specifies the type of fill pattern. If this field is set neither to 0x0000 nor to 0x0001, this record **MUST** be followed immediately by a **GelFrame** record that specifies the fill pattern. The **fillType** (as specified in [MS-ODRAW] section 2.3.7.1) of the **OPT1** field of the corresponding **GelFrame** record **MUST** be **msofillPattern** as specified in [MS-ODRAW] section 2.4.11. The default value of this field is 0x0001. This field **MUST** be set to a value from the following table.

Value	Description
0x0000	The fill pattern is None (no <b>fill</b> ). When <b>rgbFore</b> or <b>rgbBack</b> are specified, a pattern of None overrides both of those values and means that there is no fill.
0x0001	The fill pattern is solid. When solid is specified, <b>rgbFore</b> is the only color rendered, even if <b>rgbBack</b> is specified.
0x0002	The fill pattern is medium gray. Additional properties in the corresponding GelFrame record specify which of the following gray patterns is used.   <b>Percent50:</b> Specifies a 50 percent hatch. The ratio of foreground color to background color is 50:100.
0x0003	The fill pattern is dark gray. Additional properties in the corresponding GelFrame record specify which of the following dark gray patterns is used.  <b>Percent60:</b> Specifies a 60 percent hatch. The ratio of foreground color to background color is 60:100.  <b>Percent70:</b> Specifies a 70 percent hatch. The ratio of foreground color to background color is 70:100.  <b>Percent75:</b> Specifies a 75 percent hatch. The ratio of foreground color to background color is 75:100.  <b>Percent80:</b> Specifies an 80 percent hatch. The ratio of foreground color to background color is 80:100.

Value	Description
	 <b>Percent90</b> : Specifies a 90 percent hatch. The ratio of foreground color to background color is 90:100.
0x0004	<p>The fill pattern is light gray. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following light gray patterns is used.</p>  <b>Percent25</b> : Specifies a 25 percent hatch. The ratio of foreground color to background color is 25:100.
0x0005	<p>The fill pattern is horizontal stripes. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following horizontal stripe pattern is used.</p>  <b>DarkHorizontal</b> : Specifies a pattern of horizontal lines.
0x0006	<p>The fill pattern is vertical stripes. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following vertical stripes patterns is used.</p>  <b>DarkVertical</b> : Specifies a pattern of vertical lines.
0x0007	<p>The fill pattern is downward diagonal stripes. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following diagonal stripes patterns is used:</p>  <b>DarkDownwardDiagonal</b> : Specifies diagonal lines that slant to the right from top to bottom. This hatch pattern is not anti-aliased.  <b>WideDownwardDiagonal</b> : Specifies diagonal lines that slant to the right from top to bottom, are 1.5 times the width of <b>DarkDownwardDiagonal</b> , but are not anti-aliased.
0x0008	<p>The fill pattern is upward diagonal stripes. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following diagonal stripes patterns is used:</p>  <b>DarkUpwardDiagonal</b> : Specifies diagonal lines that slant to the left from top to bottom, but the lines are not anti-aliased.  <b>WideUpwardDiagonal</b> : Specifies diagonal lines that slant to the left from top to bottom, are 1.5 times the width of <b>DarkUpwardDiagonal</b> , but are not anti-aliased.
0x0009	<p>The fill pattern is grid. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following grid patterns is used:</p>  <b>LargeCheckerBoard</b> : Specifies a hatch that has the appearance of a checkerboard with squares that are twice the size of <b>SmallCheckerBoard</b> .  <b>Plaid</b> : Specifies a hatch that has the appearance of a plaid material.

Value	Description
	 <b>SmallCheckerBoard</b> : Specifies a hatch that has the appearance of a checkerboard.  <b>SolidDiamond</b> : Specifies a hatch that has the appearance of a checkerboard placed diagonally.  <b>Sphere</b> : Specifies a hatch that has the appearance of spheres laid adjacent to one another.
0x000A	<p>The fill pattern is a trellis. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following trellis patterns is used:</p>  <b>Trellis</b> : Specifies a hatch that has the appearance of a trellis.
0x000B	<p>The fill pattern is light horizontal stripes. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following light horizontal stripes patterns is used:</p>  <b>DashedHorizontal</b> : Specifies dashed horizontal lines.  <b>LightHorizontal</b> : Specifies a pattern of solid horizontal lines.  <b>NarrowHorizontal</b> : Specifies horizontal lines that are spaced 25 percent closer together than <b>LightHorizontal</b> .
0x000C	<p>The fill pattern is light vertical stripes. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following light vertical stripes patterns is used:</p>  <b>DashedVertical</b> : Specifies dashed vertical lines.  <b>LightVertical</b> : Specifies a pattern of solid vertical lines.  <b>NarrowVertical</b> : Specifies vertical lines that are spaced 25 percent closer together than <b>LightVertical</b> .
0x000D	<p>The fill pattern is light down. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following light down patterns is used:</p>  <b>DashedDownwardDiagonal</b> : Specifies dashed diagonal lines that slant to the right from top to bottom.  <b>DiagonalBrick</b> : Specifies a hatch that has the appearance of layered bricks that slant to the left from top to bottom.

Value	Description
	 <b>DottedGrid:</b> Specifies horizontal and vertical lines that cross, each of which is composed of dots.  <b>LightDownwardDiagonal:</b> Specifies diagonal lines that slant to the right from top to bottom, but are not anti-aliased.  <b>OutlinedDiamond:</b> Specifies forward diagonal and backward diagonal lines that cross, but are not anti-aliased.  <b>Shingle:</b> Specifies a hatch that has the appearance of diagonally layered shingles that slant to the right from top to bottom.  <b>Wave:</b> Specifies horizontal lines that are composed of tilde (~) characters.  <b>ZigZag:</b> Specifies horizontal lines that are composed of zigzag lines.
0x000E	<p>The fill pattern is light up. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following light up patterns is used:</p>  <b>DashedUpwardDiagonal:</b> Specifies dashed diagonal lines that slant to the left from top to bottom.  <b>Divot:</b> Specifies a hatch that has the appearance of divots.  <b>LightUpwardDiagonal:</b> Specifies diagonal lines that slant to the left from top to bottom, but are not anti-aliased.  <b>Weave:</b> Specifies a hatch that has the appearance of a woven material.
0x000F	<p>The fill pattern is light grid. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following light grid patterns is used:</p>  <b>HorizontalBrick:</b> Specifies a hatch that has the appearance of horizontally layered bricks.  <b>LargeGrid:</b> Specifies horizontal and vertical lines that cross.  <b>SmallGrid:</b> Specifies horizontal and vertical lines that cross and are spaced 50 percent closer together than hatch style <b>LargeGrid</b> .

Value	Description
0x0010	<p>The fill pattern is light trellis. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following light trellis patterns is used:</p> <p> <b>Percent30</b>: Specifies a 30 percent hatch. The ratio of foreground color to background color is 30:100.</p> <p> <b>Percent40</b>: Specifies a 40 percent hatch. The ratio of foreground color to background color is 40:100.</p> <p> <b>LargeConfetti</b>: Specifies a hatch that has the appearance of confetti and is composed of larger pieces than <b>SmallConfetti</b>.</p>
0x0011	<p>The fill pattern is grayscale of 0.125 (1/8) value. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following grayscale patterns is used:</p> <p> <b>Percent20</b>: Specifies a 20 percent hatch. The ratio of foreground color to background color is 20:100.</p> <p> <b>DottedDiamond</b>: Specifies forward diagonal and backward diagonal lines that cross, each of which is composed of dots.</p> <p> <b>SmallConfetti</b>: Specifies a hatch that has the appearance of confetti.</p>
0x0012	<p>The fill pattern is grayscale of 0.0625 (1/16) value. Additional properties in the corresponding <b>GelFrame</b> record specify which of the following grayscale patterns is used:</p> <p> <b>Percent5</b>: Specifies a 5 percent hatch. The ratio of foreground color to background color is 5:100.</p> <p> <b>Percent10</b>: Specifies a 10 percent hatch. The ratio of foreground color to background color is 10:100.</p>

**A - fAuto (1 bit):** A bit that specifies whether the fill colors are automatically set. If **fAuto** is set to 0x0001, the formatting is automatic. The default value is 1.

**B - fInvertNeg (1 bit):** A bit that specifies whether the foreground and background are swapped when the data value of the filled area is negative. This field **MUST** be ignored if the formatting is not being applied to a data point on a bar or column chart group. The default value is zero.

**reserved (14 bits):** **MUST** be set to zero, and **MUST** be ignored.

**icvFore (2 bytes):** An unsigned integer that specifies the foreground color of the fill pattern. This value **SHOULD** <14> be an **IcvChart** value. The default value is selected automatically from the next available color in the Chart color table.

**icvBack (2 bytes):** An unsigned integer that specifies the background color of the fill pattern. The value SHOULD <15> be an **IcvChart** value. The default value is 0x0009.

#### 2.4.4 AttachedLabel

This record specifies properties of a data label on a chart group, series, or data point. See the **Data Label** (section 2.2.1.10) for additional information about how this record is used and when this record is ignored.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
A	B	C	D	E	F	G	reserved																								

**A - fShowValue (1 bit):** A bit that specifies whether the value, or the vertical value on bubble or scatter chart groups, is displayed in the data label.

This value MUST be 0 if this record is in a chart group and either **fShowLabel** or **fShowPercent** is equal to 1.

**B - fShowPercent (1 bit):** A bit that specifies whether the value, represented as a percentage of the sum of the values of the series with which the data label is associated, is displayed in the data label.

MUST equal 0 if the chart group type of the corresponding chart group, series, or data point, is not a bar of pie, doughnut, pie, or pie of pie chart group.

If this record is contained in a chart group and **fShowLabelAndPerc** is equal to 1, this field MUST equal 1.

**C - fShowLabelAndPerc (1 bit):** A bit that specifies whether the **category** name and value, represented as a percentage of the sum of the values of the series with which the data label is associated, are displayed in the data label.

MUST equal 0 if the chart group type of the corresponding chart group, series, or data point is not a bar of pie, doughnut, pie, or pie of pie chart group.

**D - unused (1 bit):** Undefined and MUST be ignored.

**E - fShowLabel (1 bit):** A bit that specifies whether the category, or the horizontal value on bubble or scatter chart groups, is displayed in the data label on a non-area chart group, or the series name is displayed in the data label on an area chart group.

This field MUST equal 0 if this record is contained in a chart group and one of the following conditions is satisfied:

- The **fShowValue** field is equal to 1.
- The **fShowLabelAndPerc** field is equal to zero, and the **fShowPercent** field is equal to 1.

**F - fShowBubbleSizes (1 bit):** A bit that specifies whether the **bubble size** is displayed in the data label.

MUST equal 0 if the chart group type of the corresponding chart group, series, or data point is not a bubble chart group.

If the current record is contained in a chart group and **fShowPercent**, **fShowValue**, or **fShowLabel** equal 1, this field MUST equal 0.

**G - fShowSeriesName (1 bit):** A bit that specifies whether the data label contains the name of the series.

If the current record is contained in a chart group and **fShowLabelAndPerc**, **fShowPercent**, **fShowValue**, **fShowLabel**, or **fShowBubbleSizes** equal 1, this field MUST be equal to zero.

**reserved (9 bits):** MUST be zero, and MUST be ignored.

### 2.4.5 AxcExt

This record specifies additional extension properties of a date axis, along with a **CatSerRange** record.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
catMin																catMax															
catMajor																duMajor															
catMinor																duMinor															
duBase																catCrossDate															
A	B	C	D	E	F	G	H	reserved																							

**catMin (2 bytes):** An unsigned integer that specifies the minimum date, as a date in the **date system** specified by the **Date1904** record, in the units defined by **duBase**. SHOULD [<16>](#) be less than or equal to **catMax**. If **fAutoMin** is set to 1, this field MUST be ignored. If **fDateAxis** is set to zero, this field MUST be ignored.

**catMax (2 bytes):** An unsigned integer that specifies the maximum date, as a date in the date system specified by the **Date1904** record, in the units defined by **duBase**. SHOULD [<17>](#) be greater than or equal to **catMin**. If **fAutoMax** is set to 1, this field MUST be ignored. If **fDateAxis** is set to zero, this field MUST be ignored.

**catMajor (2 bytes):** An unsigned integer that specifies the interval at which the **major tick marks** are displayed on the axis, in the unit defined by **duMajor**. MUST be greater than or equal to **catMinor** when **duMajor** is equal to **duMinor**. If **fAutoMajor** is set to 1, this field MUST be ignored. If **fDateAxis** is set to zero, this field MUST be ignored.

**duMajor (2 bytes):** A **DateUnit** structure (section [2.5.4](#)) that specifies the unit of time for **catMajor** when the axis is a date axis. If **fDateAxis** is set to zero, this field MUST be ignored.

**catMinor (2 bytes):** An unsigned integer that specifies the interval at which the **minor tick marks** are displayed on the axis, in a unit defined by **duMinor**. MUST be less than or equal to **catMajor** when **duMajor** is equal to **duMinor**. If **fAutoMinor** is set to 1, this field MUST be ignored. If **fDateAxis** is set to zero, this field MUST be ignored.

**duMinor (2 bytes):** A **DateUnit** structure (section [2.5.4](#)) that specifies the unit of time for **catMinor** when the axis is a date axis. If **fDateAxis** is set to zero, this field MUST be ignored.

**duBase (2 bytes):** A **DateUnit** structure (section [2.5.4](#)) that specifies the smallest unit of time used by the axis. If **fAutoBase** is set to 1, this field MUST be ignored. If **fDateAxis** is set to zero, this field MUST be ignored.

**catCrossDate (2 bytes):** An unsigned integer that specifies at which date, as a date in the date system specified by the **Date1904** record, in the units defined by **duBase**, the value axis crosses



this axis. If **fDateAxis** is set to zero, this field MUST be ignored. If **fAutoCross** is set to 1, this field MUST be ignored.

**A - fAutoMin (1 bit):** A bit that specifies whether **catMin** is calculated automatically. If **fDateAxis** is set to 0, this field MUST be ignored. MUST be a value from the following table.

Value	Meaning
0	The value specified by <b>catMin</b> is used, and <b>catMin</b> is not calculated automatically.
1	The <b>catMin</b> value is calculated so that the minimum data point value can be displayed.

**B - fAutoMax (1 bit):** A bit that specifies whether **catMax** is calculated automatically. If **fDateAxis** is set to zero, this field MUST be ignored. If the value of the **fMaxCross** field in the **CatSerRange** record is set to 1, this field MUST be ignored. MUST be a value from the following table.

Value	Meaning
0	The value of <b>catMax</b> is used, and <b>catMax</b> is not calculated automatically.
1	The value of <b>catMax</b> is calculated so that the minimum data point value can be displayed.

**C - fAutoMajor (1 bit):** A bit that specifies whether **catMajor** is calculated automatically. If **fDateAxis** is set to zero, this field MUST be ignored. MUST be a value from the following table.

Value	Meaning
0	The value of <b>catMajor</b> is used, and <b>catMajor</b> is not calculated automatically.
1	The value of <b>catMajor</b> is calculated automatically.

**D - fAutoMinor (1 bit):** A bit that specifies whether **catMinor** is calculated automatically. If **fDateAxis** is set to zero, this field MUST be ignored. MUST be a value from the following table.

Value	Meaning
0	The value of <b>catMinor</b> is used, and <b>catMinor</b> is not calculated automatically.
1	The value of <b>catMinor</b> is calculated automatically.

**E - fDateAxis (1 bit):** A bit that specifies whether the axis is a date axis. MUST be a value from the following table.

Value	Meaning
0	The axis is not a date axis.
1	The axis is a date axis.

**F - fAutoBase (1 bit):** A bit that specifies whether the units of the date axis are chosen automatically. If **fDateAxis** is set to zero, this field **MUST** be ignored. **MUST** be a value from the following table

Value	Meaning
0	The value of <b>duBase</b> is used, and <b>duBase</b> is not calculated automatically.
1	The value of <b>duBase</b> is calculated automatically.

**G - fAutoCross (1 bit):** A bit that specifies whether **catCrossDate** is calculated automatically. **MUST** be a value from the following table.

Value	Meaning
0	The value of <b>catCrossDate</b> is used, and <b>catCrossDate</b> is not calculated automatically.
1	The value of <b>catCrossDate</b> is calculated automatically so that it can be displayed.

**H - fAutoDate (1 bit):** A bit that specifies whether the axis type is detected automatically. **MUST** be a value from the following table.

Value	Meaning
0	The axis is displayed as specified by the <b>fDateAxis</b> field.
1	The axis becomes a date axis automatically if the related data contains date values. Otherwise, the axis is a <b>category</b> axis.

**reserved (8 bits):** **MUST** be zero, and **MUST** be ignored.

## 2.4.6 AxesUsed

This record specifies the number of axis groups on the chart.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
cAxes																															

**cAxes (2 bytes):** An unsigned integer that specifies the number of axis groups on the chart. **MUST** be a value from the following table.

Value	Meaning
0x0001	A single primary axis group is present.
0x0002	Both a primary axis group and a secondary axis group are present.

If no chart groups are present on the chart, this field MUST be 0x0001. If the chart sheet substream contains a **Chart3d** record, this field MUST be 0x0001.

### 2.4.7 Axis

This record specifies properties of an axis and the beginning of a collection of records as defined by the chart sheet substream **ABNF** that specifies an axis.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
wType																reserved1															
...																reserved2															
...																reserved3															
...																reserved4															
...																															

**wType (2 bytes):** An unsigned integer that specifies the type of axis. The value MUST be 0x0000 if the record is the first axis in the axis group. The value MUST be 0x0001 if the record is the second axis in the axis group. The value MUST be 0x0002 if the record is the third axis in the axis group. MUST be a value from the following table.

Value	Meaning
0x0000	The axis type is a horizontal value axis for a scatter chart group or a bubble chart group, or a <b>category</b> axis for all other chart group types.
0x0001	The axis type is a vertical value axis for a scatter chart group or a bubble chart group, or a value axis for all other chart group types.
0x0002	The axis type is a series axis.

**reserved1 (4 bytes):** MUST be zero, and MUST be ignored.

**reserved2 (4 bytes):** MUST be zero, and MUST be ignored.

**reserved3 (4 bytes):** MUST be zero, and MUST be ignored.

**reserved4 (4 bytes):** MUST be zero, and MUST be ignored.

### 2.4.8 AxisLine

This record specifies which part of the axis is specified by the **LineFormat** record that follows.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
id																															

**id (2 bytes):** An unsigned integer that specifies which part of the axis is defined by the **LineFormat** record that follows. MUST be unique among all other **id** field values in **AxisLine** records in the

current axis. MUST be greater than the **id** field values in preceding **AxisLine** records in the current axis. MUST be a value from the following table:

Value	Meaning
0x0000	The axis line itself.
0x0001	The <b>major gridlines</b> along the axis.
0x0002	The <b>minor gridlines</b> along the axis.
0x0003	The <b>walls</b> or <b>floor</b> of a three-dimensional (3-D) chart.

If **id** is set to 0x0003, this record MUST be preceded by an **Axis** record with the **wType** field set to a value from the following table.

Value	Meaning
0x0000	The walls of a 3-D chart.
0x0001	The floor of a 3-D chart.

### 2.4.9 AxisParent

This record specifies properties of an axis group and the beginning of a collection of records as defined by the chart sheet substream **ABNF** that specifies an axis group.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
iax																unused (16 bytes)															
...																															
...																															
...																															

**iax (2 bytes):** A **Boolean** that specifies whether the axis group is primary or secondary. This field MUST be set to zero when it is in the first **AxisParent** record in the chart sheet substream ABNF. This field MUST be set to 1 when it is in the second **AxisParent** record in the chart sheet substream ABNF. This field MUST be a value from the following table.

Value	Meaning
0x0000	Axis group is primary.
0x0001	Axis group is secondary.

**unused (16 bytes):** Undefined and MUST be ignored.

### 2.4.10 Bar

This record specifies that the chart group is a bar chart group or a column chart group, and specifies the chart group attributes.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
pcOverlap																pcGap															
A	B	C	D	reserved																											

**pcOverlap (2 bytes):** A signed integer that specifies the overlap between data points in the same **category** as a percentage of the data point width. MUST be greater than or equal to -100 and less than or equal to 100. MUST be a value from the following table.

Value	Meaning
-100 to -1	Size of the separation between data points.
0	No overlap.
1 to 100	Size of the overlap between data points.

**pcGap (2 bytes):** An unsigned integer that specifies the width of the gap between the categories and the left and right edges of the **plot area** as a percentage of the data point width divided by 2. It also specifies the width of the gap between adjacent categories as a percentage of the data point width. MUST be less than or equal to 0x01F4.

**A - fTranspose (1 bit):** A bit that specifies whether the data points and value axis are horizontal for a bar chart group, or vertical for a column chart group. MUST be a value from the following table.

Value	Meaning
0	The data points and value axis are vertical.
1	The data points and value axis are horizontal.

**B - fStacked (1 bit):** A bit that specifies whether the data points in the chart group that share the same category are stacked.

**C - f100 (1 bit):** A bit that specifies whether the data points in the chart group are displayed as a percentage of the sum of all data points in the chart group that share the same category. MUST be 0 if **fStacked** is 0.

**D - fHasShadow (1 bit):** A bit that specifies whether one or more data points in the chart group have shadows.

**reserved (12 bits):** MUST be zero, and MUST be ignored.

### 2.4.11 Begin

This record specifies the beginning of a collection of records as defined by the chart sheet substream **ABNF**. The collection of records specifies properties of a chart.

### 2.4.12 Blank

This record specifies an empty **cell** with no **formula** or value.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rw																col															
reserved										ifmt																					

**rw (2 bytes):** A **Graph\_Rw** structure (section [2.5.15](#)) that specifies the zero-based row number of the empty cell.

**col (2 bytes):** A **Graph\_Col** structure (section [2.5.14](#)) that specifies the zero-based column number of the empty cell.

**reserved (1 byte):** MUST be zero, and MUST be ignored.

**ifmt (2 bytes):** An **IFmt** structure (section [2.5.19](#)) that specifies the number formatting to use for the data.

### 2.4.13 BOF

This record specifies properties of the substream. It also specifies the beginning of a collection of records as defined by the workbook stream **ABNF** and the chart sheet substream **ABNF**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
version																docType															
rupBuild																rupYear															
A	B	C	D	E	F	G	H	I	J	K	L	M	N	reserved1																	
verLowestBiff										O		reserved2																			

**version (2 bytes):** An unsigned integer that specifies the version of the substream. MUST be 0x0680.

**docType (2 bytes):** An unsigned integer that specifies the type of data contained in the substream. MUST be a value from the following table.

Value	Meaning
0x0005	A workbook stream
0x8000	A chart sheet substream

**rupBuild (2 bytes):** An unsigned integer that specifies the **build identifier** of the application that created the substream.

**rupYear (2 bytes):** An unsigned integer that specifies the version of the file format. This value MUST be 0x07CC or 0x07CD. This value SHOULD [<18>](#) be 0x07CD (1997).

**A - fWin (1 bit):** A bit that specifies whether this substream was last edited on a Windows-based platform. MUST be 1.

- B - fRisc (1 bit):** A bit that specifies whether the substream was last edited on a RISC platform [<19>](#). MUST be 0.
- C - fBeta (1 bit):** A bit that specifies whether this substream was last edited by a **beta** version of the application. MUST be 0.
- D - fWinAny (1 bit):** A bit that specifies whether this substream has ever been edited on a Windows-based platform. MUST be 1.
- E - fMacAny (1 bit):** A bit that specifies whether this substream has ever been edited on a Macintosh platform. MUST be 0.
- F - fBetaAny (1 bit):** A bit that specifies whether this substream has ever been edited by a beta version of the application. MUST be 0.
- G - unused1 (2 bits):** Undefined, and MUST be ignored.
- H - fRiscAny (1 bit):** A bit that specifies whether this substream has ever been edited on a RISC platform. MUST be 0.
- I - fOOM (1 bit):** A bit that specifies whether this substream caused an **out-of-memory** failure.
- J - fGIJump (1 bit):** A bit that specifies whether this substream caused an out-of-memory failure while loading chart or graph data.
- K - unused2 (2 bits):** Undefined, and MUST be ignored.
- L - fFontLimit (1 bit):** A bit that specifies whether this substream has reached the 255 font limit [<20>](#), and new **Font** records cannot be added to it.
- M - verXLHigh (4 bits):** An unsigned integer that specifies the highest version of the application that has ever saved this substream. MUST be a value from the following table.

Value	Meaning
0x0	Version-specific value that specifies the highest version of the application that has ever saved this substream. <a href="#">&lt;21&gt;</a>
0x1	Version-specific value that specifies the highest version of the application that has ever saved this substream. <a href="#">&lt;22&gt;</a>
0x2	Version-specific value that specifies the highest version of the application that has ever saved this substream. <a href="#">&lt;23&gt;</a>
0x3	Version-specific value that specifies the highest version of the application that has ever saved this substream. <a href="#">&lt;24&gt;</a>
0x4	Version-specific value that specifies the highest version of the application that has ever saved this substream. <a href="#">&lt;25&gt;</a>
0x6	Version-specific value that specifies the highest version of the application that has ever saved this substream. <a href="#">&lt;26&gt;</a>

**N - unused3 (1 bit):** Undefined and MUST be ignored.

**reserved1 (13 bits):** MUST be zero, and MUST be ignored.

**verLowestBiff (8 bits):** An unsigned integer that specifies the version of the file format. MUST be 0x06.

**O - verLastXLSaved (4 bits):** An unsigned integer that specifies the application version that saved this substream most recently. This value MUST be less than or equal to the value of the **verXLHigh** field. MUST be a value from the following table:

Value	Meaning
0x0	Specifies the application version that saved this substream most recently.<27>
0x1	Specifies the application version that saved this substream most recently.<28>
0x2	Specifies the application version that saved this substream most recently.<29>
0x3	Specifies the application version that saved this substream most recently.<30>
0x4	Specifies the application version that saved this substream most recently.<31>
0x6	Specifies the application version that saved this substream most recently.<32>

**reserved2 (20 bits):** MUST be zero, and MUST be ignored.

#### 2.4.14 BOFDatasheet

This record specifies the beginning of a collection of records, as defined by the chart sheet substream **ABNF**. The collection of records specifies a data sheet.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
unused1																unused2															

**unused1 (2 bytes):** Undefined and MUST be ignored.

**unused2 (2 bytes):** Undefined and MUST be ignored.

#### 2.4.15 BopPop

This record specifies that the chart group is a bar of pie chart group or a pie of pie chart group, and it specifies the chart group attributes.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
pst				fAutoSplit								split																			
iSplitPos																pcSplitPercent															
pcPie2Size																pcGap															
numSplitValue																															
...																															
A	reserved																														

**pst (1 byte):** An unsigned integer that specifies whether this chart group is a bar of pie chart group or a pie of pie chart group. MUST be a value from the following table.



Value	Subtype
0x01	Pie of pie chart group
0x02	Bar of pie chart group

**fAutoSplit (1 byte):** A Boolean that specifies whether the split point of the chart group is determined automatically. If this field is set to 1, the data points from the **primary pie** are selected and inserted into the **secondary bar/pie** automatically when a bar of pie chart group or pie of pie chart group is initially created.

**split (2 bytes):** An unsigned integer that specifies what determines the split between the primary pie and the secondary bar/pie. MUST be ignored if **fAutoSplit** is set to 1. MUST be a value from the following table.

Value	Type of split	Meaning
0x0000	Position	The data is split based on the position of the data point in the series, as specified by <b>iSplitPos</b> .
0x0001	Value	The data is split based on a threshold value, as specified by <b>numSplitValue</b> .
0x0002	Percent	The data is split based on a percentage threshold, and the data point values is represented as a percentage, as specified by <b>pcSplitPercent</b> .
0x0003	Custom	The data split is user-defined. A custom split is specified in a following <b>BopPopCustom</b> record.

**iSplitPos (2 bytes):** A signed integer that specifies how many data points are contained in the secondary bar/pie. Data points are contained in the secondary bar/pie, starting from the end of the series. For example, if this field is set to 0x0002, the last two data points in the series are contained in the secondary bar/pie. MUST be greater than zero and less than or equal to 0x7D00. SHOULD [<33>](#) be a value that is less than or equal to 0x0F9F. If the value is greater than the number of data points in the series, the entire series is in the secondary bar/pie, except the first data point. If **split** is not set to 0x0000 or if **fAutoSplit** is set to 0x0001, this value MUST be ignored.

**pcSplitPercent (2 bytes):** A signed integer that specifies the percentage below which each data point is contained in the secondary bar/pie, as opposed to the primary pie. The percentage value of a data point is calculated by using the following formula:

$(\text{value of the data point} \times 100) / \text{Sum of all data point in the series.}$

If **split** is not set to 0x0002 or if **fAutoSplit** is set to 0x0001, this value MUST be ignored.

**pcPie2Size (2 bytes):** A signed integer that specifies the size of the secondary bar/pie as a percentage of the size of the primary pie. MUST be a value greater than or equal to 0x0005 and less than or equal to 0x00C8.

**pcGap (2 bytes):** A signed integer that specifies the distance between the primary pie and the secondary bar/pie. The distance is specified as a percentage of the average width of the primary pie and secondary bar/pie. MUST be a value greater than or equal to zero and less than or equal to 0x00C8, where zero is zero percent of the average width of the primary pie and the secondary bar/pie, and 0x00C8 is 100 percent of the average width of the primary pie and the secondary bar/pie.

**numSplitValue (8 bytes):** An **Xnum** structure (section [2.5.33](#)) that specifies the split when the **split** field is set to 0x0001. The value of this field specifies the threshold that selects which data points of the primary pie move to the secondary bar/pie. The secondary bar/pie contains any data points with a value that is less than this field. If **split** is not set to 0x0001 or if **fAutoSplit** is set to 0x0001, this value MUST be ignored.

**A - fHasShadow (1 bit):** A bit that specifies whether one or more data points in the chart group have shadows.

**reserved (15 bits):** MUST be zero, and MUST be ignored.

#### 2.4.16 BopPopCustom

This record specifies which data points in the series are contained in the **secondary bar/pie**, instead of the **primary pie**. MUST follow a **BopPop** record that contains a **split** field that is set to "Custom (0x0003)".

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
cxi																rggrbit (variable)															
...																															

**cxi (2 bytes):** An unsigned integer that specifies the number of data points in the series plus 1. MUST be less than 0x0FA0.

**rggrbit (variable):** A sequence of bits that specifies whether each data point in the series is contained in the primary pie or the secondary bar/pie. For each data point, a corresponding bit specifies whether a data point is contained in the secondary bar/pie or the primary pie, as specified in the following table.

Value	Meaning
0	The data point is in the primary pie.
1	The data point is in the secondary bar/pie.

The size of this field, in bytes, is calculated by using the following formula:

$$\text{size of rggrbit in bytes} = 1 + \text{floor}(\text{cxi} / 8)$$

The padding of this field, in bits, is calculated by using the following formula:

$$\text{padding} = \text{size of rggrbit in bits} - \text{cxi}$$

The position of each bit in the sequence corresponds to the position of each data point in the series, arranged as follows:

- Any padding is placed in the most significant bits of the first byte. The next most significant bit of the first byte corresponds to the first data point in the series. If there are any more unused bits in

the first byte, the next most significant bit corresponds to the second data point in the series. This pattern continues until there are no remaining unused bits in the first byte.

- The most significant bit of the next byte corresponds to the next data point in the series. The next most significant bit contains the next data point in the series. This pattern continues for each byte in the sequence.
- The least significant bit of the final byte is an additional bit that specifies whether the secondary bar/pie does not contain data points. If the additional bit is 1, the value of every other bit MUST be 0.

The following figure demonstrates this order.

Byte 0							
0 (LSB)	1	2	3	4	5	6	7
Data Point 3	Data Point 2	Data Point 1	Data Point 0	Padding			

Byte 1							
8	9	10	11	12	13	14	15 (MSB)
Additional Bit	Data Point 10	Data Point 9	Data Point 8	Data Point 7	Data Point 6	Data Point 5	Data Point 4

**Figure 2: Bit arrangement in this field**

### 2.4.17 BoundSheet8

This record specifies basic information about a **sheet**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
lbPlyPos																															
A	unused						dt				stName (variable)																				
...																															

**lbPlyPos (4 bytes):** A **FilePointer** as specified in [\[MS-OSHARED\]](#) section 2.2.1.5 that specifies the stream position of the start of the **BOF** record for the sheet.

**A - hsState (2 bits):** An unsigned integer that specifies the hidden state of the sheet. MUST be 0x0, indicating a visible sheet.

**unused (6 bits):** Undefined and MUST be ignored.

**dt (8 bits):** An unsigned integer that specifies the sheet type. MUST be 0x02, indicating a chart sheet.

**stName (variable):** A **ShortXLUnicodeString** structure (section [2.5.27](#)) that specifies the name of the sheet. MUST be an empty string, with size zero, and no characters.

#### 2.4.18 BRAI

This record specifies a reference to data in a **sheet** that is used by a part of a series or by a **legend entry**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
id										rt							A	B	reserved2												
ifmt																rowcol															

**id (1 byte):** An unsigned integer that specifies the part of the chart for which the referenced data is used. MUST be a value from the following table.

Value	Meaning
0x00	Referenced data is used for the series name or the text of a legend entry.
0x01	Referenced data is used for series values.
0x02	Referenced data is used for the series <b>category</b> name.
0x03	Referenced data specifies the <b>bubble size</b> values of the series.

**rt (1 byte):** An unsigned integer that specifies the type of data that is being referenced. MUST be a value from the following table.

Value	Meaning
0x00	The data source is a category name or bubble size that is automatically generated.
0x01	The data source as specified by <b>rowcol</b> .
0x02	The data source as specified by <b>rowcol</b> .

**A - fUnlinkedIfmt (1 bit):** A bit that specifies whether the part of the chart specified by the **id** field uses number formatting from the referenced data. MUST be a value from the following table.

Value	Meaning
0	The data uses the number formatting of the referenced data.
1	The data uses the custom number formatting as specified in the <b>ifmt</b> field.

**B - reserved1 (1 bit):** MUST be 1, and MUST be ignored.

**reserved2 (14 bits):** MUST be zero, and MUST be ignored.

**ifmt (2 bytes):** An **IFmt** structure (section [2.5.19](#)) that specifies the number formatting to use for the data.

**rowcol (2 bytes):** An unsigned integer that specifies the row or column information of the reference. If the **fSeriesInRows** field of the **Orient** record is 1, it MUST contain the row number of the specified information. If the **fSeriesInRows** field of the **Orient** record is 0, it MUST contain the column number of the specified information. The row number is a zero-based index of the row in the datasheet, and the column number is a zero-based index of the column of the data sheet. MUST be less than or equal to 0x0F9F.

#### 2.4.19 CatLab

This record specifies the attributes of the axis label.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
frtHeaderOld																															
wOffset																at															
A	unused															reserved (optional)															

**frtHeaderOld (4 bytes):** An **FrtHeaderOld** structure (section [2.5.10](#)). The **frtHeaderOld.rt** field MUST be 0x0856.

**wOffset (2 bytes):** An unsigned integer that specifies the distance between the axis and axis label. It contains the offset as a percentage of the default distance. The default distance is equal to one-third the height of the font calculated in pixels. MUST be greater than or equal to 0 (zero percent) and less than or equal to 0x03E8 (1000 percent).

**at (2 bytes):** An unsigned integer that specifies the alignment of the axis label. MUST be a value from the following table.

Value	Alignment
0x0001	Top-aligned if the <b>trot</b> field of the <b>Text</b> record of the axis is not set to zero. Left-aligned if the <b>iReadingOrder</b> field of the <b>Text</b> record of the axis specifies <b>left-to-right reading order</b> ; otherwise, right-aligned.
0x0002	Centered.
0x0003	Bottom-aligned if the <b>trot</b> field of the <b>Text</b> record of the axis is not set to zero. Right-aligned if the <b>iReadingOrder</b> field of the <b>Text</b> record of the axis specifies left-to-right reading order; otherwise, left-aligned.

**A - cAutoCatLabelReal (1 bit):** A bit that specifies whether the number of **categories** between axis labels is set to the default value. MUST be a value from the following table.

Value	Description
0	The value is set to <b>caLabel</b> field as specified by the <b>CatSerRange</b> record.
1	The value is set to the default value. The number of category labels is automatically calculated by the application, based on the data in the chart.

**unused (15 bits):** Undefined, and MUST be ignored.

**reserved (2 bytes):** MUST be zero, and MUST be ignored.

## 2.4.20 CatSerRange

This record specifies the properties of a **category** axis, a date axis, or a series axis.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31			
catCross																catLabel																		
catMark																A	B	C	reserved															

**catCross (2 bytes):** A signed integer that specifies where the value axis crosses this axis. If **fMaxCross** is set to 1, this field MUST be ignored. MUST be a value from the following table.

Axis type	Range
Category axis	This field specifies the category at which the value axis crosses. For example, if this field is 2, the value axis crosses this axis at the second category on this axis. MUST be greater than or equal to 0x0001 and less than or equal to 0x7CFF.
Series axis	MUST be 0.
Date axis	MUST be equal to the value calculated by the following formula:  $\text{catCross} = \text{catCrossDate} - \text{catMin} + 1$ <p>Where <b>catCrossDate</b> is the <b>catCrossDate</b> field of the <b>AxcExt</b> record and <b>catMin</b> is the <b>catMin</b> field of the <b>AxcExt</b> record.</p>

**catLabel (2 bytes):** A signed integer that specifies the interval between axis labels on this axis. MUST be greater than or equal to 0x0001 and less than or equal to 0x7CFF. MUST be ignored for a date axis.

**catMark (2 bytes):** A signed integer that specifies the interval at which **tick marks** are displayed on the axis. **Major tick marks** and **minor tick marks** that would be visible are hidden, unless they are located at a multiple of this field. MUST be greater than or equal to 0x0001 and less than or equal to 0x7CFF. MUST be ignored for a date axis.

**A - fBetween (1 bit):** A bit that specifies whether the value axis crosses this axis between major tick marks. MUST be a value from the following table.

Value	Meaning
0	The value axis crosses this axis on a major tick mark.
1	The value axis crosses this axis between major tick marks.

**B - fMaxCross (1 bit):** A bit that specifies whether the value axis crosses this axis at the last category, the last series, or the maximum date. MUST be a value from the following table.

Value	Meaning
0	The value axis crosses this axis at the value specified by <b>catCross</b> .
1	The value axis crosses this axis at the last category, the last series, or the maximum date.

**C - fReverse (1 bit):** A bit that specifies whether the axis is displayed in reverse order. MUST be a value from the following table.

Value	Meaning
0	The axis is displayed in order.
1	The axis is display in reverse order.

**reserved (13 bits):** MUST be zero, and MUST be ignored.

#### 2.4.21 Chart

This record specifies the position and size of the **chart area** and the beginning of a collection of records as defined by the chart sheet substream **ABNF**. The collection of records specifies a chart.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
																x															
																y															
																dx															
																dy															

**x (4 bytes):** A **FixedPoint**, as specified in [MS-OSHARED] section 2.2.1.6, that specifies the horizontal position of the upper-left corner of the chart, in **points**. MUST be set to zero.

**y (4 bytes):** A **FixedPoint**, as specified in [MS-OSHARED] section 2.2.1.6, that specifies the vertical position of the upper-left corner of the chart, in points. MUST be set to zero.

**dx (4 bytes):** A **FixedPoint**, as specified in [MS-OSHARED] section 2.2.1.6, that specifies the width of the chart, in points. MUST be greater than or equal to zero.

**dy (4 bytes):** A **FixedPoint**, as specified in [MS-OSHARED] section 2.2.1.6, that specifies the height of the chart, in points. MUST be greater than or equal to zero.

#### 2.4.22 Chart3d

This record specifies that the **plot area** of the chart group is rendered in a 3-D scene and also specifies the attributes of the 3-D plot area. The preceding chart group type MUST be of type bar, pie, line, area, or surface.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
anRot																anElev															
pcDist																pcHeight															

pcDepth						pcGap
A	B	C	D	E	F	reserved2

**anRot (2 bytes):** A signed integer that specifies the clockwise rotation, in degrees, of the 3-D plot area around a vertical line through the center of the 3-D plot area. MUST be greater than or equal to zero and MUST be less than or equal to 0x0168. If the chart group type is bar and the value of field **fTranspose** in the **Bar** record is set to 1, this field MUST be less than or equal to 0x002C.

**anElev (2 bytes):** A signed integer that specifies the rotation, in degrees, of the 3-D plot area around a horizontal line through the center of the 3-D plot area. MUST be greater than or equal to -90 and MUST be less than or equal to 90. If the chart group type is bar and the value field **fTranspose** in the **Bar** record is set to 1 or chart group type pie, this field MUST be greater than or equal to zero. If the chart group type is bar and the value of field **fTranspose** in the **Bar** record is set to 1, this field MUST be less than or equal to 44.

**pcDist (2 bytes):** A signed integer that specifies the field of view angle for the 3-D plot area. MUST be greater than or equal to zero and less than or equal to 0x0064.

**pcHeight (2 bytes):** If **fNotPieChart** is set to zero, this field is an unsigned integer that specifies the thickness of the pie for a pie chart group. If **fNotPieChart** is set to 1, this field specifies the height of the 3-D plot area as a percentage of its width. MUST be greater than or equal to 0x0005 and less than or equal to 0x01F4.

**pcDepth (2 bytes):** A signed integer that specifies the depth of the 3-D plot area as a percentage of its width. MUST be greater than or equal to 0x0014 and less than or equal to 0x07D0.

**pcGap (2 bytes):** An unsigned integer that specifies the width of the gap between the series and the front and back edges of the 3-D plot area as a percentage of the data point depth divided by 2. If **fCluster** is not set to 1 and chart group type is not a bar, this field also specifies the distance between adjacent series as a percentage of the data point depth. MUST be less than or equal to 0x01F4.

**A - fPerspective (1 bit):** A bit that specifies whether the 3-D plot area is rendered with a vanishing point. If **fNotPieChart** is set to zero, this field MUST be set to zero. If **fNotPieChart** is set to 1, this field MUST be set to a value from the following table.

Value	Meaning
0	No perspective vanishing point applied.
1	Perspective vanishing point applied based on the value of <b>pcDist</b> .

**B - fCluster (1 bit):** A bit that specifies whether data points are clustered together in a bar chart group. If the chart group type is not bar or pie, this field MUST be ignored. If the chart group type is pie, this field MUST be set to zero. If the chart group type is bar, this field MUST be set to a value from the following table.

Value	Meaning
0	Data points are not clustered.
1	Data points are clustered.

**C - f3DScaling (1 bit):** A bit that specifies whether the height of the 3-D plot area is automatically determined. If **fNotPieChart** is set to zero, this field MUST be set to zero. If **fNotPieChart** is set to 1, this field MUST be set to a value from the following table.



Value	Meaning
0	The value of <b>pcHeight</b> is used to determine the height of the 3-D plot area.
1	The height of the 3-D plot area is automatically determined.

**D - reserved1 (1 bit):** MUST be zero, and MUST be ignored.

**E - fNotPieChart (1 bit):** A bit that specifies whether the chart group type is pie. MUST be set to a value from the following table.

Value	Meaning
0	Chart group type MUST be pie.
1	Chart group type MUST NOT be pie.

**F - fWalls2D (1 bit):** A bit that specifies whether the chart **walls** are rendered in 2-D<34>. If **fPerspective** is set to 1, this field MUST be ignored. If the chart group type is not bar, area, or pie, this field MUST be ignored. If the chart group type is bar and **fCluster** is set to zero, this field MUST be ignored. If the chart group type is pie, this field MUST be 0 and MUST be ignored. If the chart group type is bar or area, this field MUST be set to a value from the following table.

Value	Meaning
0	Chart walls and <b>floor</b> are rendered in 3-D.
1	Chart walls are rendered in 2-D, and the chart floor is not rendered.

**reserved2 (10 bits):** MUST be zero, and MUST be ignored.

#### 2.4.23 Chart3DBarShape

This record specifies the shape of the data points in a bar or column chart group. This record is used only for a bar or column chart group and MUST be ignored for all other chart groups. This record MUST be ignored if the current substream does not contain a **Chart3d** record.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
riser										taper																					

**riser (1 byte):** A Boolean that specifies the shape of the base of the data points in a bar or column chart group. MUST be a value from the following table.

Value	Meaning
0x00	The base of the data point is a rectangle.
0x01	The base of the data point is an ellipse.

**taper (1 byte):** An unsigned integer that specifies how the data points in a bar or column chart group taper from base to tip. MUST be a value from the following table.

Value	Meaning
0x00	The data points of the bar or column chart group do not taper. The shape at the maximum

Value	Meaning
	value of the data point is the same as the shape at the base.
0x01	The data points of the bar or column chart group taper to a point at the maximum value of each data point.
0x02	The data points of the bar or column chart group taper towards a projected point at the position of the maximum value of all data points in the chart group, but are clipped at the value of each data point.

#### 2.4.24 ChartColors

This record specifies the number of available colors in the palette.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
icvMac																															

**icvMac (2 bytes):** A signed integer that specifies the number of colors currently available. MUST be equal to the number of items in the **rgColor** field of the **Palette** record immediately following this record. MUST be set to 0x0038.

#### 2.4.25 ChartFormat

This record specifies properties of a chart group and the beginning of a collection of records as defined by the chart sheet substream **ABNF**. The collection of records specifies a chart group.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
reserved1																															
reserved2																															
reserved3																															
reserved4																															
A	reserved5															icrt															

**reserved1 (4 bytes):** MUST be zero, and MUST be ignored.

**reserved2 (4 bytes):** MUST be zero, and MUST be ignored.

**reserved3 (4 bytes):** MUST be zero, and MUST be ignored.

**reserved4 (4 bytes):** MUST be zero, and MUST be ignored.

**A - fVaried (1 bit):** A bit that specifies whether the color for each data point and the color and type for each **data marker** vary. If the chart group has multiple series or the chart group has one series and the chart group type is a surface, stock, or area, this field MUST be ignored, and the data points do not vary. For all other chart group types, if the chart group has one series, a value of 0x1 specifies that the data points vary. MUST be a value from the following table.

Value	Meaning
0x0	The color for each data point and the color and type for each data marker do not vary.
0x1	The color for data points or the color or type for data markers vary.

**reserved5 (15 bits):** MUST be zero, and MUST be ignored.

**icrt (2 bytes):** An unsigned integer that specifies the drawing order of the chart group relative to the other chart groups, where 0x0000 is the bottom of the **z-order**. This value MUST be unique for each instance of this record and MUST be less than or equal to 0x0009.

#### 2.4.26 ChartFrtInfo

This record specifies the versions of the application that originally created and last saved the file, and the future record IDs that are used in the file. This property was introduced by a version of the application [<35>](#) as a chart future record for charts.

In a file written by some versions of the application [<36>](#), this record appears before the end of the **Chart** record collection and before any other future record in the record stream. This record does not exist in a file created by certain versions of the application [<37>](#), but appears after the **End** record of the **Chart** record collection in a file updated by other versions of the application [<38>](#), in which case the **verWriter** field MUST be a certain version of the application [<39>](#), regardless of the actual value in the record.

This record MUST immediately precede the first chart-specific future record, which is a record that has a record number greater than or equal to 0x0800 and less than or equal to 0x08FF according to record enumeration.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
frtHeaderOld																															
verOriginator										verWriter										cCFRTID											
rgCFRTID (variable)																															
...																															

**frtHeaderOld (4 bytes):** An **FrtHeaderOld** structure (section [2.5.10](#)). The **FrtHeaderOld.rt** field MUST be 0x0850.

**verOriginator (1 byte):** An unsigned integer that specifies the application version that originally created the file. MUST be a value from the following table.

Value	Meaning
0x09	Specifies the application version that originally created the file. <a href="#">&lt;40&gt;</a>
0x0A	Specifies the application version that originally created the file. <a href="#">&lt;41&gt;</a>

Value	Meaning
0x0C	Specifies the application version that originally created the file.<42>
0x0E	Specifies the application version that originally created the file.<43>

**verWriter (1 byte):** An unsigned integer that specifies the application version that last saved the file. MUST be a value from the following table.

Value	Meaning
0x09	Specifies the application version that last saved the file.<44>
0x0A	Specifies the application version that last saved the file.<45>
0x0C	Specifies the application version that last saved the file.<46>
0x0E	Specifies the application version that last saved the file.<47>

**cCFRTID (2 bytes):** An unsigned integer that specifies the number of elements in **rgCFRTID**. The value depends on the value in **verWriter**. This field MUST be set to 0x0001 if **verWriter** is set to 0x09, 0x0003 if **verWriter** is set to 0x0A, or 0x0004 if **verWriter** is set to 0x0C or 0x0E.

**rgCFRTID (variable):** An array of **CFrtId** structures (section 2.5.2). This field specifies the list of future record type ID value ranges used in the chart. The value is determined by the value of **verWriter** as specified in the following table.

verWriter value	Value (first, last)
0x09	0x0850, 0x085A
0x0A	0x0850, 0x085A 0x0861, 0x0861 0x086A, 0x086B
0x0C or 0x0E	0x0850, 0x085A 0x0861, 0x0861 0x086A, 0x086B 0x089D, 0x08A6

## 2.4.27 ClrtClient

This record specifies a custom **color palette** for a chart sheet.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
ccv																rgColor (variable)															
...																															

**ccv (2 bytes):** A signed integer that specifies the number of colors in the **rgColor** array. MUST be 0x0003.

**rgColor (variable):** An array of **LongRGB** structures (section 2.5.20). The array specifies the colors of the color palette. MUST be a value from the following table.

Index	Element	Value
0	Foreground color	MUST be equal to the system window text color of the <b>system palette</b> .
1	Background color	MUST be equal to the system window color of the system palette.
2	Neutral color	MUST be black

#### 2.4.28 CodePage

This record specifies **code page** information for the **graph object**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
cv																															

**cv (2 bytes):** An unsigned integer that specifies the code page of the graph object. The value MUST be one of the code page values specified in [\[CODEPG\]](#) or the special value 0x04B0, which means that the text of the graph object is Unicode.

#### 2.4.29 ColumnWidth

This record specifies the width of one or more columns of the **datasheet**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
colFirst																	colLast														
colWidth																															

**colFirst (2 bytes):** A **Graph\_Col** structure (section 2.5.14) that specifies the first column of the range of columns that have their width specified by **colWidth**.

**colLast (2 bytes):** A **Graph\_Col** structure (section 2.5.14) that specifies the last column of the range of columns that have their width specified by **colWidth**. MUST be greater than or equal to **colFirst**.

**colWidth (2 bytes):** An unsigned integer that specifies the column width for all columns from **colFirst** through **colLast**, inclusively. The width is calculated in 256ths of the width of an average character of the current datasheet font.

#### 2.4.30 Continue

This record specifies a continuation of the data of the preceding record. Records with data longer than 8224 bytes MUST be split into several records. The first section of the data appears in the base record,

and subsequent sections appear in one or more **Continue** records that appear after the base record. Records with data shorter than 8225 bytes can also store data in the base record and following **Continue** records. For example, the size of **TxO** record is less than 8225 bytes, but it is followed by **Continue** records that store the string data and **formatting runs**.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
continue (variable)																															
...																															

**continue (variable):** A binary stream that specifies the record data. The number of bytes in this stream **MUST** be less than 8225.

### 2.4.31 Country

This record specifies the country value. that is unused and **MUST** be ignored.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
unused1																unused2															

**unused1 (2 bytes):** Undefined and **MUST** be ignored.

**unused2 (2 bytes):** Undefined and **MUST** be ignored.

### 2.4.32 CrtLine

This record specifies the presence of **drop lines**, **high-low lines**, **series lines**, or **leader lines** on the chart group. This record is followed by a **LineFormat** record that specifies the format of the lines.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
id																															

**id (2 bytes):** An unsigned integer that specifies the type of line that is present on the chart group. This field value **MUST** be unique among the other **id** field values in **CrtLine** records in the current chart group. This field **MUST** be greater than the **id** field values in preceding **CrtLine** records in the current chart group. **MUST** be a value from the following table.

Value	Type of line
0x0000	Drop lines below the data points of line, area, and stock chart groups.
0x0001	High-low lines around the data points of line and stock chart groups.
0x0002	Series lines connecting data points of stacked column and bar chart groups, and the <b>primary pie</b> to the <b>secondary bar/pie</b> of bar of pie and pie of pie chart groups.
0x0003	Leader lines with non-default formatting connecting data labels to the

Value	Type of line
	data point of pie and pie of pie chart groups.

### 2.4.33 CrtLink

This record is written but unused.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
unused																															
...																															
...																															

**unused (10 bytes):** Undefined and MUST be ignored.

### 2.4.34 Dat

This record specifies the beginning of a collection of records as defined by the chart sheet substream **ABNF**. The collection of records specifies the options of the data table that can be displayed in a **chart area**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
A	B	C	D	reserved																											

**A - fHasBordHorz (1 bit):** A bit that specifies whether horizontal **cell borders** are displayed in the data table.

**B - fHasBordVert (1 bit):** A bit that specifies whether vertical cell borders are displayed in the data table.

**C - fHasBordOutline (1 bit):** A bit that specifies whether an **outline** is displayed around the data table.

**D - fShowSeriesKey (1 bit):** A bit that specifies whether the **legend key** is displayed next to the name of the series. If this field is set to 1, the legend key symbols are displayed next to the name of the series.

**reserved (12 bits):** MUST be set to 0x000, and MUST be ignored.

### 2.4.35 DataFormat

This record specifies the data point and series to which the formatting information that follows applies and the beginning of a collection of records as defined by the chart sheet substream **ABNF**. This collection of records specifies formatting properties for the data point and series.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
xi																yi																
iss																A	reserved															

**xi (2 bytes):** An unsigned integer that specifies the zero-based index into the data point in the series specified by **yi**. If this field is set to 0xFFFF, the formatting information that follows applies to the series. Otherwise, the formatting information that follows applies to a data point. This value **MUST** be less than or equal to 0x7CFF. This value **MUST** be less than or equal to 0x0F9F for a chart that contains a **Chart3d** record. This value **MUST** be set to 0xFFFF if the formatting information in this record is applied to a trendline or error bar.

**yi (2 bytes):** An unsigned integer that specifies the zero-based index into a **Series** record in the collection of **Series** records in this chart sheet substream. **SHOULD** <48> be less than or equal to 0x00FE.

**iss (2 bytes):** An unsigned integer that specifies properties of the data series, trendline, or error bar, depending on the type of records contained in the sequence of records that conforms to the **SERIESFORMAT** rule that contains the sequence of records that conforms to the **SS** rule that contains this record.

If the **SERIESFORMAT** rule does not contain a **SerAuxTrend** or a **SerAuxErrBar** record, this field specifies the plot order of the data series. If the series order has been changed, this field can be different from **yi**. **SHOULD** <49> be less than or equal to the number of series in the chart. **MUST** be unique among **iss** values for all instances of this record contained in the **SERIESFORMAT** rule that does not contain a **SerAuxTrend** or a **SerAuxErrBar** record.

If the **SERIESFORMAT** rule contains a **SerAuxTrend** record in the chart group, this field specifies the trendline number for the series.

If the **SERIESFORMAT** rule contains a **SerAuxErrBar** record in the chart group, this field specifies a zero-based index into a **Series** record in the collection of **Series** records in the current chart sheet substream to which the error bar applies.

**A - fXL4iss (1 bit):** **MUST** be zero and **MUST** be ignored.

**reserved (15 bits):** **MUST** be zero, and **MUST** be ignored.

### 2.4.36 DataLabExt

This record specifies the beginning of a collection of records as defined by the chart sheet substream **ABNF**. The collection specifies an extended data label.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
frtHeader																															
...																															
...																															

**frtHeader (12 bytes):** An **FrtHeader** structure (section [2.5.9](#)). The **frtHeader.rt** field **MUST** be 0x086A.



## 2.4.37 DataLabExtContents

This record specifies the contents of an extended data label.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
frtHeader																															
...																															
...																															
A	B	C	D	E	reserved																rgchSep (variable)										
...																															

**frtHeader (12 bytes):** An **FrtHeader** structure (section [2.5.9](#)). The **FrtHeader.rt** field MUST be 0x086B.

**A - fSerName (1 bit):** A bit that specifies whether the name of the series is displayed in the extended data label.

**B - fCatName (1 bit):** A bit that specifies whether the **category** name, or the horizontal value in bubble or scatter chart groups, is displayed in the extended data label. MUST be a value from the following table.

Value	Meaning
0	Neither of the data values is displayed in the extended <b>data label</b> .
1	If the chart group is bubble or scatter, the horizontal value is displayed in the extended data label. Otherwise, the category name is displayed in the extended data label.

**C - fValue (1 bit):** A bit that specifies whether the data value, or the vertical value in bubble or scatter chart groups, is displayed in the extended data label. MUST be a value from the following table.

Value	Meaning
0	Neither of the data values is displayed in the data label.
1	If the chart group is bubble or scatter, the vertical value is displayed in the extended data label. Otherwise, the data value is displayed in the extended data label.

**D - fPercent (1 bit):** A bit that specifies whether the value of the corresponding data point, represented as a percentage of the sum of the values of the series with which the data label is associated, is displayed in the extended data label.

MUST be set to zero if the chart group type of the corresponding chart group, series, or data point is not bar of pie, doughnut, pie, or pie of pie.

**E - fBubSizes (1 bit):** A bit that specifies whether the **bubble size** is displayed in the data label.

MUST be set to zero if the chart group type of the corresponding chart group, series, or data point is not bubble.

**reserved (11 bits):** MUST be zero, and MUST be ignored.

**rgchSep (variable):** A case-sensitive **XLUnicodeStringMin2** structure (section [2.5.31](#)) that specifies the string that is inserted between every data value to form the extended data label. For example, if **fCatName** and **fValue** are set to 1, the labels are formed as follows: "*Category Name*<value of **rgchSep**>*Data Value*". The length of the string is contained in the **cch** field of the **XLUnicodeStringMin2** structure.

### 2.4.38 Date1904

This record specifies the **date system** that is used by the **workbook**.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
f1904DateSystem																															

**f1904DateSystem (2 bytes):** A Boolean that specifies the date system used in this workbook. MUST be a value from the following table.

Value	Meaning
0x0000	The workbook uses the 1900 date system. The first date of the 1900 date system is 00:00:00 on January 1, 1900, specified by a serial value of 1.
0x0001	The workbook uses the 1904 date system. The first date of the 1904 date system is 00:00:00 on January 1, 1904, specified by a serial value of zero.

### 2.4.39 DefaultText

This record specifies the text elements that are formatted using the information specified by the **Text** record immediately following this record.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
id																															

**id (2 bytes):** An unsigned integer that specifies the text elements that are formatted using the position and appearance information specified by the **Text** record immediately following this record.

If this record is located in the sequence of records that conforms to the CRT rule as specified by the chart sheet substream **ABNF**, this record MUST be set to 0x0000 or 0x0001. If this record is not located in the CRT rule as specified by the chart sheet substream **ABNF**, this record MUST be set to 0x0002 or 0x0003. MUST be a value from the following table.

Value	Meaning
0x0000	Format all <b>Text</b> records in the chart group where <b>fShowPercent</b> is set to zero or <b>fShowValue</b> is set to zero.
0x0001	Format all <b>Text</b> records in the chart group where <b>fShowPercent</b> is set to 1 or <b>fShowValue</b> is set to 1.

Value	Meaning
0x0002	Format all <b>Text</b> records in the chart where <b>fScalable</b> of the associated <b>FontInfo</b> structure (section 2.5.6) is set to zero.
0x0003	Format all <b>Text</b> records in the chart where <b>fScalable</b> of the associated <b>FontInfo</b> structure (section 2.5.6) is set to 1.

#### 2.4.40 Dimensions

This record specifies the number of non-empty rows and non-empty **cells** in the longest row of a **Graph object**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
reserved1																															
rwMac																															
reserved2																colMac															
reserved3																															

**reserved1 (4 bytes):** MUST be zero, and MUST be ignored.

**rwMac (4 bytes):** An unsigned integer that specifies the number of non-empty cells in the longest row in the data sheet of a Graph object. MUST be less than or equal to 0x00000F9F.

**reserved2 (2 bytes):** MUST be zero, and MUST be ignored.

**colMac (2 bytes):** An unsigned integer that specifies the number of non-empty rows in the data sheet of a Graph object. MUST be less than or equal to 0x00FF.

**reserved3 (2 bytes):** MUST be zero, and MUST be ignored.

#### 2.4.41 DropBar

This record specifies the attributes of the **up bars** or **down bars** between multiple series of a line chart group and specifies the beginning of a collection of records as defined by the chart sheet substream **ABNF**. The first collection in the line chart group specifies the attributes of the up bars. The second specifies the attributes of the down bars. If this record exists, the chart group type MUST be line, and the field **cSer** in the record **SeriesList** MUST be greater than 1.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
pcGap																															

**pcGap (2 bytes):** A signed integer that specifies the width of the gap between the up bars or the down bars. MUST be set to a value in the range from zero through 0x01F4. The width of the gap in SPRCs is equal to **pcGap** plus 1.

## 2.4.42 End

This record specifies the end of a collection of records as defined by the chart sheet substream **ABNF**. The collection of records specifies properties of a chart.

## 2.4.43 EndBlock

This record specifies the end of a collection of records. Future records contained in this collection specify saved features to enable applications that do not support the feature to preserve the information. This record **MUST** have an associated **StartBlock** record. **StartBlock** and **EndBlock** pairs can be nested. Up to 100 levels of blocks can be nested.

**EndBlock** records **MUST** be written according to the following rules:

- If a **StartBlock** record with **iObjectKind** set to 0x0002 without a matching **EndBlock** exists, the matching **EndBlock** record **MUST** be written immediately before writing the **End** record of the current **AttachedLabel**.
- If a **StartBlock** record with **iObjectKind** set to 0x0004 without a matching **EndBlock** exists, the matching **EndBlock** record **MUST** be written immediately before writing the **End** record of the current axis.
- If a **StartBlock** record with **iObjectKind** set to 0x0005 without a matching **EndBlock** exists, the matching **EndBlock** record **MUST** be written immediately before writing the **End** record of the current chart group.
- If a **StartBlock** record with **iObjectKind** set to 0x0000 without a matching **EndBlock** exists, the matching **EndBlock** record **MUST** be written immediately before writing the **End** record of the current axis group.
- If a **StartBlock** record with **iObjectKind** set to 0x000D without a matching **EndBlock** exists, the matching **EndBlock** record **MUST** be written immediately before writing the **End** record of the current sheet.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
frtHeaderOld																															
iObjectKind																unused1 (optional)															
unused2 (optional)																unused3 (optional)															

**frtHeaderOld (4 bytes):** An **FrtHeaderOld** structure (section [2.5.10](#)). The **frtHeaderOld.rt** field **MUST** be 0x0853.

**iObjectKind (2 bytes):** An unsigned integer that specifies the type of object that is encompassed by the block. **MUST** be equal to the **iObjectKind** field of the matching **StartBlock** record. **MUST** be a value from the following table.

Value	Type of object
0x0000	Axis group
0x0002	Attached label
0x0004	Axis

Value	Type of object
0x0005	Chart group
0x000D	Sheet

**unused1 (2 bytes):** Undefined and MUST be ignored.

**unused2 (2 bytes):** Undefined and MUST be ignored.

**unused3 (2 bytes):** Undefined and MUST be ignored.

#### 2.4.44 EndObject

This record specifies properties of a future record type (FRT) as defined by the chart sheet substream **ABNF**. The collection of records specifies a feature saved as an FRT so that an application that does not support the feature can preserve it. This record MUST have an associated **StartObject** record. **StartObject** and **EndObject** pairs can be nested. Up to 100 levels of blocks can be nested.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
frtHeaderOld																															
iObjectKind																unused1 (optional)															
unused2 (optional)																unused3 (optional)															

**frtHeaderOld (4 bytes):** An **FrtHeaderOld** structure (section [2.5.10](#)). The **frtHeaderOld.rt** field MUST be 0x0855.

**iObjectKind (2 bytes):** An unsigned integer that specifies the type of object that is encompassed by the block. MUST be equal to the **iObjectKind** field of the associated **StartObject** record. MUST be a value from the following table.

Value	Type of object
0x0010	<b>YMult</b>
0x0011	<b>FrtFontList</b>
0x0012	<b>DataLabExt</b>

**unused1 (2 bytes):** Undefined and MUST be ignored.

**unused2 (2 bytes):** Undefined and MUST be ignored.

**unused3 (2 bytes):** Undefined and MUST be ignored.

#### 2.4.45 EOF

This record specifies the end of the chart sheet substream in the workbook stream.

### 2.4.46 ExcludeColumns

This record specifies which columns of the data sheet are to be included or excluded from the chart.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
bData (variable)																															
...																															

**bData (variable):** An array of unsigned short integers indicating which data sheet columns are included or excluded from the chart. MUST be empty if no columns are included as part of the chart. The array contains zero-based column numbers at which the included or excluded status changes. The first number of the array is the first column included in the chart; the next number is the following column that is excluded from the chart; the next number is the following column that is included in the chart; and so on. There MUST be an even number of unsigned short integers in this field.

### 2.4.47 ExcludeRows

This record specifies which rows of the data sheet are to be included or excluded from the chart.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
bData (variable)																															
...																															

**bData (variable):** An array of unsigned short integers indicating which data sheet rows are included or excluded from the chart. MUST be empty if no rows are included as part of the chart. The total number of included rows MUST be less than or equal to 255. The array contains zero-based row numbers at which the included or excluded status changes. The first number of the array is the first row included in the chart; the next number is the following row that is excluded from the chart; the next number is the following row that is included in the chart; and so on. There MUST be an even number of unsigned short integers in this field.

### 2.4.48 Fbi

This record specifies the **font** information at the time the scalable font was added to the chart. <50>

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
dmixBasis																dmiyBasis															
twpHeightBasis																scab															
ifnt																															

**dmixBasis (2 bytes):** An unsigned integer that specifies the font width, in **twips**, when the font was first applied. MUST be greater than or equal to zero and less than or equal to 0x7FFF.

**dmixBasis (2 bytes):** An unsigned integer that specifies the font width, in **twips**, when the font was first applied. MUST be greater than or equal to zero and less than or equal to 0x7FFF.

**twpHeightBasis (2 bytes):** An unsigned integer that specifies the default font height, in twips. MUST be greater than or equal to 0x0014 and less than or equal to 0x1FF4.

**scab (2 bytes):** A Boolean that specifies the scale to use. MUST be a value from the following table.

Value	Meaning
0x0000	Scale by <b>chart area</b> .
0x0001	Scale by <b>plot area</b> .

**ifnt (2 bytes):** A **FontIndex** structure (section [2.5.5](#)) that specifies the font. MUST be used when this field is less than or equal to 0x00FF.

#### 2.4.49 Fbi2

This record specifies the **font** information at the time the scalable font was added to the chart. [<51>](#)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
dmixBasis																dmixBasis															
twpHeightBasis																scab															
ifnt																															

**dmixBasis (2 bytes):** An unsigned integer that specifies the font width, in **twips**, when the font was first applied. MUST be greater than or equal to zero and less than or equal to 0x7FFF.

**dmixBasis (2 bytes):** An unsigned integer that specifies the font height, in twips, when the font was first applied. MUST be greater than or equal to zero and less than or equal to 0x7FFF.

**twpHeightBasis (2 bytes):** An unsigned integer that specifies the default font height, in twips. MUST be greater than or equal to 0x0014 and less than or equal to 0x1FF4.

**scab (2 bytes):** A Boolean that specifies the scale to use. MUST be a value from the following table.

Value	Meaning
0x0000	Scale by <b>chart area</b> .
0x0001	Scale by <b>plot area</b> .

**ifnt (2 bytes):** A **FontIndex** structure (section [2.5.5](#)) that specifies the font. MUST be used when this field is greater than 0x00FF.

#### 2.4.50 Font

This record specifies a **font** and font formatting information.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31		
dyHeight																A	B	C	D	E	F	G	H	reserved									
icv																bls																	
sss																uls								bFamily									
bCharSet								unused3								fontName (variable)																	
...																																	

**dyHeight (2 bytes):** An unsigned integer that specifies the height of the font, in **twips**. MUST be greater than or equal to 0x0014 and less than 0x1FFF.

**A - unused1 (1 bit):** Undefined and MUST be ignored.

**B - fItalic (1 bit):** A bit that specifies whether the font is italic.

**C - unused2 (1 bit):** Undefined and MUST be ignored.

**D - fStrikeOut (1 bit):** A bit that specifies whether the font has **strikethrough formatting** applied.

**E - fOutline (1 bit):** A bit that specifies whether the font has an **outline effect** applied.

**F - fShadow (1 bit):** A bit that specifies whether the font has a **shadow effect** applied.

**G - fCondense (1 bit):** A bit that specifies whether the font is condensed. If this field is set to 1, the font is condensed.

**H - fExtend (1 bit):** A bit that specifies whether the font is extended. If this field is set to 1, the font is extended.

**reserved (8 bits):** MUST be set to zero, and MUST be ignored.

**icv (2 bytes):** An unsigned integer that specifies the color of the font. This value MUST be an **IcvFont** value (section [2.5.18](#)) or set to zero. A value of zero specifies a black font.

**bls (2 bytes):** An unsigned integer that specifies the font weight. This field MUST be a value from the following table [52](#).

Value	Meaning
0x0190	Normal font weight
0x02BC	Bold font weight

**sss (2 bytes):** An unsigned integer that specifies whether superscript, subscript, or normal script is used. MUST be a value from the following table.

Value	Meaning
0x0000	Normal script
0x0001	Superscript



Value	Meaning
0x0002	Subscript

**uls (1 byte):** An unsigned integer that specifies the underline style. MUST be a value from the following table.

Value	Meaning
0x0	No underline
0x1	Single underline
0x2	Double underline

**bFamily (1 byte):** An unsigned integer that specifies the font family, as defined by the **LOGFONT** structure described in [\[MSDN-FONTS\]](#). MUST be greater than or equal to zero and less than or equal to 0x05.

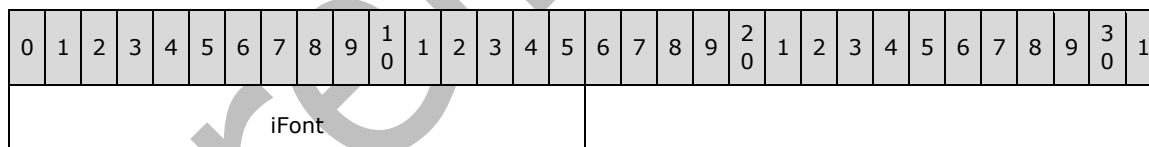
**bCharSet (1 byte):** An unsigned integer that specifies the **character set**, as defined by the **LOGFONT** structure described in [\[MSDN-FONTS\]](#).

**unused3 (1 byte):** Undefined and MUST be ignored.

**fontName (variable):** A **ShortXLUnicodeString** structure (section [2.5.27](#)) that specifies the name of this font. The string length MUST be greater than or equal to 0x1 and less than or equal to 0x1F. The **fontName.fHighByte** field MUST be set to 1. MUST NOT contain any null characters.

### 2.4.51 FontX

This record specifies the **font** for a text element. The font referenced by **iFont** can be in the chart sheet or in the **workbook** substream.



**iFont (2 bytes):** An unsigned integer that specifies the font to use for subsequent records. This font can be the default font of the chart, part of the collection of **Font** records following the **FrFontList** record, or part of the collection of **Font** records in the workbook. If **iFont** is set to 0x0000, this record specifies the default font of the chart. If this field is less than or equal to the number of **Font** records in the workbook, this field is a one-based index to a **Font** record in the workbook. Otherwise, this field is a one-based index into the collection of **Font** records in the chart sheet substream, where the index is equal to **iFont** minus  $n$  and  $n$  is the number of **Font** records in the workbook.

### 2.4.52 Format

This record specifies a **number format**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
ifmt																stFormat (variable)															
...																															

**ifmt (2 bytes):** An **IFmt** structure (section 2.5.19) that specifies the identifier of the **format string** specified by **stFormat**. The value of **ifmt.ifmt** MUST be a value in one of the following ranges or in the range from 0x017F through 0x0188.

- 0x0005 through 0x0008
- 0x0017 through 0x001A
- 0x0029 through 0x002C
- 0x003F through 0x0042
- 0x00A4 through 0x017E

**stFormat (variable):** An **XLUnicodeString** structure (section 2.5.30) that specifies the format string for this number format. The format string indicates how to format the numeric value of the **cell**. The length of this field MUST be greater than or equal to 1 character and less than or equal to 255 characters. For more information about how format strings are interpreted, see [ECMA-376] Part 4: section 3.8.31, Markup Language Reference.

The **ABNF** grammar for the format string is specified in [MS-XLS] section 2.4.126.

## 2.4.53 Frame

This record specifies the type, size, and position of the frame around a chart element as defined by the chart sheet substream **ABNF**. A chart element frame is specified by the following **Frame** record.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
frt																A	B	reserved													

**frt (2 bytes):** An unsigned integer that specifies the type of frame to be drawn. MUST be a value from the following table.

Value	Type of frame
0x0000	A frame surrounding the chart element.
0x0004	A frame with shadow surrounding the chart element.

**A - fAutoSize (1 bit):** A bit that specifies whether the size of the frame is automatically calculated. If this field is set to 1, the size of the frame is automatically calculated. In this case, the width and height specified by the chart element are ignored, and the size of the frame is calculated automatically. If this field is set to zero, the width and height specified by the chart element are used as the size of the frame.

**B - fAutoPosition (1 bit):** A bit that specifies whether the position of the frame is automatically calculated. If this field is set to 1, the position of the frame is automatically calculated. In this case, the x and y coordinates specified by the chart element are ignored, and the position of the

frame is automatically calculated. If this field is set to zero, the x and y coordinate location specified by the chart element is used as the position of the frame.

**reserved (14 bits):** MUST be set to zero, and MUST be ignored.

### 2.4.54 FrtFontList

This record specifies **font** information used on the chart and the beginning of a collection of **Font** records as defined by the chart sheet substream **ABNF**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
frtHeaderOld																															
verExcel								reserved														cFont									
rgFontInfo (variable)																															
...																															

**frtHeaderOld (4 bytes):** An **FrtHeaderOld** structure (section [2.5.10](#)). The **frtHeaderOld.rt** field MUST be set to 0x085A.

**verExcel (1 byte):** An unsigned integer that specifies the application version in which new chart elements were introduced that use the font information specified by **rgFontInfo**. MUST be equal to **iObjectInstance1** of the **StartObject** record that immediately follows this record as defined by the chart sheet substream ABNF, and MUST be a value from the following table.

Value	Meaning
0x09	This record pertains to new objects introduced in a specific version of the application <a href="#">&lt;53&gt;</a> . The <b>rgFontInfo</b> element specifies the font information that is used by <b>display units</b> labels specified by <b>YMult</b> .
0x0A	This record pertains to new objects introduced in specific version of the application <a href="#">&lt;54&gt;</a> . The <b>rgFontInfo</b> element specifies the font information that is used by extended data labels specified by <b>DataLabExt</b> .

**reserved (1 byte):** MUST be set to zero, and MUST be ignored.

**cFont (2 bytes):** An unsigned integer that specifies the number of items in **rgFontInfo**.

**rgFontInfo (variable):** An array of **FontInfo** structures (section [2.5.6](#)) that specify the font information. The number of elements in this array MUST be equal to the value specified in **cFont**.

### 2.4.55 FrtWrapper

This record wraps around a non-future record type (FRT) record and converts it into an **FRT** record.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
frtHeaderOld																															

wrappedRecord (variable)
...
frtWrapperPadding (variable)
...

**frtHeaderOld (4 bytes):** An **FrtHeaderOld** structure (section [2.5.10](#)). The **frtHeaderOld.rt** MUST be set to 0x0851.

**wrappedRecord (variable):** A **Font**, **Continue**, **LineFormat**, **AreaFormat**, **SeriesText**, **DefaultText**, **Text**, **FontX**, **ObjectLink**, **Frame**, **Begin**, **End**, **PicF**, **Pos**, **AIRuns**, **BRAI**, **Fbi**, or **GelFrame** that specifies the record being wrapped. These records MUST be wrapped in this **FrtWrapper** record if they are a part of a collection specified by **StartObject** and **EndObject**. These records appear in the file structure according to their record name, and not as **FrtWrapper** in the **ABNF** specified in the chart sheet substream.

**frtWrapperPadding (variable):** An array of bytes that is used to pad a **FrtWrapper**. Each element MUST be set to zero, and MUST be ignored. This field MUST exist if, and only if, the size of the **wrappedRecord** field is less than 8 bytes. If this field exists, its size MUST be equal to the value as specified by the following formula:

$$8 \text{ bytes} - (\text{size of } \mathbf{wrappedRecord})$$

The size of the padded **FrtWrapper** MUST NOT be less than 12 bytes.

## 2.4.56 GelFrame

This record specifies the properties of a **fill pattern** for parts of a chart. The record consists of an **OfficeArtFOPT**, as specified in [\[MS-ODRAW\]](#) section 2.2.9, and an **OfficeArtTertiaryFOPT**, as specified in [\[MS-ODRAW\]](#) section 2.2.11, which both contain properties for the fill pattern applied. If two consecutive **GelFrame** records appear, the second **GelFrame** is a continuation of the first.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
OPT1 (variable)																															
...																															
OPT2 (variable)																															
...																															

**OPT1 (variable):** An **OfficeArtFOPT**, as specified in [\[MS-ODRAW\]](#) section 2.2.9, specifies the primary properties of the fill pattern. MUST contain only the subset of **OfficeArtFOPT** properties specified in the following list:

- **fillType**, as specified in [\[MS-ODRAW\]](#) section 2.3.7.1.
- **fillColor**, as specified in [\[MS-ODRAW\]](#) section 2.3.7.2.
- **fillOpacity**, as specified in [\[MS-ODRAW\]](#) section 2.3.7.3.

- **fillBackColor**, as specified in [MS-ODRAW] section 2.3.7.4.
- **fillBackOpacity**, as specified in [MS-ODRAW] section 2.3.7.5.
- **fillCrMod**, as specified in [MS-ODRAW] section 2.3.7.6.
- **fillBlip\_complex**, as specified in [MS-ODRAW] section 2.3.7.8.
- **fillBlipName\_complex**, as specified in [MS-ODRAW] section 2.3.7.10.
- **fillBlipFlags**, as specified in [MS-ODRAW] section 2.3.7.11.
- **fillWidth**, as specified in [MS-ODRAW] section 2.3.7.12.
- **fillHeight**, as specified in [MS-ODRAW] section 2.3.7.13.
- **fillAngle**, as specified in [MS-ODRAW] section 2.3.7.14.
- **fillFocus**, as specified in [MS-ODRAW] section 2.3.7.15.
- **fillToLeft**, as specified in [MS-ODRAW] section 2.3.7.16.
- **fillToTop**, as specified in [MS-ODRAW] section 2.3.7.17.
- **fillToRight**, as specified in [MS-ODRAW] section 2.3.7.18.
- **fillToBottom**, as specified in [MS-ODRAW] section 2.3.7.19.
- **fillRectLeft**, as specified in [MS-ODRAW] section 2.3.7.20.
- **fillRectTop**, as specified in [MS-ODRAW] section 2.3.7.21.
- **fillRectRight**, as specified in [MS-ODRAW] section 2.3.7.22.
- **fillRectBottom**, as specified in [MS-ODRAW] section 2.3.7.23.
- **fillDztype**, as specified in [MS-ODRAW] section 2.3.7.24.
- **fillShadePreset**, as specified in [MS-ODRAW] section 2.3.7.25.
- **fillShadeColors\_complex**, as specified in [MS-ODRAW] section 2.3.7.27.
- **fillOriginX**, as specified in [MS-ODRAW] section 2.3.7.28.
- **fillOriginY**, as specified in [MS-ODRAW] section 2.3.7.29.
- **fillShapeOriginX**, as specified in [MS-ODRAW] section 2.3.7.30.
- **fillShapeOriginY**, as specified in [MS-ODRAW] section 2.3.7.31.
- **fillShadeType**, as specified in [MS-ODRAW] section 2.3.7.32.
- **fFilled**, as specified in [MS-ODRAW] section 2.3.7.43.
- **fHitTestFill**, as specified in [MS-ODRAW] section 2.3.7.43.
- **fillShape**, as specified in [MS-ODRAW] section 2.3.7.43.
- **fillUseRect**, as specified in [MS-ODRAW] section 2.3.7.43.
- **fNoFillHitTest**, as specified in [MS-ODRAW] section 2.3.7.43.

**OPT2 (variable):** An **OfficeArtTertiaryFOPT**, as specified in [MS-ODRAW] section 2.2.11 specifies the additional properties of the fill pattern. MUST contain only the subset of **OfficeArtTertiaryFOPT** properties specified in the following list<55>:

- **fillColorExt**, as specified in [MS-ODRAW] section 2.3.7.33.
- **fillColorExtMod**, as specified in [MS-ODRAW] section 2.3.7.35.
- **fillBackColorExt**, as specified in [MS-ODRAW] section 2.3.7.37.
- **fillBackColorExtMod**, as specified in [MS-ODRAW] section 2.3.7.39.
- **fRecolorFillAsPicture**, as specified in [MS-ODRAW] section 2.3.7.43.
- **fUseShapeAnchor**, as specified in [MS-ODRAW] section 2.3.7.43.

### 2.4.57 IFmtRecord

This record specifies the **number format** to use for the text on an axis.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
ifmt																															

**ifmt (2 bytes):** An **IFmt** structure (section 2.5.19) that specifies the number format identifier.

### 2.4.58 Label

This record specifies a **cell** that contains a string constant.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rw																col															
reserved										ifmt										stLabel (variable)											
...																															

**rw (2 bytes):** A **Graph\_Rw** structure (section 2.5.15) that specifies the zero-based row number of the cell that contains the string constant.

**col (2 bytes):** A **Graph\_Col** structure (section 2.5.14) that specifies the zero-based column number of the cell that contains the string constant.

**reserved (1 byte):** MUST be set to zero, and MUST be ignored.

**ifmt (2 bytes):** An **IFmt** structure (section 2.5.19) that specifies the formatting for the data.

**stLabel (variable):** A **ShortXLUnicodeString** structure (section 2.5.27) that contains the string constant.

## 2.4.59 Legend

This record specifies properties of a legend and the beginning of a collection of records defined by chart sheet substream **ABNF**. The collection of records specifies a legend.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31						
x																																					
y																																					
dx																																					
dy																																					
unused								wSpace								A	B	C	D	E	F	reserved2															

**x (4 bytes):** An unsigned integer that specifies the x position, in SPRC, of the upper-left corner of the **bounding rectangle** of the legend. MUST be ignored, and the **x1** field of the following **Pos** record MUST be used instead.

**y (4 bytes):** An unsigned integer that specifies the y position, in SPRC, of the upper-left corner of the bounding rectangle of the legend. MUST be ignored, and the **y1** field of the following **Pos** record MUST be used instead.

**dx (4 bytes):** An unsigned integer that specifies the width, in SPRC, of the bounding rectangle of the legend. MUST be ignored, and the **x2** field of the following **Pos** record MUST be used instead.

**dy (4 bytes):** An unsigned integer that specifies the height, in SPRC, of the bounding rectangle of the legend. MUST be ignored, and the **y2** field of the following **Pos** record MUST be used instead.

**unused (1 byte):** Undefined and MUST be ignored.

**wSpace (1 byte):** An unsigned integer that specifies the space between **legend entries**. MUST be set to 0x01, which represents 40 **twips** between legend entries.

**A - fAutoPosition (1 bit):** A bit that specifies whether the legend is automatically positioned. If this field is set to 0x1, **fAutoPosX** MUST be set to 0x1, and **fAutoPosY** MUST be set to 0x1.

**B - reserved1 (1 bit):** MUST be set to 1, and MUST be ignored.

**C - fAutoPosX (1 bit):** A bit that specifies whether the positioning of the legend on the x axis is automatic.

**D - fAutoPosY (1 bit):** A bit that specifies whether the positioning of the legend on the y axis is automatic.

**E - fVert (1 bit):** A bit that specifies the layout of the legend entries. MUST be set to 0x1 if **fWasDataTable** is set to 0x1. MUST be a value from the following table.

Value	Meaning
0x0	The legend contains multiple columns of legend entries, or the size of the legend has been manually changed from the default size.

Value	Meaning
0x1	The legend contains a single column of legend entries.

**F - fWasDataTable (1 bit):** A bit that specifies whether the legend is shown in a data table.

**reserved2 (10 bits):** MUST be set to zero, and MUST be ignored.

#### 2.4.60 LegendException

This record specifies information about a **legend entry** that has been changed from the default legend entry settings and specifies the beginning of a collection of records as defined by the chart sheet substream **ABNF**. The collection of records specifies legend entry formatting. On a chart on which the legend contains legend entries for the series and trendlines, as defined in the legend overview, there MUST be zero or one instances of this record in the sequence of records that conforms to the SERIESFORMAT rule.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
iss																A	B	reserved													

**iss (2 bytes):** An unsigned integer that specifies the legend entry.

This field has different interpretations, depending on the content of the legend in the chart. The legend overview specifies the types of content the legend can contain.

- On a chart on which the legend contains legend entries for the series and trendlines, this field MUST be set to 0xFFFF. This record specifies the legend entry of the series or trendline that contains this record.
- On a chart on which the legend contains legend entries for each data point on the chart, this field specifies the zero-based index of a legend entry in the legend, where 0x0000 is the legend entry for the first data point in the series.
- On a chart with a surface chart group, this field specifies the zero-based index of a legend entry in the legend, where 0x0000 is the legend entry for the lowest band of the surface chart group.

**A - fDeleted (1 bit):** A bit that specifies whether the legend entry specified by **iss** has been deleted.

**B - fLabel (1 bit):** A bit that specifies whether the legend entry specified by **iss** has been formatted. If this field is set to 1, there MUST be a sequence of records that conforms to the ATTACHEDLABEL rule in the chart sheet substream ABNF following this record.

**reserved (14 bits):** MUST be set to zero, and MUST be ignored.

#### 2.4.61 Line

This record specifies that the chart group is a line chart group and specifies the chart group attributes.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
A	B	C	reserved																												



**A - fStacked (1 bit):** A bit that specifies whether the data points in the chart group that share the same **category** are stacked one on top of the next.

**B - f100 (1 bit):** A bit that specifies whether the data points in the chart group are displayed as a percentage of the sum of all data points in the chart group that share the same category. MUST be set to zero if **fStacked** is set to zero.

**C - fHasShadow (1 bit):** A bit that specifies whether one or more **data markers** in the chart group has shadows.

**reserved (13 bits):** MUST be set to zero, and MUST be ignored.

### 2.4.62 LineFormat

This record specifies the appearance of a line.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rgb																															
Ins																we															
A	B	C	D	reserved2														icv													

**rgb (4 bytes):** A **LongRGB** structure (section [2.5.20](#)) that specifies the color of the line. The color MUST be the same as the color specified by **icv**.

**Ins (2 bytes):** A signed integer that specifies the style of the line. MUST be a value from the following table.

Value	Meaning
0x0000	Solid
0x0001	Dash
0x0002	Dot
0x0003	Dash-dot
0x0004	Dash dot dot
0x0005	None
0x0006	Dark gray pattern
0x0007	Medium gray pattern
0x0008	Light gray pattern

When this field is set to 0x0005 (None), the values of **we** and **icv** MUST be set to the values in the following table.

Attribute	Default value
Line thickness ( <b>we</b> )	0x0000 (Narrow)
Line color ( <b>icv</b> )	0x004D

**we (2 bytes):** A signed integer that specifies the thickness of the line. MUST be a value from the following table.

Value	Meaning
0xFFFF (-1)	Hairline
0x0000	Narrow (single)
0x0001	Medium (double)
0x0002	Wide (triple)

**A - fAuto (1 bit):** A bit that specifies whether the line has default formatting.

If this field is set to zero, the line has formatting as specified by **lns**, **we**, and **icv**.

If this field is set to 1, **lns**, **we**, **icv**, and **rgb** MUST be ignored, and default values are used as specified in the following table.

Attribute	Default value
Line pattern ( <b>lns</b> )	0x0000 (Solid)
Line thickness ( <b>we</b> )	0x0000 (Narrow)
Line color ( <b>icv</b> )	0x004D
Line color ( <b>rgb</b> )	The same as the default color used for <b>icv</b> .

**B - reserved1 (1 bit):** MUST be set to zero, and MUST be ignored.

**C - fAxisOn (1 bit):** A bit that specifies whether the axis line is displayed.

If the previous record is an **AxisLine** record and the value of the **id** field of the **AxisLine** record is set to 0x0000, this field MUST be set to a value from the following table.

fAxisOn	lns	Meaning
0	0x0005	The axis line is not displayed.
0	Any legal value except 0x0005	The axis line is displayed.
1	Any legal value	The axis line is displayed.

If the previous record is not an **AxisLine** record with the **id** field set to 0x0000, this field MUST be set to zero and MUST be ignored.

**D - fAutoCo (1 bit):** A bit that specifies whether **icv** is set to 0x004D. If this field is set to 1, **icv** MUST be set to 0x004D. If this field is set to zero, **icv** MUST NOT be set to 0x004D.

**reserved2 (12 bits):** MUST be set to zero, and MUST be ignored.

**icv (2 bytes):** An unsigned integer that specifies the color of the line. The value MUST [<56>](#) be an **IcvChart** value (section [2.5.17](#)). The color MUST be the same as the color specified by **rgb**.

### 2.4.63 LinkedSelection

This record specifies where in the data sheet window to paste the **selection** from the OLE stream.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rwFirst																rwLast															

colFirst	colLast
----------	---------

**rwFirst (2 bytes):** A **Graph\_Rw** structure (section 2.5.15) that specifies the first row in the data sheet window into which to paste the selection from the OLE stream. MUST be set to 0x0000 if the first row of the selection from the OLE stream contains any non-numeric values. MUST be set to 0x0001 if the first row of the selection from the OLE stream contains only numeric values.

**rwLast (2 bytes):** A **Graph\_Rw** structure (section 2.5.15) that MUST be the same as **rwFirst**.

**colFirst (2 bytes):** A **Graph\_Col** structure (section 2.5.14) that specifies the first column in the data sheet window into which to paste the selection from the OLE stream. MUST be set to 0x0000 if the first column of the selection from the OLE stream contains any non-numeric values. MUST be set to 0x0001 if the first column of the selection from the OLE stream contains only numeric values.

**colLast (2 bytes):** A **Graph\_Col** structure (section 2.5.14) that MUST be the same as **colFirst**.

### 2.4.64 MainWindow

This record specifies the location of the **OLE server** window in the parent document window when the chart data was saved.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
wLeft																wTop															
wWidth																wHeight															

**wLeft (2 bytes):** A signed integer that specifies the location, in **twips**, of the left edge of the window relative to the left edge of the primary monitor.

**wTop (2 bytes):** A signed integer that specifies the location, in twips, of the top edge of the window relative to the top edge of the primary monitor.

**wWidth (2 bytes):** A signed integer that specifies the width of the window, in twips. MUST be greater than or equal to zero.

**wHeight (2 bytes):** A signed integer that specifies the height of the window, in twips. MUST be greater than or equal to zero.

### 2.4.65 MarkerFormat

This record specifies the color, size, and shape of the associated **data markers** that appear on line, radar, and scatter chart groups. The associated data markers are specified by the preceding **DataFormat** record. If this record is not present in the sequence of records that conforms to the SS rule, all fields have default values. Otherwise, all the fields MUST contain a value.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31				
rgbFore																																			
rgbBack																																			
imk																A	B	C	D	reserved2															

icvFore	icvBack
miSize	

**rgbFore (4 bytes):** A **LongRGB** structure (section [2.5.20](#)) that specifies the **border** color of the data marker. The color **MUST** be the same as the color specified by **icvFore**. The default value of this field is automatically selected from the next available color in the Chart color table.

**rgbBack (4 bytes):** A **LongRGB** structure (section [2.5.20](#)) that specifies the interior color of the data marker. The color **MUST** be the same as the color specified by **icvBack**. The default value of this field is the same as the default value of **rgbFore** when the default **imk** is set to 0x0001, 0x0002, 0x0003, or 0x0008. Otherwise, the default value is 0xFFFFFFFF.

**imk (2 bytes):** An unsigned integer that specifies the type of data marker. The default value for this field is automatically selected from the list of data marker type and cannot be set to 0x0000. **MUST** be a value from the following table.

Value	Meaning
0x0000	No marker.
0x0001	Square markers.
0x0002	Diamond-shaped markers.
0x0003	Triangular markers.
0x0004	Square markers with an X.
0x0005	Square markers with an asterisk (*).
0x0006	Short bar markers.
0x0007	Long bar markers.
0x0008	Circular markers.
0x0009	Square markers with a plus sign (+).

**A - fAuto (1 bit):** A bit that specifies whether the data marker is automatically generated. The default value is 1. **MUST** be a value from the following table.

Value	Meaning
Zero	The data marker is not automatically generated.
1	The data marker type, size, and color are automatically generated, and the values are set accordingly.

**B - reserved1 (3 bits):** **MUST** be set to zero, and **MUST** be ignored.

**C - fNotShowInt (1 bit):** A bit that specifies whether to show the data marker interior. The default value is zero. **MUST** be a value from the following table.

Value	Meaning
Zero	The data marker interior is shown.
1	The data marker interior is not shown.

**D - fNotShowBrd (1 bit):** A bit that specifies whether to show the data marker border. The default value is zero. **MUST** be a value from the following table.

Value	Meaning
Zero	The data marker border is shown.
1	The data marker border is not shown.

**reserved2 (10 bits):** MUST be set to zero, and MUST be ignored.

**icvFore (2 bytes):** An unsigned integer that specifies the border color of the data marker. The value MUST [<57>](#) be an **IcvChart** value (section [2.5.17](#)). The color MUST be the same as the color specified by **rgbFore**. The default value is automatically set to be the same as the color specified by **rgbFore**.

**icvBack (2 bytes):** An unsigned integer that specifies the interior color of the data marker. The value MUST [<58>](#) be an **IcvChart** value (section [2.5.17](#)). The color MUST be the same as the color specified by **rgbBack**. The default value is automatically set to be the same as the color specified by **rgbBack**.

**miSize (4 bytes):** An unsigned integer that specifies the size, in **twips**, of the data marker. MUST be greater than or equal to 0x0028 and less than or equal to 0x05A0. The default value for this field is 0x0064.

#### 2.4.66 MaxStatus

This record is unused.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	1	2	3	4	5	6	7	8	9	30	1
unused1										unused2																					

**unused1 (1 byte):** Undefined and MUST be ignored.

**unused2 (1 byte):** Undefined and MUST be ignored.

#### 2.4.67 MsoDrawing

This record specifies a **drawing object** and the beginning of a collection of **Continue** records as defined by the chart sheet substream **ABNF**. The collection of **Continue** records specifies additional feature data to complete this record.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	1	2	3	4	5	6	7	8	9	30	1		
rgChildRec (variable)																																	
...																																	

**rgChildRec (variable):** An **OfficeArtDgContainer** as specified in [\[MS-ODRAW\]](#) that specifies the drawing object.

#### 2.4.68 MsoDrawingGroup

This record specifies a group of **drawing objects** and the beginning of a collection of **Continue** records. The collection of **Continue** records specifies additional feature data to complete this record.

0	1	2	3	4	5	6	7	8	9	1	0	1	2	3	4	5	6	7	8	9	2	0	1	2	3	4	5	6	7	8	9	3	0	1
rgChildRec (variable)																																		
...																																		

**rgChildRec (variable):** An **OfficeArtDggContainer** as specified in [\[MS-ODRAW\]](#) that specifies the group of drawing objects.

### 2.4.69 MsoDrawingSelection

This record specifies **selected drawing objects** and the drawing objects in focus on the **sheet**.

0	1	2	3	4	5	6	7	8	9	1	0	1	2	3	4	5	6	7	8	9	2	0	1	2	3	4	5	6	7	8	9	3	0	1
selection (variable)																																		
...																																		

**selection (variable):** An **OfficeArtFDGSL** as specified in [\[MS-ODRAW\]](#) that specifies the selected drawing objects and the drawing objects in focus on the sheet.

### 2.4.70 Number

This record specifies a **cell** that contains an **Xnum** structure (section [2.5.33](#)).

0	1	2	3	4	5	6	7	8	9	1	0	1	2	3	4	5	6	7	8	9	2	0	1	2	3	4	5	6	7	8	9	3	0	1
rw															col																			
reserved										ifmt										num														
...																																		
...																																		

**rw (2 bytes):** A **Graph\_Rw** structure (section [2.5.15](#)) that specifies the zero-based row number of the cell that contains an **Xnum** structure (section [2.5.33](#)).

**col (2 bytes):** A **Graph\_Col** structure (section [2.5.14](#)) that specifies the zero-based column number of the cell that contains an **Xnum** structure (section [2.5.33](#)).

**reserved (1 byte):** MUST be set to zero, and MUST be ignored.

**ifmt (2 bytes):** An **IFmt** structure (section [2.5.19](#)) that specifies the number formatting for the data.

**num (8 bytes):** An **Xnum** structure (section [2.5.33](#)) that specifies the cell value.

### 2.4.71 Obj

This record specifies the properties of an object on a **graph object**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
cmo (22 bytes)																															
...																															
...																															
...																gmo (optional)															
...																															
pictFormat (optional)																															
...																reserved1 (optional)															
reserved2 (optional)																unused (optional)															
reserved3 (optional)																															

**cmo (22 bytes):** An **FtCmo** structure (section [2.5.12](#)) that specifies the common properties of this object.

**gmo (6 bytes):** An optional **FtGmo** structure (section [2.5.13](#)) that specifies the properties of this group object. This field **MUST** exist if, and only if, **cmo.ot** is set to 0x0000.

**pictFormat (6 bytes):** An optional **FtCf** structure (section [2.5.11](#)) that specifies the format of this picture object. This field **MUST** exist if, and only if, **cmo.ot** is set to 0x0008.

**reserved1 (2 bytes):** Optional. This field **MUST** exist if, and only if, **cmo.ot** is set to 0x0008. If **cmo.ot** is set to 0x08, this field **MUST** be set to 0x0008 and **MUST** be ignored.

**reserved2 (2 bytes):** Optional. This field **MUST** exist if, and only if, **cmo.ot** is set to 0x0008. If **cmo.ot** is set to 0x08, this field **MUST** be set to 0x0002 and **MUST** be ignored.

**unused (2 bytes):** Optional. This field **MUST** exist if, and only if, **cmo.ot** is set to 0x0008. If **cmo.ot** is set to 0x0008, this field is undefined and **MUST** be ignored.

**reserved3 (4 bytes):** **MUST** be set to zero, and **MUST** be ignored.

## 2.4.72 ObjectLink

This record specifies an object on a chart or the entire chart, to which a **Text** record is linked.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
wLinkObj																wLinkVar1															
wLinkVar2																															

**wLinkObj (2 bytes):** An unsigned integer that specifies the object to which the **Text** is linked. MUST be a value from the following table.

Value	Meaning
0x0001	Entire chart.
0x0002	Value axis, or vertical value axis on bubble and scatter chart groups.
0x0003	<b>category</b> axis, or horizontal value axis on bubble and scatter chart groups.
0x0004	Series or data points.
0x0007	Series axis.
0x000C	<b>Display units</b> labels of an axis.

**wLinkVar1 (2 bytes):** An unsigned integer that specifies the zero-based index into a **Series** record in the collection of **Series** records in the current chart sheet substream. Each referenced **Series** record specifies a series for the chart group to which the **Text** is linked. When the **wLinkObj** field is set to 0x0004, this value MUST be less than or equal to 0x00FE. When the **wLinkObj** field is not set to 0x0004, this field MUST be set to 0x0000, and MUST be ignored.

**wLinkVar2 (2 bytes):** An unsigned integer that specifies the zero-based index into the category in the series specified by **wLinkVar1**, to which the **Text** is linked. When the **wLinkObj** field is set to 0x0004, if the **Text** is linked to a series instead of a single data point, the value MUST be set to 0xFFFF; if the **Text** is linked to a data point, the value MUST be less than or equal to 0x7CFF. When the **wLinkObj** field is not set to 0x0004, this field MUST be set to 0x0000, and MUST be ignored.

### 2.4.73 Orient

This record specifies how the series data of a chart is arranged.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
fSeriesInRows										rowSeriesX										colSeriesX											
...										Reserved3																					

**fSeriesInRows (1 byte):** A Boolean that specifies whether series are arranged by rows or columns from the data specified in a data sheet window. MUST be a value from the following table.

Value	Meaning
0x00	Series are arranged by columns.
0x01	Series are arranged by rows.

**rowSeriesX (2 bytes):** An unsigned integer that specifies a zero-based row index into the data sheet. The referenced row is used to calculate axis values along the horizontal value axis in a scatter chart group. MUST be set to 0x0000 for other chart group types. MUST be set to **colSeriesX** and MUST be ignored if **fSeriesInRows** is set to 0x00.

**colSeriesX (2 bytes):** An unsigned integer that specifies a zero-based column index into the data sheet. The referenced column is used to calculate axis values along the horizontal value axis in a



scatter chart group. MUST be set to 0x0000 for other chart group types. MUST be set to **rowSeriesX** and MUST be ignored if **fSeriesInRows** is set to 0x01.

**Reserved3 (1 byte):** MUST be set to 0x01, and MUST be ignored.

### 2.4.74 Palette

This record specifies a custom **color palette**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
ccv																rgColor (variable)															
...																															

**ccv (2 bytes):** A signed integer that specifies the number of colors in the **rgColor** array. This value MUST be set to 0x0038.

**rgColor (variable):** An array of **LongRGB** values (section [2.5.20](#)) that specifies the colors of the color palette. The number of items in the array MUST be equal to the value specified in the **ccv** field.

### 2.4.75 PicF

This record specifies the layout of a picture attached to a picture-filled chart element.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
ptyp																unused															
reserved1										A	B	C	D			numScale															
...																															

**ptyp (2 bytes):** An unsigned integer that specifies the picture layout. If this record is not located in the sequence of records that conforms to the SS rule, this field MUST be set to 0x0001. If this record is located in the SS rule, MUST be a value from the following table.

Value	Meaning
0x0001	Stretched. The picture is scaled to fit in the filled areas of the chart element.
0x0002	Stacked. The pictures in the data points are stacked on top of each other in the direction of the value axis.
0x0003	Stacked and scaled. The pictures in the data points are stacked next to or on top of each other, and each picture is scaled to fit in the number of units on the value axis as specified by <b>numScale</b> .

**unused (2 bytes):** Undefined and MUST be ignored.

**reserved1 (9 bits):** MUST be set to zero, and MUST be ignored.

**A - fTopBottom (1 bit):** A bit that specifies whether the picture covers the top and bottom fill areas of the data points. The top and bottom fill areas of the data points are parallel to the **floor** in a 3-D **plot area**. If a **Chart3d** record does not exist in the chart sheet substream, this record is not in a sequence of records that conforms to the SS rule, or this record is in an SS rule that contains a **Chart3DBarShape** with the **riser** field set to 0x01, this field **MUST** be set to 0x1.

**B - fBackFront (1 bit):** A bit that specifies whether the picture covers the front and back fill areas of the data points on a bar or column chart group. If a **Chart3d** record does not exist in the chart sheet substream, this record is not in a sequence of records that conforms to the SS rule, or this record is in an SS rule that contains a **Chart3DBarShape** with the **riser** field set to 0x01, this field **MUST** be set to 0x1.

**C - fSide (1 bit):** A bit that specifies whether the picture covers the side fill areas of the data points on a bar or column chart group. If a **Chart3d** record does not exist in the chart sheet substream, this record is not in a sequence of records that conforms to the SS rule, or this record is in an SS rule that contains a **Chart3DBarShape** with the **riser** field set to 0x01, this field **MUST** be set to 0x1.

**D - reserved2 (4 bits):** **MUST** be set to zero, and **MUST** be ignored.

**numScale (8 bytes):** An **Xnum** structure (section 2.5.33) that specifies the number of units on the value axis into which to fit the entire picture. The picture is scaled to fit within this number of units. If the value of **ptyp** is not set to 0x0003, this field is undefined and **MUST** be ignored.

## 2.4.76 Pie

This record specifies that the chart group is a pie chart group or a doughnut chart group, and specifies the chart group attributes.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
anStart																pcDonut															
A	B	reserved																													

**anStart (2 bytes):** An unsigned integer that specifies the starting angle of the first data point, clockwise from the top of the circle. **MUST** be less than or equal to 0x0168.

**pcDonut (2 bytes):** An unsigned integer that specifies the size of the center hole in a doughnut chart group as a percentage of the **plot area** size. **MUST** be a value from the following table.

Value	Meaning
0x0000	Pie chart group
0x000A to 0x005A	Doughnut chart group

**A - fHasShadow (1 bit):** A bit that specifies whether one or more data points in the chart group has shadows.

**B - fShowLdrLines (1 bit):** A bit that specifies whether the **leader lines** to the data labels are shown.

**reserved (14 bits):** **MUST** be set to zero, and **MUST** be ignored.

### 2.4.77 PieFormat

This record specifies the distance of a data point or data points in a series from the center of one of the following:

- The **plot area** for a doughnut or pie chart group.
- The **primary pie** in a pie of pie or bar of pie chart group.
- The **secondary bar/pie** of a pie of pie chart group.

The data point or data points in a series are specified by the sequence of records that conforms to the SS rule that contains this record.

MUST not exist on chart group types other than pie, doughnut, bar of pie, or pie of pie. MUST NOT exist if the chart group type is doughnut and the series is not the outermost series. MUST NOT exist for the data points on the secondary bar/pie of a bar of pie chart group.

0	1	2	3	4	5	6	7	8	9	1	0	1	2	3	4	5	6	7	8	9	2	0	1	2	3	4	5	6	7	8	9	3	0	1
pcExplode																																		

**pcExplode (2 bytes):** A signed integer that specifies the distance of a data point or data points in a series from the center of one of the following:

- The plot area for a doughnut or pie chart group.
- The primary pie in a pie of pie or bar of pie chart group.
- The secondary bar/pie of a pie of pie chart group.

The value of this field specifies the distance as a percentage. If this value is 0x0000, the data point or data points in a series are as close to the center as possible for the chart group type. If this value is set to 0x0064, the data point is at the edge of the **chart area**. If this value is greater than 0x0064 so that the data point is beyond the edge of the chart area, all the data points in the chart group are scaled down to fit inside the chart area so that the data point with the highest **pcExplode** value is at the edge of the chart area.

MUST be greater than or equal to 0x0000.

### 2.4.78 PlotArea

This empty record specifies that the following **Frame** record specifies properties of the **plot area**.

### 2.4.79 PlotGrowth

This record specifies the scale factors for **font scaling** in the **plot area**. The values from the **Fbi** record are used in conjunction with this record to render the scaled fonts in the plot area. If there is no **Fbi** record in which **Fbi.scab** is set to 0x0001, this record is unused and MUST be ignored.

If the scaled font size is the same as the font size of the **Font** record (as specified by **Fbi.ifnt** and **Font.dyHeight**), that **Font** record is used to render the scaled fonts. If the scaled font size is not the same as the size of the **Font** record (as specified by **Fbi.ifnt**), a new **Font** record is added to the font table, and the following font scaling algorithm is used to determine the scaled font size:

1. Multiply **Fbi.twpHeightBasis** by **dxPlotGrowth**, and call this **twpX**.
2. Multiply **Fbi.twpHeightBasis** by **dyPlotGrowth**, and call this **twpY**.

3. Take the smaller of **twpX** and **twpY**, and call this **twpNew**. If this value is less than 10, set it to 10.
4. Round this value from **twips** to the nearest quarter **point**.
5. If **Fbi.twpHeightBasis** is in the range from 0x00A0 through 0x00F0 and **twpNew** is in the range from 0x0078 through 0x00A0, set **twpNew** to 0x00A0. If **Fbi.twpHeightBasis** is in the range from 0x00A0 through 0x00F0 and **twpNew** is in the range from 0x00F0 through 0x0118, set **twpNew** to 0x00F0.
6. The **twpNew** value is the new scaled font size. If the **dyHeight** field of the **Font** record (as specified by **Fbi.ifnt**) is the same as **twpNew**, load that **Font** record for the scaled font. If it is not the same, **twpNew** is the new scaled font size, and a new **Font** record is added to the font table.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
dxPlotGrowth																															
dyPlotGrowth																															

**dxPlotGrowth (4 bytes):** A **FixedPoint** as specified in [MS-OSHARED] section 2.2.1.6 that specifies the horizontal growth, in points, of the plot area for font scaling.

**dyPlotGrowth (4 bytes):** A **FixedPoint** as specified in [MS-OSHARED] section 2.2.1.6 that specifies the vertical growth, in points, of the plot area for font scaling.

#### 2.4.80 Pos

This record specifies the size and position of a legend, an attached label, or the **plot area**, as specified by the primary axis group. This record **MUST** be ignored for the plot area when the **fManPlotArea** field of **ShtProps** is set to 1.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
mdTopLt																mdBotRt															
x1																unused1															
y1																unused2															
x2																unused3															
y2																unused4															

**mdTopLt (2 bytes):** A **PositionMode** structure (section 2.5.25) that specifies the positioning mode for the upper-left corner of a legend, an attached label, or the plot area. The valid combinations of **mdTopLt** and **mdBotRt** and the meaning of **x1**, **y1**, **x2**, **y2** are specified in the Valid Combinations of mdTopLt and mdBotRt by Type table.

**mdBotRt (2 bytes):** A **PositionMode** structure (section 2.5.25) that specifies the positioning mode for the lower-right corner of a legend, an attached label, or the plot area. The valid combinations of **mdTopLt** and **mdBotRt** and the meaning of **x1**, **y1**, **x2**, **y2** are specified in the following table:

Type	Position mode of mdTopLt	Position mode of mdBotRt	Meaning
Plot area (axis group)	<b>MDPARENT</b>	<b>MDPARENT</b>	The <b>x1</b> and <b>y1</b> coordinates specify the horizontal and vertical offsets of the upper-left corner of the primary axis group, relative to the upper-left corner of the <b>chart area</b> , in SPRC. The <b>x2</b> and <b>y2</b> coordinates specify the width and height of the primary axis group, in SPRC.
Legend	<b>MDCHART</b>	<b>MDABS</b>	The <b>x1</b> and <b>y1</b> coordinates specify the horizontal and vertical offsets of the upper-left corner of the legend, relative to the upper-left corner of the chart area, in SPRC. The <b>x2</b> and <b>y2</b> coordinates specify the width and height of the legend, in <b>points</b> .
Legend	<b>MDCHART</b>	<b>MDPARENT</b>	The <b>x1</b> and <b>y1</b> coordinates specify the horizontal and vertical offsets of the upper-left corner of the legend, relative to the upper-left corner of the chart area, in SPRC. The <b>x2</b> and <b>y2</b> coordinates <b>MUST</b> be ignored. The size of the legend is determined by the application.
Legend	<b>MDKTH</b>	<b>MDPARENT</b>	The <b>x1</b> , <b>y1</b> , <b>x2</b> , and <b>y2</b> coordinates <b>MUST</b> be ignored. The legend is located in a data table.
Attached label	<b>MDPARENT</b>	<b>MDPARENT</b>	The <b>x1</b> and <b>y1</b> coordinates are specified in the following table. The <b>x2</b> and <b>y2</b> coordinates <b>MUST</b> be ignored. The size of the attached label is determined by the application.

The following table describes types of attached labels. For more information about determining the type of an attached label, see **Charts** (section [2.2.1.3](#))

Type of attached label	Meaning
Chart title	The <b>x1</b> and <b>y1</b> coordinates specify the horizontal and vertical offset of the title, relative to its default position, in SPRC.
Axis title	The <b>x1</b> and <b>y1</b> coordinates specify the offset of the title along the direction of the axis. The <b>x1</b> coordinate specifies an offset along the <b>category</b> axis, date axis, or horizontal value axis. The <b>y1</b> coordinate specifies an offset along the value axis. Both offsets are relative to the default position of the title, in 1000ths of the axis length.
Data label	<p>If the chart is not a pie or a radar chart group, the <b>x1</b> and <b>y1</b> coordinates specify the offset of the label along the direction of the axis. The <b>x1</b> coordinate is an offset along the category axis, date axis, or horizontal value axis. The <b>y1</b> coordinate is an offset along the value axis, opposite to the direction of the axis. Both offsets are relative to the default position of the label, in 1000ths of the axis length.</p> <p>For a pie chart group, the <b>x1</b> coordinate specifies the clockwise angle, in degrees, and the <b>y1</b> coordinate specifies the radius offset of the label relative to its default position, in 1000ths of the pie radius length. A label moved towards the pie center has a negative radius offset.</p> <p>For a radar chart group, the <b>x1</b> and <b>y1</b> coordinates specify the horizontal and vertical offset of the label relative to its default position, in 1000ths of the axis length.</p>

**x1 (2 bytes):** A signed integer that specifies a position. The meaning is specified in the first table in this section.

**unused1 (2 bytes):** Undefined and MUST be ignored.

**y1 (2 bytes):** A signed integer that specifies a position. The meaning is specified in the first table in this section.

**unused2 (2 bytes):** Undefined and MUST be ignored.

**x2 (2 bytes):** A signed integer that specifies a width. The meaning is specified in the first table in this section.

**unused3 (2 bytes):** Undefined and MUST be ignored.

**y2 (2 bytes):** A signed integer that specifies a height. The meaning is specified in the first table in this section.

**unused4 (2 bytes):** Undefined and MUST be ignored.

### 2.4.81 Radar

This record specifies that the chart group is a radar chart group, and specifies the chart group attributes.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
A	B	reserved														unused															

**A - fRdrAxLab (1 bit):** A bit that specifies whether **category** labels are displayed.

**B - fHasShadow (1 bit):** A bit that specifies whether one or more **data markers** in the chart group has shadows.

**reserved (14 bits):** MUST be set to zero, and MUST be ignored.

**unused (2 bytes):** Undefined and MUST be ignored.

### 2.4.82 RadarArea

This record specifies that the chart group is a filled radar chart group, and specifies the chart group attributes.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
A	B	reserved														unused															

**A - fRdrAxLab (1 bit):** A bit that specifies whether **category** labels are displayed.

**B - fHasShadow (1 bit):** A bit that specifies whether the data points in the chart group has shadows.

**reserved (14 bits):** MUST be set to zero, and MUST be ignored.

**unused (2 bytes):** Undefined and MUST be ignored.

### 2.4.83 Scatter

This record specifies that the chart group is a scatter or a bubble chart group, and specifies the chart group attributes.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
pcBubbleSizeRatio																wBubbleSize															
A	B	C	reserved																												

**pcBubbleSizeRatio (2 bytes):** An unsigned integer that specifies the size of the data points in a bubble chart group as a percentage of their default size. If this field is set to 0x0064, all data points appear in their default size, as determined by the application. The value of **pcBubbleSizeRatio** MUST be greater than or equal to zero and less than or equal to 0x012C. MUST be ignored if the **fBubbles** field is set to zero.

**wBubbleSize (2 bytes):** An unsigned integer that specifies how the default size of the data points in a bubble chart group represents the values. MUST be ignored if the **fBubbles** field is set to zero. MUST be a value from the following table.

Value	Meaning
0x0001	The area of the data point represents the value.
0x0002	The width of the data point represents the value.

**A - fBubbles (1 bit):** A bit that specifies whether this chart group is a scatter or a bubble chart group. MUST be a value from the following table.

Value	Meaning
Zero	Scatter chart group
1	Bubble chart group

**B - fShowNegBubbles (1 bit):** A bit that specifies whether data points with negative values in the chart group are shown on the chart. MUST be ignored if the **fBubbles** field is set to zero.

**C - fHasShadow (1 bit):** A bit that specifies whether one or more **data markers** in a scatter chart group or data points in a bubble chart group has shadows.

**reserved (13 bits):** MUST be set to zero, and MUST be ignored.

#### 2.4.84 Scl

This record specifies the **zoom level** of the current view in the window used to display the **chart window** as a fraction calculated by the following formula:

$$\text{Fraction} = \text{nscl} / \text{dscl}$$

The fraction MUST be greater than or equal to 1/10 and less than or equal to 4/1.

This record MUST exist if the zoom of the current view is not set to 1.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
nscl																dscl															

**nscl (2 bytes):** A signed integer that specifies the numerator of the fraction. The value MUST be greater than or equal to 1.

**dscl (2 bytes):** A signed integer that specifies the denominator of the fraction. The value MUST be greater than or equal to 1.

### 2.4.85 Selection

This record specifies the **selection** in the data sheet window.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
pnn										rwAct										colAct											
...										reserved										unused											
...										rwFirst										rwLast											
...										colFirst										colLast											
...																															

**pnn (1 byte):** An unsigned integer that MUST be set to 0x03.

**rwAct (2 bytes):** A **Graph\_Rw** structure (section [2.5.15](#)) that specifies the row number of the **active cell**. MUST be greater than or equal to **rwFirst**. MUST be less than or equal to **rwLast**.

**colAct (2 bytes):** A **Graph\_Col** structure (section [2.5.14](#)) that specifies the column number of the active cell. MUST be greater than or equal to **colFirst**. MUST be less than or equal to **colLast**.

**reserved (2 bytes):** MUST be set to zero, and MUST be ignored.

**unused (2 bytes):** Undefined and MUST be ignored.

**rwFirst (2 bytes):** A **Graph\_Rw** structure (section [2.5.15](#)) that specifies the top row of the **active selection**.

**rwLast (2 bytes):** A **Graph\_Rw** structure (section [2.5.15](#)) that specifies bottom row of the active selection.

**colFirst (2 bytes):** A **Graph\_Col** structure (section [2.5.14](#)) that specifies the left-most column of the active selection.

**colLast (2 bytes):** A **Graph\_Col** structure (section [2.5.14](#)) that specifies the right-most column of the active selection.

### 2.4.86 SerAuxErrBar

This record specifies properties of an error bar.



0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31								
sertm										ebsrc										fTeeTop										reserved									
numValue																																							
...																																							
unused																																							

**sertm (1 byte):** An unsigned integer that specifies the direction of the error bars. MUST be a value from the following table.

Value	Meaning
0x01	Error bars are horizontal in the plus direction.
0x02	Error bars are horizontal in the minus direction.
0x03	Error bars are vertical in the plus direction.
0x04	Error bars are vertical in the minus direction.

**ebsrc (1 byte):** An unsigned integer that specifies the error amount type of the error bars. MUST be a value from the following table.

Value	Meaning
0x01	Percentage
0x02	Fixed value
0x03	Standard deviation
0x05	Standard error

**fTeeTop (1 byte):** A Boolean that specifies whether the error bars are T-shaped.

**reserved (1 byte):** MUST be ignored, and MUST be set to 0x01.

**numValue (8 bytes):** An **Xnum** structure (section [2.5.33](#)) that specifies the fixed value, percentage, or number of standard deviations for the error bars. If **ebsrc** is set to 0x05, this field MUST be ignored.

**unused (2 bytes):** Undefined and MUST be ignored.

#### 2.4.87 SerAuxTrend

This record specifies a trendline.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
regt										ordUser										numIntercept											
...																															

...	fEquation	fRSquared
numForecast		
...		
numBackcast		
...		

**regt (1 byte):** An unsigned integer that specifies the type of trendline. MUST be a value from the following table:

Value	Meaning
0x00	Polynomial
0x01	Exponential
0x02	Logarithmic
0x03	Power
0x04	Moving average

**ordUser (1 byte):** An unsigned integer that specifies the polynomial order or moving average period. MUST be greater than or equal to 0x02 and less than or equal to 0x06 if **regt** is set to 0x00. MUST be greater than or equal to 0x02 and less than or equal to the value of the **cValx** field of the **Series** record specified by the preceding **SerParent** record minus 1 if **regt** is set to 0x04. MUST be ignored for other values of **regt**.

**numIntercept (8 bytes):** A **ChartNumNillable** structure (section 2.5.3) that specifies where the trendline intersects the value axis or vertical axis on bubble and scatter chart groups. If no intercept is specified, this **ChartNumNillable** structure MUST specify a **NilChartNum** structure (section 2.5.21), and the value of the **type** field in the **NilChartNum** structure MUST be set to 0x0100.

**fEquation (1 byte):** A Boolean that specifies whether the trendline equation is displayed in the trendline label. MUST be ignored if **regt** is set to 0x04. MUST be ignored if the chart sheet substream contains an attached label with an **ObjectLink** record that contains both a **wLinkObj** field set to 0x0004 and a **wLinkVar1** field set to the zero-based index into a **Series** record in the collection of **Series** records in the current chart sheet substream that represents this trendline, and the attached label contains a **SeriesText** record.

**fRSquared (1 byte):** A Boolean that specifies whether the R-squared value is displayed in the trendline label. MUST be ignored if **regt** is set to 0x04. MUST be ignored if the chart sheet substream contains an attached label with an **ObjectLink** record that contains both a **wLinkObj** field set to 0x0004 and a **wLinkVar1** field set to the zero-based index into a **Series** record in the collection of **Series** records in the current chart sheet substream that represents this trendline, and the attached label contains a **SeriesText** record.

**numForecast (8 bytes):** An **Xnum** structure (section 2.5.33) that specifies the number of periods to **forecast** forward.

**numBackcast (8 bytes):** An **Xnum** structure (section 2.5.33) that specifies the number of periods to forecast backward.

## 2.4.88 SerFmt

This record specifies properties of the associated data points, **data markers**, or lines of the series. The associated data points, data markers, or lines of the series are specified by the preceding **DataFormat** record. If this record is not present in the sequence of records that conforms to the SS rule, all fields have default values. Otherwise, all fields **MUST** contain a value.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
A	B	C	reserved																												

**A - fSmoothedLine (1 bit):** A bit that specifies whether the lines of the series are displayed with a smooth line effect on a scatter, radar, and line chart group. The default value for this field is 0.

**B - f3DBubbles (1 bit):** A bit that specifies whether the data points of a bubble chart group are displayed with a 3-D effect. **MUST** be ignored for all other chart groups. The default value for this field is 0.

**C - fArShadow (1 bit):** A bit that specifies whether the data markers are displayed with a shadow on bubble, scatter, radar, stock, and line chart groups. The default value for this field is 0.

**reserved (13 bits):** **MUST** be set to zero, and **MUST** be ignored.

## 2.4.89 Series

This record specifies properties of the data for a series, a trendline, or error bars, and specifies the beginning of a collection of records as defined by the chart sheet substream **ABNF**. The collection of records specifies a series, a trendline, or error bars.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
sdtX																sdtY															
cValx																cValy															
sdtBSize																cValBSize															

**sdtX (2 bytes):** An unsigned integer that specifies the type of data in **categories**, or horizontal values on bubble and scatter chart groups, in the series. **MUST** be a value from the following table.

Value	Meaning
0x0001	The series contains categories, or horizontal values on bubble and scatter chart groups, with numeric information.
0x0003	The series contains categories, or horizontal values on bubble and scatter chart groups, with text information.

**sdtY (2 bytes):** An unsigned integer that specifies that the values, or vertical values on bubble and scatter chart groups, in the series contain numeric information. **MUST** be set to 0x0001, and **MUST** be ignored.

**cValx (2 bytes):** An unsigned integer that specifies the count of categories, or horizontal values on bubble and scatter chart groups, in the series. This value MUST be less than or equal to 0x0F9F.

**cValy (2 bytes):** An unsigned integer that specifies the count of values, or vertical values on bubble and scatter chart groups, in the series. This value MUST be less than or equal to 0x0F9F.

**sdtBSize (2 bytes):** An unsigned integer that specifies that the **bubble size** values in the series contain numeric information. This value MUST be set to 0x0001, and MUST be ignored.

**cValBSize (2 bytes):** An unsigned integer that specifies the count of bubble size values in the series. This value MUST be less than or equal to 0x0F9F.

#### 2.4.90 SeriesList

This record specifies the series for the chart.

0	1	2	3	4	5	6	7	8	9	1	0	1	2	3	4	5	6	7	8	9	2	0	1	2	3	4	5	6	7	8	9	3	0	1
cser											rgiser (variable)																							
...																																		

**cser (2 bytes):** An unsigned integer that specifies number of items in the **rgiser** field.

**rgiser (variable):** An array of 2-byte unsigned integers, each of which specifies a one-based index of a **Series** record in the collection of **Series** records in the current chart sheet substream. Each referenced **Series** specifies a series for the chart.

#### 2.4.91 SeriesText

This record specifies the text for a series, trendline name, trendline label, axis title, or chart title.

0	1	2	3	4	5	6	7	8	9	1	0	1	2	3	4	5	6	7	8	9	2	0	1	2	3	4	5	6	7	8	9	3	0	1
reserved											stText (variable)																							
...																																		

**reserved (2 bytes):** MUST be set to zero, and MUST be ignored.

**stText (variable):** A **ShortXLUnicodeString** structure (section [2.5.27](#)) that specifies the text string.

#### 2.4.92 SerParent

This record specifies the series to which the current trendline or error bar corresponds.

0	1	2	3	4	5	6	7	8	9	1	0	1	2	3	4	5	6	7	8	9	2	0	1	2	3	4	5	6	7	8	9	3	0	1
series																																		

**series (2 bytes):** An unsigned integer that specifies the one-based index of a **Series** record in the collection of **Series** records in the current chart sheet substream. The referenced **Series** record

specifies the series associated with the current trendline or error bar. The value MUST be greater than or equal to 0x0001 and less than or equal to 0x00FE.

#### 2.4.93 SerToCrt

This record specifies the chart group for the current series.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
id																															

**id (2 bytes):** An unsigned integer that specifies the zero-based index of a **ChartFormat** record in the collection of **ChartFormat** records in the current chart sheet substream. The referenced **ChartFormat** record specifies the chart group that contains the current series.

#### 2.4.94 ShtProps

This record specifies properties of a chart as defined by the chart sheet substream **ABNF**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
A	B	C	D	E	reserved2											mdBlank					reserved3										

**A - fManSerAlloc (1 bit):** A bit that specifies whether series are automatically allocated for the chart.

**B - unused1 (1 bit):** Undefined and MUST be ignored.

**C - reserved1 (1 bit):** MUST be set to zero, and MUST be ignored.

**D - fManPlotArea (1 bit):** If **fAlwaysAutoPlotArea** is set to 1, this field MUST be set to 1. If **fAlwaysAutoPlotArea** is set to zero, this field MUST be ignored.

**E - fAlwaysAutoPlotArea (1 bit):** A bit that specifies whether the default **plot area** dimension is used. MUST be a value from the following table.

Value	Meaning
0	Use the default plot area dimension regardless of the <b>Pos</b> record information.
1	Use the plot area dimension of the <b>Pos</b> record, and <b>fManPlotArea</b> MUST be set to 1.

**reserved2 (11 bits):** MUST be set to zero, and MUST be ignored.

**mdBlank (1 byte):** An unsigned integer that specifies how the empty cells are plotted. MUST be a value from the following table.

Value	Meaning
0x00	Empty cells are not plotted.
0x01	Empty cells are plotted as zero.
0x02	Empty cells are plotted as interpolated.

**reserved3 (1 byte):** MUST be set to zero, and MUST be ignored.

## 2.4.95 StartBlock

This record specifies the beginning of a collection of records. Future records contained in this collection specify saved features to enable applications that do not support the feature to preserve the information. This record MUST have a matching **EndBlock** record. **StartBlock** and **EndBlock** pairs can be nested. Up to 100 levels of blocks can be nested.

Prior to writing a chart-specific future record, which is a record with a record number that is greater than or equal to 2048 and less than or equal to 2303 according to record enumeration, a **StartBlock** record MUST be written according to the following rules:

1. A **StartBlock** record MUST not be written if this record is preceded by a **StartObject** record, but not preceded by the matching **EndObject** record. That is, **StartBlock** and **EndBlock** pairs MUST not belong to any collection defined by **StartObject** and **EndObject**.
2. If no **StartBlock** record with **iObjectKind** set to 0x000D without a matching **EndBlock** record exists, a corresponding **StartBlock** record with **iObjectKind** set to 0x000D MUST be written.
3. If the chart-specific future record is in an axis group and no **StartBlock** record with **iObjectKind** set to 0x0000 without a matching **EndBlock** record exists, a corresponding **StartBlock** record with **iObjectKind** set to 0x0000 and **iObjectInstance1** equal to the **iax** field of the **AxisParent** record of the axis group MUST be written. If a **StartBlock** record is written due to rule number 2, this **StartBlock** record MUST be written immediately after that record.
4. If the chart-specific future record is in a chart group and no **StartBlock** record with **iObjectKind** set to 0x0005 without a matching **EndBlock** record exists, a corresponding **StartBlock** record with **iObjectKind** set to 0x0005 and **iObjectInstance1** equal to the **iax** field of the **AxisParent** record of the axis group MUST be written. If any **StartBlock** records are written due to rules number 2 or 3, this **StartBlock** record MUST be written immediately after those records.
5. If the chart-specific future record is in an axis and no **StartBlock** record with **iObjectKind** set to 0x0004 without a matching **EndBlock** record exists, the following apply:
  1. If the chart-specific future record exists in the sequence of records that conforms to the **IVAXIS** rule, a corresponding **StartBlock** record with **iObjectKind** set to 0x0004 and **iObjectInstance1** set to 0x0000 MUST be written. If any **StartBlock** records are written due to rules number 2, 3, or 4, this **StartBlock** record MUST be written immediately after those records.
  2. If the chart-specific future record exists in the sequence of records that conforms to the **SERIESAXIS** rule, a corresponding **StartBlock** record with **iObjectKind** set to 0x0004 and **iObjectInstance1** set to 0x0002 MUST be written. If any **StartBlock** records are written due to rules number 2, 3, or 4, this **StartBlock** record MUST be written immediately after those records.
  3. If the chart-specific future record exists in the sequence of records that conforms to the **DVAXIS** rule and **wType** of the axis in the **DVAXIS** rule is set to zero, a corresponding **StartBlock** record with **iObjectKind** set to 0x0004 and **iObjectInstance1** set to 0x0001 MUST be written. If any **StartBlock** records are written due to rules number 2, 3, or 4, this **StartBlock** record MUST be written immediately after those records.
  4. If the chart-specific future record exists in the sequence of records that conforms to the **DVAXIS** rule and **wType** of the axis in the **DVAXIS** rule is set to 1, a corresponding **StartBlock** record with **iObjectKind** set to 0x0004 and **iObjectInstance1** set to 0x0003 MUST be written. If any **StartBlock** records are written due to rules number 2, 3, or 4, this **StartBlock** record MUST be written immediately after those records.

6. If the chart-specific future record is in an **AttachedLabel** and no **StartBlock** record with **iObjectKind** set to 0x0002 without a matching **EndBlock** record exists, the following apply:
  1. If the chart-specific future record exists in the sequence of records that conforms to the DFTTEXT rule of a chart group, a corresponding **StartBlock** record with **iObjectKind** set to 0x0002, **iObjectContext** set to 0x0002 and **iObjectInstance1** equal to the **id** field of the **DefaultText** record of the DFTTEXT rule **MUST** be written. If any **StartBlock** records are written due to rules number 2, 3, 4, or 5, this **StartBlock** record **MUST** be written immediately after those records.
  2. If a corresponding **StartBlock** record with **iObjectKind** set to 0x0002, **iObjectContext** set to 0x0005, **iObjectInstance1** equal to the **wLinkVar1** field of the **ObjectLink** record of the **AttachedLabel**, and **iObjectInstance2** equal to the **wLinkVar2** field of the **ObjectLink** record of the **AttachedLabel** **MUST** be written. If any **StartBlock** records are written due to rules number 2, 3, 4, or 5, this **StartBlock** record **MUST** be written immediately after those records.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1
frtHeaderOld																															
iObjectKind																iObjectContext															
iObjectInstance1																iObjectInstance2															

**frtHeaderOld (4 bytes):** An **FrtHeaderOld** structure (section [2.5.10](#)). The **frtHeaderOld.rt** field **MUST** be set to 0x0852.

**iObjectKind (2 bytes):** An unsigned integer that specifies the type of object that is encompassed by the block. **MUST** be a value from the following table.

Value	Object type
0x0000	Axis group
0x0002	Attached label
0x0004	Axis
0x0005	Chart group
0x000D	<b>Sheet</b>

**iObjectContext (2 bytes):** An unsigned integer that specifies the context of the object. This value further specifies the object specified in **iObjectKind**. **MUST** be a value from the following table.

iObjectKind	iObject context	iObject instance1	iObject instance2	Description of future record type contents
0x0000 (Axis group)	0x0000	0x0000	0x0000	Primary axis group of the current chart.
0x0000 (Axis group)	0x0000	0x0001	0x0000	Secondary axis group of the current chart.
0x0002	0x0002	0x0000	0x0000	Default data labels on the chart that are not displayed as a percentage of the sum

<b>iObjectKind</b>	<b>iObject context</b>	<b>iObject instance1</b>	<b>iObject instance2</b>	<b>Description of future record type contents</b>
<b>(AttachedLabel)</b>				of all data points and do not contain values.
0x0002 (AttachedLabel)	0x0002	0x0001	0x0000	Default data labels that contain values or percentage values.
0x0002 <b>(AttachedLabel)</b>	0x0005	Data Point Index	Series Index	Data labels for data points in visible series, identified by the zero-based index of the <b>Series</b> record of the current chart and the zero-based index of the <b>AttachedLabel</b> record of the current series.
0x0002 <b>(AttachedLabel)</b>	0x0005	0xFFFF	Series Index	Default data labels for a series, identified by the zero-based index of the <b>Series</b> record of the current chart.
0x0004 (Axis)	0x0000	0x0000	0x0000	<b>Category</b> axis or fill effect of the walls of the current chart.
0x0004 (Axis)	0x0000	0x0001	0x0000	Value axis or fill effect of the walls of the current chart.
0x0004 (Axis)	0x0000	0x0002	0x0000	Series axis.
0x0004 (Axis)	0x0000	0x0003	0x0000	Horizontal value axis for scatter chart group.
0x0005 (Chart group)	0x0000	0x0000	0x0000	Chart group of the primary axis group.
0x0005 (Chart group)	0x0000	0x0001	0x0000	Chart group of the secondary axis group.
0x000D (Sheet (1))	0x0000	0x0000	0x0000	Current sheet (1).

**iObjectInstance1 (2 bytes):** An unsigned integer that specifies additional information about the context of the object, along with **iObjectContext**, **iObjectInstance2**, and **iObjectKind**. This field MUST be one of those specified in the previous table in the **iObjectContext** column.

**iObjectInstance2 (2 bytes):** An unsigned integer that specifies more information about the object context, along with **iObjectContext**, **iObjectInstance1**, and **iObjectKind**. This field MUST be one of those specified in the previous table in the **iObjectContext** column.

## 2.4.96 StartObject

This record specifies the beginning of a collection of future record type records as defined by the chart sheet substream **ABNF**. The collection of records specifies a feature saved as a future record type so that an application not supporting the feature can preserve it. This record MUST have a matching **EndObject** record. **StartObject** and **EndObject** pairs can be nested. Up to 100 levels of blocks can be nested.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
firtHeaderOld																															



iObjectKind	iObjectContext
iObjectInstance1	iObjectInstance2

**frtHeaderOld (4 bytes):** An **FrtHeaderOld** structure (section [2.5.10](#)). The **frtHeaderOld.rt** field MUST be set to 0x0854.

**iObjectKind (2 bytes):** An unsigned integer that specifies the type of object that is encompassed by the block. MUST be a value from the following table.

Value	Object type encompassed by the block	Description of future record type contents
0x0010	A sequence of records that conforms to the ATTACHEDLABEL rule is encompassed by the block.	<b>Display units</b> labels of the current axis.
0x0011	A *(Font [Fbi]) is encompassed by the block, as defined by the sequence of records that conforms to the FONTLIST rule.	<b>Font</b> cache for a specified application version. The block contains only fonts for records introduced in the specified application version.
0x0012	A <b>DefaultText</b> record followed by an ATTACHEDLABEL rule is encompassed by the block when in a sequence of records that conforms to the DFTTEXT rule. An ATTACHEDLABEL rule is encompassed by the block when in a sequence of records that conforms to the CHARTFORMATS rule.	An extended data label.

**iObjectContext (2 bytes):** An unsigned integer that specifies the object context. MUST be set to 0x0000.

**iObjectInstance1 (2 bytes):** An unsigned integer that specifies additional information about the context of the object, along with **iObjectContext**, **iObjectInstance2**, and **iObjectKind**. This field MUST set to 0x0000 if **iObjectKind** is set to 0x0010 or 0x0012. MUST be a value from the following table if **iObjectKind** is set to 0x0011.

iObjectInstance1	Application version
0x0008	Specifies the application version. <a href="#">&lt;59&gt;</a>
0x0009	Specifies the application version. <a href="#">&lt;60&gt;</a>
0x000A	Specifies the application version. <a href="#">&lt;61&gt;</a>
0x000B	Specifies the application version. <a href="#">&lt;62&gt;</a>
0x000C	Specifies the application version. <a href="#">&lt;63&gt;</a>
0x000E	Specifies the application version. <a href="#">&lt;64&gt;</a>

**iObjectInstance2 (2 bytes):** An unsigned integer that specifies more information about the object context, along with **iObjectContext**, **iObjectInstance1**, and **iObjectKind**. This field MUST be set to 0x0000.

## 2.4.97 Surf

This record specifies that the chart group is a surface chart group and specifies the chart group attributes.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	1	2	3	4	5	6	7	8	9	30	1
A	B	reserved																													

**A - fFillSurface (1 bit):** A bit that specifies whether the surface chart group is wireframe or has a fill. MUST be a value from the following table.

Value	Meaning
0	Surface chart group is wireframe.
1	Surface chart group has a fill.

**B - f3DPhongShade (1 bit):** A bit that specifies whether **3-D Phong shading** is displayed.

**reserved (14 bits):** MUST be set to zero, and MUST be ignored.

## 2.4.98 Text

This record specifies the properties of an attached label and specifies the beginning of a collection of records as defined by the chart sheet substream **ABNF**. This collection of records specifies an attached label.

The **fShowKey**, **fShowValue**, **fShowLabelAndPerc**, **fShowPercent**, **fShowBubbleSizes**, and **fShowLabel** fields MUST be set to zero and MUST be ignored if the current attached label does not contain an **ObjectLink** record that satisfies the following conditions:

- The **wLinkObj** field of the **ObjectLink** record is set to 0x0004.
- The **wLinkVar1** field of the **ObjectLink** record references a series.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	1	2	3	4	5	6	7	8	9	30	1
at										vat										wBkgMode											
rgbText																															
x																															
y																															
dx																															
dy																															
A	B	C	D	E	F	G	H	I	J	K	L	M	N	icvText																	
dlp				unused3												O	trot														

**at (1 byte):** An unsigned integer that specifies the **horizontal alignment** of the text. MUST be a value from the following table.

Value	Alignment
0x01	Left alignment if <b>iReadingOrder</b> specifies <b>left-to-right reading order</b> ; otherwise, right alignment.
0x02	Center alignment.
0x03	Right alignment if <b>iReadingOrder</b> specifies left-to-right reading order; otherwise, left-alignment.
0x04	Justify-alignment.
0x07	<b>Distributed alignment.</b>

**vat (1 byte):** An unsigned integer that specifies the **vertical alignment** of the text. MUST be a value from the following table.

Value	Alignment
0x01	Top alignment
0x02	Center alignment
0x03	Bottom alignment
0x04	Justify alignment
0x07	Distributed alignment

**wBkgMode (2 bytes):** An unsigned integer that specifies the display mode of the background of the text. MUST be a value from the following table.

Value	Background mode
0x0001	Transparent background
0x0002	Opaque background

**rgbText (4 bytes):** A **LongRGB** structure (section [2.5.20](#)) that specifies the color of the text.

**x (4 bytes):** A signed integer that specifies the horizontal position of the text, relative to the upper left of the **chart area**, in SPRC. This value MUST be ignored when this record is preceded by a **DefaultText** record or is followed by a **Pos** record; otherwise, MUST be greater than or equal to zero and less than or equal to 0x7FFF. SHOULD [<65>](#) be less than or equal to 0x0FA0.

**y (4 bytes):** A signed integer that specifies the vertical position of the text, relative to the upper left of the chart area, in SPRC. This value MUST be ignored when this record is preceded by a **DefaultText** record or is followed by a **Pos** record; otherwise, MUST be greater than or equal to zero and less than or equal to 0x7FFF. SHOULD [<66>](#) be less than or equal to 0x0FA0.

**dx (4 bytes):** A signed integer that specifies the horizontal size of the text, relative to the chart area, in SPRC. This value MUST be ignored when this record is followed by a **Pos** record; otherwise, MUST be greater than or equal to zero, and less than or equal to 0x7FFF. SHOULD [<67>](#) be less than or equal to 0x0FA0.

**dy (4 bytes):** A signed integer that specifies the vertical size of the text, relative to the chart area, in SPRC. This value MUST be ignored when this record is followed by a **Pos** record; otherwise, MUST be greater than or equal to zero and less than or equal to 0x7FFF. SHOULD <68> be less than or equal to 0x0FA0.

**A - fAutoColor (1 bit):** A bit that specifies whether the foreground text color is determined automatically.

**B - fShowKey (1 bit):** A bit that specifies whether the text is attached to a **legend key**.

**C - fShowValue (1 bit):** A bit that specifies whether the value, or the vertical value on bubble or scatter chart groups, is displayed in the data label.

If the current attached label contains a **DataLabExtContents** record and the **fPercent** field of the **DataLabExtContents** record is set to zero, this field MUST equal the **fValue** field of the **DataLabExtContents** record.

This field MUST be set to zero if the current attached label does not contain a **DataLabExtContents** record and one or more of the following conditions are satisfied:

- The **fShowLabelAndPerc** field is set to 1.
- The **fShowPercent** field is set to 1.

**D - unused1 (1 bit):** Undefined and MUST be ignored.

**E - fAutoText (1 bit):** A bit that specifies whether the text value of this text field is automatically generated and unchanged.

**F - fGenerated (1 bit):** A bit that specifies whether the properties of this text field are automatically generated and unchanged.

**G - fDeleted (1 bit):** A bit that specifies whether this data label has been deleted by the user.

**H - fAutoMode (1 bit):** A bit that specifies whether the **background color** is determined automatically.

**I - unused2 (3 bits):** Undefined and MUST be ignored.

**J - fShowLabelAndPerc (1 bit):** A bit that specifies whether the **category** name and the value, represented as a percentage of the sum of the values of the series with which the data label is associated, are displayed in the data label.

MUST be set to zero if the chart group type of the corresponding chart group, series, or data point is not a bar of pie, doughnut, pie, or pie of pie chart group.

This field MUST be set to 1 if the current attached label contains a **DataLabExtContents** record and both of the following conditions are satisfied:

- The **fCatName** and **fPercent** fields of the **DataLabExtContents** record are set to 1.
- The **fSerName**, **fValue**, and **fBubSizes** fields of the **DataLabExtContents** record are set to zero.

This field MUST be set to zero if the current attached label contains a **DataLabExtContents** record and one or more of the following conditions is satisfied:

- The **fCatName** or **fPercent** fields of the **DataLabExtContents** record are set to zero.
- The **fSerName**, **fValue**, or **fBubSizes** fields of the **DataLabExtContents** record are set to 1.

MUST be ignored if **fAutoText** is set to zero.

**K - fShowPercent (1 bit):** A bit that specifies whether the value, represented as a percentage of the sum of the values of the series with which the data label is associated, is displayed in the data label.

MUST be set to zero if the chart group type of the corresponding chart group, series, or data point is not a bar of pie, doughnut, pie, or pie of pie chart group.

If the current attached label contains a **DataLabExtContents** record, this field MUST equal the **fPercent** field of the **DataLabExtContents** record.

If the current attached label does not contain a **DataLabExtContents** record and **fShowLabelAndPerc** is set to 1, this field MUST be set to 1.

MUST be ignored if **fAutoText** is set to zero.

**L - fShowBubbleSizes (1 bit):** A bit that specifies whether the **bubble size** is displayed in the data label.

MUST be set to zero if the chart group type of the corresponding chart group, series, or data point is not a bubble chart group.

If the current attached label contains a **DataLabExtContents** record and the **fPercent**, **fValue**, and **fCatName** fields of the **DataLabExtContents** record is set to zero, this field MUST equal the **fBubSizes** field of the **DataLabExtContents** record.

If the current attached label contains a **DataLabExtContents** record and the **fPercent**, **fValue**, or **fCatName** fields of the **DataLabExtContents** record are set to 1, this field MUST be set to zero.

If the current attached label does not contain a **DataLabExtContents** record and **fShowPercent**, **fShowValue** or **fShowLabel** is set to 1, this field MUST be set to zero.

MUST be ignored if **fAutoText** is set to zero.

**M - fShowLabel (1 bit):** A bit that specifies whether the category, or the horizontal value on bubble or scatter chart groups, is displayed in the data label for a non-area chart group, or the series name is displayed in the data label for an area chart group.

This field MUST equal the **fCatName** field of the **DataLabExtContents** record if the current attached label contains a **DataLabExtContents** record, the chart group is non-area, and both of the following conditions are satisfied:

- The **fValue** field of the **DataLabExtContents** record is set to zero.
- The **fShowLabelAndPerc** field is set to 1, or the **fPercent** field is set to zero.

This field MUST equal the **fSerName** field of the **DataLabExtContents** record if the current attached label contains a **DataLabExtContents** record, the chart group is area or filled radar, and the following condition is satisfied:

- The **fValue** field of the **DataLabExtContents** record is set to zero.

If the current attached label contains a **DataLabExtContents** record and the **fValue** field of the **DataLabExtContents** record is set to 1, this field MUST be set to zero.

This field MUST be set to zero if the current attached label does not contain a **DataLabExtContents** record and one of the following conditions is satisfied:

- The **fShowValue** field is set to 1.
- The **fShowLabelAndPerc** field is set to zero, and the **fShowPercent** field is set to 1.

MUST be ignored if **fAutoText** is set to zero.

**N - reserved (1 bit):** MUST be set to zero, and MUST be ignored.

**icvText (2 bytes):** An **Icv** structure (section 2.5.16) that specifies the color of the text.

**dlp (4 bits):** An unsigned integer that specifies the data label positioning of the text, relative to the **graph object** item to which the text is attached. For all data label text fields, MUST be a value from the following table.

Data label position	Value	Valid in chart type
Auto	0x0	Pie chart group.
Right	0x0	Line, bubble, or scatter chart group.
Outside	0x0	Bar or column chart group with <b>fStacked</b> set to zero.
Center	0x0	Bar or column chart group with <b>fStacked</b> set to 1.
Outside End	0x1	Bar, column, or pie chart group.
Inside End	0x2	Bar, column, or pie chart group.
Center	0x3	Bar, column, line, bubble, scatter or pie chart group.
Inside Base	0x4	Bar or column chart group.
Above	0x5	Line, bubble, or scatter chart group.
Below	0x6	Line, bubble, or scatter chart group.
Left	0x7	Line, bubble, or scatter chart group.
Right	0x8	Line, bubble, or scatter chart group.
Auto	0x9	Pie chart group.
Moved by user	0xA	All.

For all non-data label text fields, MUST be set to 0x0.

**unused3 (10 bits):** Undefined, and MUST be ignored.

**O - iReadingOrder (2 bits):** An unsigned integer that specifies the reading order of the text. MUST be a value from the following table.

Value	Reading order
0x0	The reading order is equal to the <b>iReadingOrder</b> value of the <b>Text</b> record immediately following the closest preceding <b>Chart</b> , <b>DataFormat</b> , <b>Legend</b> , <b>Series</b> , or <b>YMult</b> record, where <b>iReadingOrder</b> is not set to 0x0. If no such preceding record exists, the <b>DefaultText</b> settings of the chart are used. If the <b>DefaultText</b> settings also specify 0x0, the reading order is determined by the application.
0x1	Left-to-right.
0x2	<b>Right-to-left.</b>

**trot (2 bytes):** An unsigned integer that specifies the text rotation. MUST be a value from the following table.

Value	Angle description
0 to 90	Text rotated zero to 90 degrees counter-clockwise.
91 to 180	Text rotated 1 to 90 degrees clockwise. (Angle is <b>trot</b> minus 90.)
255	Text top-to-bottom with letters upright.

#### 2.4.99 Tick

This record specifies the attributes of the axis labels, **major tick marks**, and **minor tick marks** associated with an axis.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31		
tkMajor										tkMinor										tlt						wBkgMode							
rgb																																	
reserved1																																	
reserved2																																	
reserved3																																	
reserved4																																	
A	B	rot			C	unused						D	icv																				
trot																																	

**tkMajor (1 byte):** An unsigned integer that specifies the location of major tick marks. MUST be a value from the following table.

Value	Tick mark location
0x00	None. No major tick marks are drawn on the axis.
0x01	Inside. Major tick marks are drawn toward the <b>plot area</b> .
0x02	Outside. Major tick marks are drawn away from the plot area.
0x03	Crossing. Major tick marks are drawn evenly on both sides of the axis.

**tkMinor (1 byte):** An unsigned integer that specifies the location of minor tick marks. MUST be a value from the following table.

Value	Tick mark location
0x00	None. No minor tick marks are present on the axis.
0x01	Inside. Minor tick marks are drawn toward the plot area.
0x02	Outside. Minor tick marks are drawn away from the plot area.
0x03	Crossing. Minor tick marks are drawn evenly on both sides of the axis.

**tlf (1 byte):** An unsigned integer that specifies the location of axis labels. MUST be a value from the following table.

Value	Tick mark label location
0x00	The <b>reading order</b> is equal to the <b>iReadingOrder</b> value of the closest preceding <b>Chart</b> , <b>DataFormat</b> , <b>Legend</b> , <b>Series</b> , or <b>YMult</b> record, where <b>iReadingOrder</b> is not set to 0x0. If no such preceding record exists, the <b>DefaultText</b> settings of the chart are used. If the <b>DefaultText</b> settings also specify 0x0, the system settings are used.
0x01	Low. Axis labels are drawn to the left of the plot area for a vertical axis, or below the plot area for a horizontal axis for all chart group types, except radar. Axis labels for radar chart group types are drawn as if the value were 0x0003.
0x02	High. Axis labels are drawn to the right of the plot area for a vertical axis, or above the plot area for a horizontal axis for all chart group types, except radar. Axis labels for radar chart group types are drawn as if the value were 0x0003.
0x03	Next to axis. Axis labels are drawn next to the axis.

**wBkgMode (1 byte):** An unsigned integer that specifies the display mode of the background of the text of the axis labels. MUST be ignored if the value of **fAutoCo** is set to 1. MUST be a value from the following table.

Value	Background mode
0x01	Transparent background.
0x02	Opaque background. The background color matches the <b>rgbBack</b> field in the associated <b>AreaFormat</b> record, as specified by the sequence of records that conforms to the AXS rule.

**rgb (4 bytes):** A **LongRGB** structure (section [2.5.20](#)) that specifies the color of the text for the axis labels. MUST be ignored if **fAutoCo** is set to 1.

**reserved1 (4 bytes):** MUST be set to zero, and MUST be ignored.

**reserved2 (4 bytes):** MUST be set to zero, and MUST be ignored.

**reserved3 (4 bytes):** MUST be set to zero, and MUST be ignored.

**reserved4 (4 bytes):** MUST be set to zero, and MUST be ignored.



**A - fAutoCo (1 bit):** A bit that specifies whether the foreground text color of the axis labels is determined automatically. MUST be a value from the following table.

Value	Meaning
Zero	The text uses the color specified by <b>rgb</b> .
1	The text uses an automatically selected foreground color, based on the display settings of the computer.

**B - fAutoMode (1 bit):** A bit that specifies whether the background color of the axis label is determined automatically. MUST be a value from the following table.

Value	Meaning
Zero	The background color is specified by <b>wBkgMode</b> .
1	The background color is set according to the <b>DefaultText</b> settings of the chart.

**rot (3 bits):** An unsigned integer that specifies text rotation of the axis labels. MUST be a value from the following table.

Value	Text rotation
0x000	Text is drawn with the rotation specified by the value of <b>trot</b> .
0x001	Text is drawn stacked, top-to-bottom, with the letters upright.
0x002	Text is drawn rotated at 90 degrees counter-clockwise.
0x003	Text is drawn rotated at 90 degrees clockwise.

**C - fAutoRot (1 bit):** A bit that specifies whether the text rotation of the axis label is determined automatically. MUST be a value from the following table.

Value	Meaning
Zero	The axis labels do not rotate automatically, but follow the rotation specified by <b>rot</b> and <b>trot</b> .
1	The axis labels rotate as the location of the axis changes, and the value of the <b>rot</b> and <b>trot</b> fields MUST be ignored.

**unused (8 bits):** Undefined, and MUST be ignored.

**D - iReadingOrder (2 bits):** An unsigned integer that specifies the reading order of the axis labels. MUST be a value from the following table.

Value	Meaning
0x00	The reading order is equal to the <b>iReadingOrder</b> value of the <b>Text</b> record immediately following the closest preceding <b>Chart</b> , <b>DataFormat</b> , <b>Legend</b> , <b>Series</b> , or <b>YMult</b> record, where <b>iReadingOrder</b> is not set to 0x0. If no such preceding record exists, the <b>DefaultText</b> settings of the chart are used. If the <b>DefaultText</b> settings also specify 0x0, the reading order is determined by the application.
0x01	<b>Left-to-right.</b>
0x02	<b>Right-to-left.</b>

**icv (2 bytes):** An unsigned integer that specifies the color of the text in the **color palette**. The value MUST [<69>](#) be an **IcvChart** value (section [2.5.17](#)). The color MUST be the same as **rgb**.

**trot (2 bytes):** An unsigned integer that specifies the text rotation of the axis label. MUST be a value from the following table.

Value	Meaning
0x0 to 0x005A	Text rotated zero to 90 degrees counterclockwise.
0x005B to 0x00B4	Text rotated 1 to 90 degrees clockwise. (Angle is <b>trot</b> minus 90.)
0x00FF	Text top-to-bottom with letters upright.

#### 2.4.100 TxO

This record specifies the text in a text box or a form control. This record can be followed by a collection of **Continue** records that specifies additional feature data to complete this record.

If the field **cchText** is not set to zero, this record does not fully specify the text. The rest of the data that MUST be specified is the text string and the **formatting runs** information. That data is specified in two sets of **Continue** records:

**Text string specification:** The first set of **Continue** records specifies the text string. Each of these **Continue** records contains an **XLUnicodeStringNoCch** structure (section [2.5.32](#)) that specifies part of the string. The total number of characters in all **XLUnicodeStringNoCch** MUST be **cchText**.

**Formatting run specification:** The second set of **Continue** records specifies formatting runs. These **Continue** records contain a **TxORuns** structure (section [2.5.29](#)). If the size of the **TxORuns** structure is greater than 8224 bytes, it is split across multiple **Continue** records.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
A	B	C	D	E	F	G	H	rot																							
reserved4															reserved5																
...															cchText																

cbRuns	ifntEmpty
cbFmla	

**A - reserved1 (1 bit):** MUST be set to zero, and MUST be ignored.

**B - hAlignment (3 bits):** An unsigned integer that specifies the **horizontal alignment**. The value MUST be one of the values in the following table.

Value	Meaning
1	Specifies left alignment.
2<70>	Specifies centered alignment.
3	Specifies right alignment.
4	Specifies justified alignment.
7	Specifies <b>justify distributed</b> alignment.

**C - vAlignment (3 bits):** An unsigned integer that specifies the **vertical alignment**. The value MUST be one of the values in the following table

Value	Meaning
1	Specifies top alignment.
2	Specifies middle alignment.
3	Specifies bottom alignment.
4	Specifies justified alignment<71>.
7	Specifies justify distributed alignment.

**D - reserved2 (2 bits):** MUST be set to zero, and MUST be ignored.

**E - unused1 (1 bit):** Undefined and MUST be ignored.

**F - reserved3 (4 bits):** MUST be set to zero, and MUST be ignored.

**G - fJustLast (1 bit):** A bit that specifies whether justify alignment or justify distributed alignment is used on the last line of the text in specific versions of the application<72>.

**H - unused2 (1 bit):** Undefined and MUST be ignored.

**rot (2 bytes):** An unsigned integer that specifies the orientation of the text in the object boundary. MUST be a value from the following table.

Value	Meaning
Zero	Specifies no rotation.
1	Specifies stacked or vertical orientation.
2	Specifies 90-degree counter-clockwise rotation.
3	Specifies 90-degree clockwise rotation.

**reserved4 (2 bytes):** MUST be set to zero, and MUST be ignored.

**reserved5 (4 bytes):** MUST be set to zero, and MUST be ignored.

**cchText (2 bytes):** An unsigned integer that specifies the number of characters in the text string contained in the **Continue** records immediately following this record. MUST be less than or equal to 0x00FF.

**cbRuns (2 bytes):** An unsigned integer that specifies the number of bytes of formatting run information in the **TxORuns** structure (section 2.5.29) contained in the **Continue** records following this record. If **cchText** is set to zero, this field MUST be set to zero. Otherwise, the value MUST be greater than or equal to 0x0010 and MUST be a multiple of 8.

**ifntEmpty (2 bytes):** A **FontIndex** structure (section 2.5.5) that specifies the **font** when **cchText** is set to zero.

**cbFmla (2 bytes):** MUST be set to zero, and MUST be ignored.

#### 2.4.101 Units

This record MUST be set to zero, and MUST be ignored.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																
reserved																																															

**reserved (2 bytes):** MUST be set to zero, and MUST be ignored.

#### 2.4.102 ValueRange

This record specifies the properties of a value axis.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31																	
numMin																																																
...																																																
numMax																																																
...																																																

numMajor								
...								
numMinor								
...								
numCross								
...								
A	B	C	D	E	F	G	H	unused

**numMin (8 bytes):** An **Xnum** structure (section [2.5.33](#)) that specifies the minimum value of the value axis. MUST be less than **numMax**. If the value of **fAutoMin** is set to 1, this field MUST be ignored.

**numMax (8 bytes):** An **Xnum** structure (section 2.5.33) that specifies the maximum value of the value axis. MUST be greater than **numMin**. If the value of **fAutoMax** is set to 1, this field MUST be ignored.

**numMajor (8 bytes):** An **Xnum** structure (section 2.5.33) that specifies the interval at which **major tick marks** and **major gridlines** are displayed. MUST be greater than or equal to **numMinor**. If the value of **fAutoMajor** is set to 1, this field MUST be ignored.

**numMinor (8 bytes):** An **Xnum** structure (section 2.5.33) that specifies the interval at which **minor tick marks** and **minor gridlines** are displayed. MUST be greater than or equal to zero. If the value of **fAutoMinor** is set to 1, this field MUST be ignored.

**numCross (8 bytes):** An **Xnum** structure (section 2.5.33) that specifies at which value the other axes in the axis group cross this value axis. If the value of **fAutoCross** is set to 1, this field MUST be ignored.

**A - fAutoMin (1 bit):** A bit that specifies whether **numMin** is calculated automatically. MUST be one of the values in the following table.

Value	Meaning
0	The value specified by <b>numMin</b> is used as the minimum value of the value axis.
1	The <b>numMin</b> value is calculated so that the data point with the minimum value can be displayed in the <b>plot area</b> .

**B - fAutoMax (1 bit):** A bit that specifies whether **numMax** is calculated automatically. MUST be one of the values in the following table.

Value	Meaning
0	The value specified by <b>numMax</b> is used as the maximum value of the value axis.
1	The <b>numMax</b> value is calculated so that the data point with the maximum value can be displayed in the plot area.

**C - fAutoMajor (1 bit):** A bit that specifies whether **numMajor** is calculated automatically. MUST be one of the values in the following table.

Value	Meaning
0	The value specified by <b>numMajor</b> is used as the interval at which major tick marks and major gridlines are displayed.
1	The <b>numMajor</b> value is calculated automatically.

**D - fAutoMinor (1 bit):** A bit that specifies whether **numMinor** is calculated automatically. MUST be one of the values in the following table.

Value	Meaning
0	The value specified by <b>numMinor</b> is used as the interval at which minor tick marks and minor gridlines are displayed.
1	The <b>numMinor</b> value is calculated automatically.

**E - fAutoCross (1 bit):** A bit that specifies whether **numCross** is calculated automatically. MUST be one of the values in the following table.

Value	Meaning
0	The value specified by <b>numCross</b> is used as the point at which the other axes in the axis group cross this value axis.
1	The <b>numCross</b> value is calculated so that the crossing point is displayed in the plot area.

**F - fLog (1 bit):** A bit that specifies whether the value axis has a logarithmic scale. MUST be one of the values in the following table.

Value	Meaning
0	The scale of the value axis is linear.
1	The scale of the value axis is logarithmic in base 10.

**G - fReversed (1 bit):** A bit that specifies whether the values on the value axis are displayed in reverse order. MUST be one of the values in the following table.

Value	Meaning
0	Values are displayed from smallest to largest from left to right or bottom to top, depending on the orientation of the axis.
1	The values are displayed in reverse order, meaning largest to smallest from left to right or bottom to top.

**H - fMaxCross (1 bit):** A bit that specifies whether the other axes in the axis group cross this value axis at the maximum value. MUST be one of the values in the following table.

Value	Meaning
0	The other axes in the axis group cross this value axis at the value specified by <b>numCross</b> .
1	The other axes in the axis group cross the value axis at the maximum value. If <b>fMaxCross</b> is set to 1, both <b>fAutoCross</b> and <b>numCross</b> MUST be ignored.

**unused (8 bits):** Undefined and MUST be ignored.

### 2.4.103 WinDoc

This record specifies the window that was selected when the **graph object** was saved.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
fChartSelected																															

**fChartSelected (1 byte):** A Boolean that specifies which window was selected when the graph object was saved. MUST be a value from the following table.

Value	Meaning
0x00	The window used to display the data sheet was selected when the graph object was saved.
0x01	The chart window was selected when the graph object was saved.

### 2.4.104 Window1

This record specifies the size and position of the **chart window** in the **OLE server** window that is contained in the parent document window. When this record follows a **MainWindow** record, the **Window1\_10** record MUST be used to define the position of the data sheet window.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
xWn																yWn															
dxWn																dyWn															
reserved																unused1															
unused2																unused3															
unused4																															

**xWn (2 bytes):** A signed integer that specifies the X location of the upper-left corner of the chart window in the OLE server window, in **twips**. SHOULD [<73>](#) be greater than or equal to zero.

**yWn (2 bytes):** A signed integer that specifies the Y location of the upper-left corner of the chart window in the OLE server window, in **twips**. SHOULD [<74>](#) be greater than or equal to zero.

**dxWn (2 bytes):** An unsigned integer that specifies the width of the chart window in the OLE server window, in twips. MUST be greater than or equal to 0x0001.

**dyWn (2 bytes):** An unsigned integer that specifies the height of the chart window in the OLE server window, in twips. MUST be greater than or equal to 0x0001.

**reserved (2 bytes):** MUST be set to zero, and MUST be ignored.

**unused1 (2 bytes):** Undefined, and MUST be ignored.

**unused2 (2 bytes):** Undefined, and MUST be ignored.

**unused3 (2 bytes):** Undefined, and MUST be ignored.

**unused4 (2 bytes):** Undefined, and MUST be ignored.

#### 2.4.105 Window1\_10

This record specifies the size and position of the data sheet window in the **OLE server** window that is contained in the parent document window. MUST immediately follow a **MainWindow** record.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
xWn																yWn															
dxWn																dyWn															
reserved																															

**xWn (2 bytes):** An unsigned integer that specifies the X location of the upper-left corner of the data sheet window in the OLE server window, in **twips**.

**yWn (2 bytes):** An unsigned integer that specifies the Y location of the upper-left corner of the data sheet window in the OLE server window, in twips.

**dxWn (2 bytes):** An unsigned integer that specifies the width of the data sheet window in the OLE server window, in twips. MUST be greater than or equal to 0x0001.

**dyWn (2 bytes):** An unsigned integer that specifies the height of the data sheet window in the OLE server window, in twips. MUST be greater than or equal to 0x0001.

**reserved (2 bytes):** MUST be set to zero, and MUST be ignored.

#### 2.4.106 Window2Graph

This record specifies the visible portion of the **datasheet**. It specifies the first row or first column to display when showing the datasheet.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
reserved1								reserved2								reserved3								reserved4							
reserved5								rowFirst																colFirst							



...	reserved6	reserved7
reserved8		

**reserved1 (1 byte):** MUST be set to 0x01, and MUST be ignored.

**reserved2 (1 byte):** MUST be set to 0x01, and MUST be ignored.

**reserved3 (1 byte):** MUST be set to 0x01, and MUST be ignored.

**reserved4 (1 byte):** MUST be set to zero, and MUST be ignored.

**reserved5 (1 byte):** MUST be set to 0x01, and MUST be ignored.

**rowFirst (2 bytes):** A **Graph\_Rw** structure (section [2.5.15](#)) that specifies the first (top-left) visible row of the datasheet. MUST be greater than or equal to 1.

**colFirst (2 bytes):** A **Graph\_Col** structure (section [2.5.14](#)) that specifies the first (top-left) visible column of the datasheet. MUST be greater than or equal to 1.

**reserved6 (1 byte):** MUST be set to 0x01, and MUST be ignored.

**reserved7 (2 bytes):** MUST be set to zero, and MUST be ignored.

**reserved8 (2 bytes):** MUST be set to zero, and MUST be ignored.

#### 2.4.107 YMult

This record specifies properties of the value multiplier for a value axis and specifies the beginning of a collection of records as defined by the chart sheet substream **ABNF**. The collection of records specifies a **display units** label.

0	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	20	1	2	3	4	5	6	7	8	9	30	1			
frtHeaderOld																																		
axmid																numLabelMultiplier																		
...																																		
...																A	B	C	reserved2															

**frtHeaderOld (4 bytes):** An **FrtHeaderOld** structure (section [2.5.10](#)). The **frtHeaderOld.rt** field MUST be set to 0x0857.

**axmid (2 bytes):** A signed integer that specifies the axis multiplier type. MUST be a value from the following table.

Value	Multiplier type
0xFFFF (-1)	Custom multiplier. Multiplier value MUST be stored in <b>numLabelMultiplier</b> .
0x0000	Values on axis are multiplied by 1.0.
0x0001	Values on axis are multiplied by 100.0.
0x0002	Values on axis are multiplied by 1000.0.

Value	Multiplier type
0x0003	Values on axis are multiplied by 10,000.0.
0x0004	Values on axis are multiplied by 100,000.0.
0x0005	Values on axis are multiplied by 1,000,000.0.
0x0006	Values on axis are multiplied by 10,000,000.0.
0x0007	Values on axis are multiplied by 100,000,000.0.
0x0008	Values on axis are multiplied by 1,000,000,000.0.
0x0009	Values on axis are multiplied by 1,000,000,000,000.0.

**numLabelMultiplier (8 bytes):** An **Xnum** structure (section [2.5.33](#)) that specifies a custom multiplier. The value on the axis is multiplied by the value of this field. MUST be greater than 0.0. If **axmid** is set to a value other than 0xFFFF, this field is ignored.

**A - reserved1 (1 bit):** MUST be set to 1, and MUST be ignored.

**B - fAutoShowMultiplier (1 bit):** A bit that specifies whether the display units label is displayed.

**C - fBeingEdited (1 bit):** A bit that specifies whether the display units label is currently being edited.

**reserved2 (13 bits):** MUST be set to zero, and MUST be ignored.

## 2.5 Structures

### 2.5.1 Boolean

An unsigned integer of size greater than 1 bit that specifies a Boolean value. MUST be a value from the following table. All other bits in the field MUST be set to zero.

Value	Meaning
0x0	Boolean value FALSE
0x1	Boolean value TRUE

### 2.5.2 CFrtId

This structure specifies a **range** of future record type ID values.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rtFirst																rtLast															

**rtFirst (2 bytes):** An unsigned integer that specifies the first future record type in the range. The value MUST be less than or equal to **rtLast**.

**rtLast (2 bytes):** An unsigned integer that specifies the last future record type in the range.

### 2.5.3 ChartNumNillable

An 8-byte union that specifies a floating-point value or a non-numeric value defined by the containing record. The type and meaning of the union contents are determined by the most significant 2 bytes, as specified in the following table:

Value of most significant 2 bytes	Type and meaning of union contents
0xFFFF	A <b>NilChartNum</b> structure (section <a href="#">2.5.21</a> ) that specifies a non-numeric value, as defined by the containing record.
Any other value.	An <b>Xnum</b> structure (section <a href="#">2.5.33</a> ) that specifies a floating-point value.

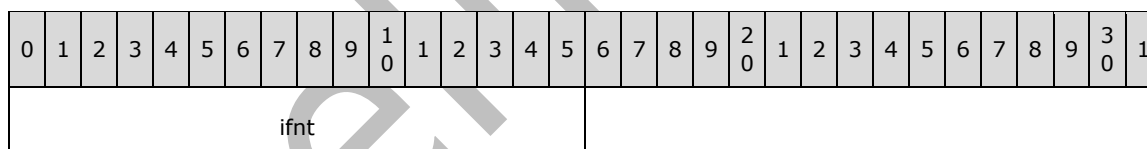
### 2.5.4 DateUnit

The **DateUnit** enumeration specifies the unit of measurement of a date value as specified in the following table.

Name	Value	Meaning
<b>DUDAYS</b>	0x0000	Time value is measured in days.
<b>DUMONTHS</b>	0x0001	Time value is measured in months.
<b>DUYEARS</b>	0x0002	Time value is measured in years.

### 2.5.5 FontIndex

This structure specifies a **Font** record in the file.



**ifnt (2 bytes):** If this value is less than 4, it specifies a zero-based index of a **Font** record in the collection of **Font** records in the chart sheet substream. If this value is greater than 4, it specifies a one-based index of a **Font** record in the collection of **Font** records in the chart sheet substream. **MUST NOT** equal 0x0004, and **MUST** be less than or equal to 0x03FE. **SHOULD** <75> be less than or equal to 0x01FE.

The **Font** records in the chart sheet substream are organized into two sections. The first section contains four **Font** records that **MUST** be present and **MUST** be ordered as shown in the following table. Note that these four default structures are saved out as identical **Font** structures. They are placeholders for their respective font formatting properties.

Value	Meaning
Zero	Default font.
1	Default <b>font</b> , bold.
2	Default font, italic.
3	Default font, bold and italic.

The second section, which is optional, contains **Font** records for any additional font formatting properties present in the file. For example, the fifth **Font** record in the file is referred to by **ifnt** value 0x0005.

## 2.5.6 FontInfo

This structure specifies a **font** entry used by the **FrtFontList** record.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	1	2	3	4	5	6	7	8	9	30	1													
A															reserved															ifnt														

**A - fScaled (1 bit):** A bit that specifies whether the fonts are scaled. MUST be a value from the following table. <76>

Value	Meaning
0x0	Font has fixed size.
0x1	Font scales with <b>chart area</b> in a chart or <b>plot area</b> .

**reserved (15 bits):** MUST be set to zero, and MUST be ignored.

**ifnt (2 bytes):** A **FontIndex** structure (section 2.5.5) that specifies the font used by the **FrtFontList** record.

## 2.5.7 FormatRun

This structure specifies formatting information for a **text run**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	1	2	3	4	5	6	7	8	9	30	1
ich															ifnt																

**ich (2 bytes):** An unsigned integer that specifies the zero-based index of the first character of the text in the **TxO** record that contains the text run. When **FormatRun** is used in an array, this value MUST be in increasing order.

**ifnt (2 bytes):** A **FontIndex** structure (section 2.5.5) that specifies the **font**. If **ich** is equal to the length of the text, this field is undefined and MUST be ignored.

## 2.5.8 FrtFlags

This structure specifies flags used in future record headers.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	1	2	3	4	5	6	7	8	9	30	1
A	B	reserved																													

**A - fFrtRef (1 bit):** A bit that specifies whether the containing record specifies a **range of cells**. MUST be a value from the following table.

Value	Meaning
0	The containing record does not specify a range of cells. The <b>ref8.rwFirst</b> , <b>ref8.rwLast</b> , <b>ref8.colFirst</b> , and <b>ref8.colLast</b> fields of the containing record MUST all be set to zero.
1	The containing record specifies a range of cells.

**B - fFrtAlert (1 bit):** A bit that specifies whether to alert the user of possible problems when saving the file without having recognized this record.

**reserved (14 bits):** MUST be set to zero, and MUST be ignored.

### 2.5.9 FrtHeader

This structure specifies a future record.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rt																grbitFrt															
reserved																															
...																															

**rt (2 bytes):** An unsigned integer that specifies the record type identifier. MUST be identical to the record type identifier of the containing record.

**grbitFrt (2 bytes):** An **FrtFlags** structure (section [2.5.8](#)) that specifies attributes for this record. The value of **grbitFrt.fFrtRef** MUST be set to zero.

**reserved (8 bytes):** MUST be set to zero, and MUST be ignored.

### 2.5.10 FrtHeaderOld

This structure specifies a future record.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rt																grbitFrt															

**rt (2 bytes):** An unsigned integer that specifies the record type identifier. MUST be identical to the record type identifier of the containing record.

**grbitFrt (2 bytes):** An **FrtFlags** structure (section [2.5.8](#)) that specifies attributes for this record. The value of **grbitFrt.fFrtRef** MUST be set to zero.

### 2.5.11 FtCf

This structure specifies the clipboard format of the picture-type **Obj** record containing this **FtCf**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
ft																cb															
cf																															

**ft (2 bytes):** Reserved. MUST be set to 0x0007.

**cb (2 bytes):** Reserved. MUST be set to 0x0002.

**cf (2 bytes):** An unsigned integer that specifies the Windows clipboard format of the data associated with the picture. MUST be a value from the following table.

Value	Format
0x0002	Specifies that the format of the picture is an enhanced <b>metafile</b> .
0x0009	Specifies that the format of the picture is a bitmap.
0xFFFF	Specifies that the picture is in an unspecified format that is neither an enhanced metafile nor a bitmap.

### 2.5.12 FtCmo

This structure specifies the common properties of the **Obj** record that contains this **FtCmo**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
ft																cb															
ot																id															
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	unused8															
...																unused9															
...																unused10															
...																															

**ft (2 bytes):** Reserved. MUST be set to 0x0015.

**cb (2 bytes):** Reserved. MUST be set to 0x0012.

**ot (2 bytes):** An unsigned integer that specifies the type of object represented by the **Obj** record containing this **FtCmo**. MUST be a value from the following table.

Value	Type of object
0x0000	Group
0x0001	Line
0x0002	Rectangle
0x0003	Oval
0x0004	Arc
0x0006	Text
0x0008	Picture
0x0009	Polygon
0x001E	OfficeArt object

**id (2 bytes):** An unsigned integer that specifies the ID of this object. This object ID is used by other types to refer to this object. The value of **id** MUST be unique among all **Obj** records within the Chart Sheet substream **ABNF**.

- A - fLocked (1 bit):** MUST be ignored and MUST be set to 1.
- B - reserved (1 bit):** Reserved. MUST be set to zero.
- C - fDefaultSize (1 bit):** MUST be ignored and MUST be set to zero.
- D - fPublished (1 bit):** MUST be set to zero and MUST be ignored.
- E - fPrint (1 bit):** MUST be ignored and MUST be set to 1.
- F - unused1 (1 bit):** Undefined and MUST be ignored.
- G - unused2 (1 bit):** Undefined and MUST be ignored
- H - fDisabled (1 bit):** MUST be ignored and MUST be set to zero.
- I - fUIObj (1 bit):** MUST be ignored and MUST be set to zero.
- J - fRecalcObj (1 bit):** Undefined and MUST be ignored.
- K - unused3 (1 bit):** Undefined and MUST be ignored.
- L - unused4 (1 bit):** Undefined and MUST be ignored.
- M - fRecalcObjAlways (1 bit):** Undefined and MUST be ignored.
- N - unused5 (1 bit):** Undefined and MUST be ignored.
- O - unused6 (1 bit):** Undefined and MUST be ignored.
- P - unused7 (1 bit):** Undefined and MUST be ignored.
- unused8 (4 bytes):** Undefined and MUST be ignored.

**unused9 (4 bytes):** Undefined and MUST be ignored.

**unused10 (4 bytes):** Undefined and MUST be ignored.

### 2.5.13 FtGmo

This structure appears in a group-type **Obj** record.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
ft																cb															
unused																															

**ft (2 bytes):** Reserved. MUST be set to 0x0006.

**cb (2 bytes):** Reserved. MUST be set to 0x0002

**unused (2 bytes):** Undefined and MUST be ignored.

### 2.5.14 Graph\_Col

This structure specifies a column index in a **datasheet**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
col																															

**col (2 bytes):** An unsigned integer that specifies a zero-based index of a column in the datasheet that contains this structure. MUST be less than or equal to 0x0F9F.

### 2.5.15 Graph\_Rw

This structure specifies a row index in a **datasheet**.

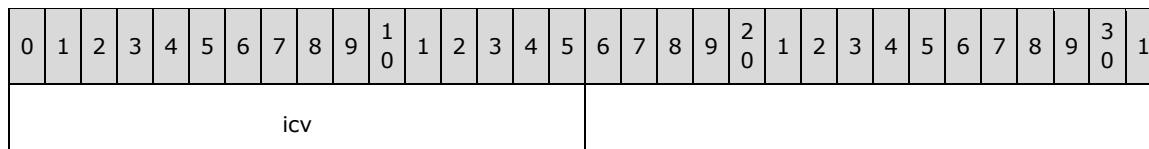
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rw																															

**rw (2 bytes):** An unsigned integer that specifies a zero-based index of a row in the datasheet that contains this structure. MUST be less than or equal to 0x0F9F.

### 2.5.16 Icv

This structure specifies a color in the color table.





**icv (2 bytes):** An unsigned integer that specifies a color from the color table. The value MUST be one of the following values:

- A value greater than or equal to 0x0000 and less than or equal to 0x0041.
- A value greater than or equal to 0x004D and less than or equal to 0x004F.
- A value of 0x7FFF.

The values that are greater than or equal to 0x0000 and less than or equal to 0x0007 specify built-in color constants as specified by the following table.

icv value	Color description	Red value	Green value	Blue value
0x0000	Black	0	0	0
0x0001	White	255	255	255
0x0002	Red	255	0	0
0x0003	Green	0	255	0
0x0004	Blue	0	0	255
0x0005	Yellow	255	255	0
0x0006	Magenta	255	0	255
0x0007	Cyan	0	255	255

The next 56 values in the table, the **icv** values greater than or equal to 0x0008 and less than or equal to 0x003F, specify the palette colors in the table. If a **Palette** record exists in this file, these **icv** values specify colors from the **rgColor** array in the **Palette** record. If no **Palette** record exists, these values specify colors in the default palette. The next 56 values in this part of the color table are specified as shown in the following table.

Value	If a Palette record exists in this file	Default red value (if no palette record in file)	Default green value (if no palette record in file)	Default blue value (if no palette record in file)
0x0008	Field <b>rgColor</b> [0] of Palette	0	0	0
0x0009	Field <b>rgColor</b> [1] of Palette	255	255	255
0x000A	Field <b>rgColor</b> [2] of Palette	255	0	0
0x000B	Field <b>rgColor</b> [3] of Palette	0	255	0
0x000C	Field <b>rgColor</b> [4] of Palette	0	0	255
0x000D	Field <b>rgColor</b> [5] of Palette	255	255	0
0x000E	Field <b>rgColor</b> [6] of Palette	255	0	255
0x000F	Field <b>rgColor</b> [7] of Palette	0	255	255
0x0010	Field <b>rgColor</b> [8] of Palette	128	0	0
0x0011	Field <b>rgColor</b> [9] of Palette	0	128	0

<b>Value</b>	<b>If a Palette record exists in this file</b>	<b>Default red value (if no palette record in file)</b>	<b>Default green value (if no palette record in file)</b>	<b>Default blue value (if no palette record in file)</b>
0x0012	Field <b>rgColor</b> [10] of Palette	0	0	128
0x0013	Field <b>rgColor</b> [11] of Palette	128	128	0
0x0014	Field <b>rgColor</b> [12] of Palette	128	0	128
0x0015	Field <b>rgColor</b> [13] of Palette	0	128	128
0x0016	Field <b>rgColor</b> [14] of Palette	192	192	192
0x0017	Field <b>rgColor</b> [15] of Palette	128	128	128
0x0018	Field <b>rgColor</b> [16] of Palette	153	153	255
0x0019	Field <b>rgColor</b> [17] of Palette	153	51	102
0x001A	Field <b>rgColor</b> [18] of Palette	255	255	204
0x001B	Field <b>rgColor</b> [19] of Palette	204	255	255
0x001C	Field <b>rgColor</b> [20] of Palette	102	0	102
0x001D	Field <b>rgColor</b> [21] of Palette	255	128	128
0x001E	Field <b>rgColor</b> [22] of Palette	0	102	204
0x001F	Field <b>rgColor</b> [23] of Palette	204	204	255
0x0020	Field <b>rgColor</b> [24] of Palette	0	0	128
0x0021	Field <b>rgColor</b> [25] of Palette	255	0	255
0x0022	Field <b>rgColor</b> [26] of Palette	255	255	0
0x0023	Field <b>rgColor</b> [27] of Palette	0	255	255
0x0024	Field <b>rgColor</b> [28] of Palette	128	0	128
0x0025	Field <b>rgColor</b> [29] of Palette	128	0	0
0x0026	Field <b>rgColor</b> [30] of Palette	0	128	128
0x0027	Field <b>rgColor</b> [31] of Palette	0	0	255
0x0028	Field <b>rgColor</b> [32] of Palette	0	204	255
0x0029	Field <b>rgColor</b> [33] of Palette	204	255	255
0x002A	Field <b>rgColor</b> [34] of Palette	204	255	204
0x002B	Field <b>rgColor</b> [35] of Palette	255	255	153
0x002C	Field <b>rgColor</b> [36] of Palette	153	204	255
0x002D	Field <b>rgColor</b> [37] of Palette	255	153	204
0x002E	Field <b>rgColor</b> [38] of Palette	204	153	255
0x002F	Field <b>rgColor</b> [39] of Palette	255	204	153
0x0030	Field <b>rgColor</b> [40] of Palette	51	102	255
0x0031	Field <b>rgColor</b> [41] of Palette	51	204	204

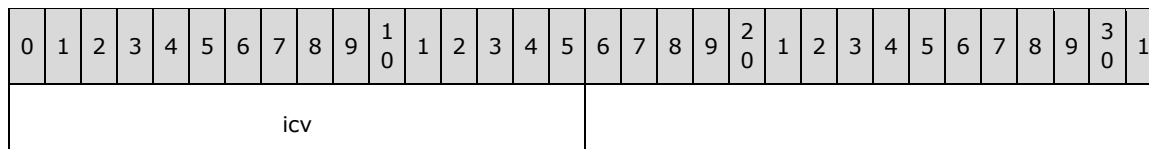
Value	If a Palette record exists in this file	Default red value (if no palette record in file)	Default green value (if no palette record in file)	Default blue value (if no palette record in file)
0x0032	Field <b>rgColor</b> [42] of Palette	153	204	0
0x0033	Field <b>rgColor</b> [43] of Palette	255	204	0
0x0034	Field <b>rgColor</b> [44] of Palette	255	153	0
0x0035	Field <b>rgColor</b> [45] of Palette	255	102	0
0x0036	Field <b>rgColor</b> [46] of Palette	102	102	153
0x0037	Field <b>rgColor</b> [47] of Palette	150	150	150
0x0038	Field <b>rgColor</b> [48] of Palette	0	51	102
0x0039	Field <b>rgColor</b> [49] of Palette	51	153	102
0x003A	Field <b>rgColor</b> [50] of Palette	0	51	0
0x003B	Field <b>rgColor</b> [51] of Palette	51	51	0
0x003C	Field <b>rgColor</b> [52] of Palette	153	51	0
0x003D	Field <b>rgColor</b> [53] of Palette	153	51	102
0x003E	Field <b>rgColor</b> [54] of Palette	51	51	153
0x003F	Field <b>rgColor</b> [55] of Palette	51	51	51

The remaining values in the color table specify colors associated with application display settings as specified by the following table.

Value	Meaning
0x0040	Default foreground color. This is the window text color in the data sheet display.
0x0041	Default background color. This is the window background color in the data sheet display and is the default background color for a <b>cell</b> .
0x004D	Default chart foreground color. This is the window text color in the chart display.
0x004E	Default chart background color. This is the window background color in the chart display.
0x004F	Chart neutral color, which is black, an RGB value of (0, 0, 0).
0x7FFF	<b>Font</b> automatic color. This is the window text color.

### 2.5.17 IcvChart

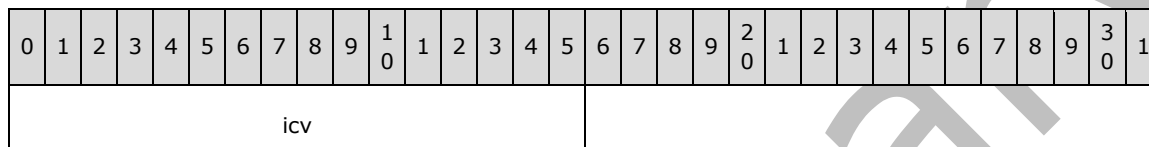
This structure specifies a color in the chart color table. The chart color table is a subset of the full color table. See **Icv** (section [2.5.16](#)) for more information about the colors in the chart color table.



**icv (2 bytes):** An **Icv** structure (section 2.5.16) that specifies a color from the chart color table. MUST be greater than or equal to 0x0008 and less than or equal to 0x003F, or greater than or equal to 0x004D and less than or equal to 0x004F.

### 2.5.18 IcvFont

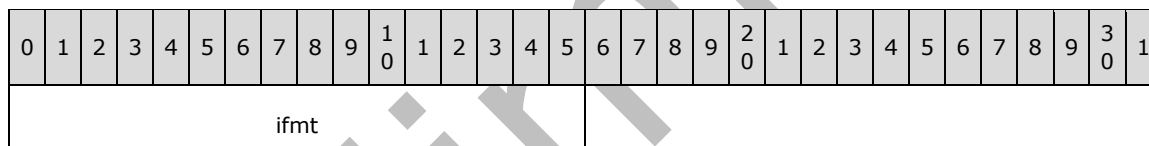
This structure specifies a color that is used by **fonts**. The font colors are a subset of the full color table.



**icv (2 bytes):** An **Icv** structure (section 2.5.16) that specifies a font color. MUST be greater than or equal to 0x0008 and less than or equal to 0x003F, or be set to 0x7FFF.

### 2.5.19 IFmt

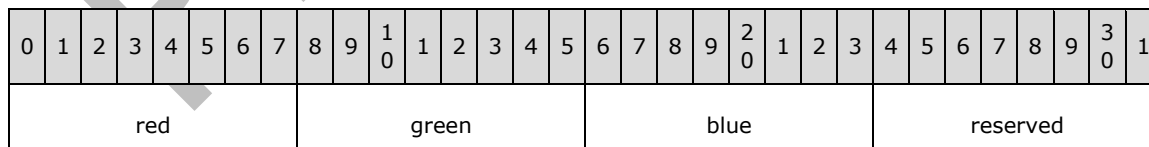
This structure specifies the identifier of a **number format**.



**ifmt (2 bytes):** An unsigned integer that specifies the identifier of a number format. The identifier specified by this field MUST be a valid built-in number format identifier or the identifier of a custom number format as specified using a **Format** record. Custom number format identifiers MUST be greater than or equal to 0x00A4 and less than or equal to 0x0188, and SHOULD be less than or equal to 0x017E. The built-in number formats are listed in [ECMA-376] Part 4 section 3.8.30: Markup Language Reference.

### 2.5.20 LongRGB

This structure specifies a color as a combination of red, green, and blue.



**red (1 byte):** An unsigned integer that specifies the relative intensity of red.

**green (1 byte):** An unsigned integer that specifies the relative intensity of green.

**blue (1 byte):** An unsigned integer that specifies the relative intensity of blue.

**reserved (1 byte):** MUST be set to zero, and MUST be ignored.

### 2.5.21 NilChartNum

A structure that specifies a non-numeric value (also known as "NaN" or "Not a Number") that is used in place of a numeric value.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
unused																															
type																reserved															

**unused (4 bytes):** Undefined and MUST be ignored.

**type (2 bytes):** An unsigned integer that specifies the interpretation of this value. This field is undefined and MUST be ignored, unless defined by the containing record.

**reserved (2 bytes):** MUST be set to 0xFFFF and MUST be ignored.

### 2.5.22 OfficeArtClientAnchorChart

This structure specifies the **anchor** position of a **drawing object**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31					
rh																																				
...																																				
A	B	C	D	E	unused																lx1															
...																ly1																				
...																lx2																				
...																ly2																				
...																																				

**rh (8 bytes):** An **OfficeArtRecordHeader** (as specified in [\[MS-ODRAW\]](#) section 2.2.1) that specifies the header for this structure. The sub-fields of **OfficeArtRecordHeader** (as specified in [\[MS-ODRAW\]](#) section 2.2.1) are further specified in the following table:

Field	Meaning
<b>rh.recVer</b>	MUST be set to 0x0000.
<b>rh.recInstance</b>	MUST be set to 0x0000.
<b>rh.recType</b>	MUST be set to 0xF010.
<b>rh.recLen</b>	An unsigned integer that specifies the number of bytes following the header. This value MUST be set to 0x0012.

**A - fMove (1 bit):** MUST be set to zero and MUST be ignored.

**B - fSize (1 bit):** A bit that specifies whether the shape sizes with the **chart area**. MUST be a value from the following table.

Value	Meaning
0	Sizes with the chart area.
1	Does not size with the chart area.

**C - reserved1 (1 bit):** MUST be set to zero and MUST be ignored

**D - reserved2 (1 bit):** Undefined and MUST be ignored.

**E - reserved3 (1 bit):** MUST be set to zero and MUST be ignored.

**unused (11 bits):** Undefined and MUST be ignored.

**lx1 (4 bytes):** A signed integer that specifies the horizontal offset of the upper-left corner of the **bounding rectangle** of the drawing object, relative to the upper-left corner of the chart area, in SPRC.

**ly1 (4 bytes):** A signed integer that specifies the vertical offset of the upper-left corner of the bounding rectangle of the drawing object, relative to the upper-left corner of the chart area, in SPRC.

**lx2 (4 bytes):** A signed integer that specifies the horizontal offset of the bottom-right corner of the bounding rectangle of the drawing object, relative to the upper-left corner of the chart area, in SPRC.

**ly2 (4 bytes):** A signed integer that specifies the vertical offset of the bottom-right corner of the bounding rectangle of the drawing object, relative to the upper-left corner of the chart area, in SPRC.

### 2.5.23 OfficeArtClientData

This structure specifies the client data of a **drawing object**. This structure MUST be at the last structure of the **rgChildRec** field of the **MsoDrawing** record. The next record MUST be **Obj**, which contains the detailed data information about this drawing object.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rh																															
...																															

**rh (8 bytes):** An **OfficeArtRecordHeader**, as specified in [\[MS-ODRAW\]](#) section 2.2.1, that specifies the header for this structure. The subfields of **OfficeArtRecordHeader** are further specified in the following table.

Field	Meaning
<b>rh.recVer</b>	MUST be set to 0x0000.
<b>rh.recInstance</b>	MUST be set to 0x0000.
<b>rh.recType</b>	MUST be set to 0xF011.
<b>rh.recLen</b>	An unsigned integer that specifies the number of bytes following the header. The value MUST be set

Field	Meaning
	to zero.

### 2.5.24 OfficeArtClientTextbox

This structure specifies the client textbox of a **drawing object**. This structure MUST be the last structure of the **rgChildRec** field of the **MsoDrawing** record. The next record MUST be **TxO**, which contains the detailed textbox information about this drawing object.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rh																															
...																															

**rh (8 bytes):** An **OfficeArtRecordHeader**, as specified in [\[MS-ODRAW\]](#) section 2.2.1, that specifies the header for this structure. The subfields of **OfficeArtRecordHeader** are further specified in the following table.

Field	Meaning
<b>rh.recVer</b>	MUST be set to 0x0000.
<b>rh.recInstance</b>	MUST be set to 0x0000.
<b>rh.recType</b>	MUST be set to 0xF00D.
<b>rh.recLen</b>	An unsigned integer that specifies the number of bytes following the header. This value MUST be set to zero.

### 2.5.25 PositionMode

This structure specifies positioning mode for position information saved in a **Pos** record, as specified by the following table.

Name	Value	Meaning
<b>MDFX</b>	0x0000	Relative position to the chart, in <b>points</b> .
<b>MDABS</b>	0x0001	Absolute width and height in points. Can be applied only to the <b>mdBotRt</b> field of a <b>Pos</b> record.
<b>MDPARENT</b>	0x0002	Owner of the <b>Pos</b> record determines how to interpret the position data.
<b>MDKTH</b>	0x0003	Offset to default position, in thousandths of the <b>plot area</b> size.
<b>MDCHART</b>	0x0005	Relative position to the chart, in <b>SPRC</b> .

### 2.5.26 Run

This structure specifies formatting information for a **text run**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
formatRun																															
unused1																unused2															

**formatRun (4 bytes):** A **FormatRun** structure (section [2.5.7](#)) that specifies the formatting information of this record.

**unused1 (2 bytes):** Undefined and MUST be ignored.

**unused2 (2 bytes):** Undefined and MUST be ignored.

### 2.5.27 ShortXLUnicodeString

This structure specifies a Unicode string.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
cch								A	reserved							rgb (variable)															
...																															

**cch (1 bytes):** An unsigned integer that specifies the count of characters in the string.

**A - fHighByte (1 bit):** A bit that specifies whether the characters in **rgb** are double-byte characters. MUST be a value from the following table.

Value	Meaning
0x0	All the characters in the string have a high byte of 0x00 and only the low bytes are in <b>rgb</b> .
0x1	All the characters in the string are saved as double-byte characters in <b>rgb</b> .

**reserved (7 bits):** MUST be set to zero, and MUST be ignored.

**rgb (variable):** An array of bytes that specifies the characters. If **fHighByte** is set to 0x0, the size of the array MUST be equal to the value of **cch**. If **fHighByte** is set to 0x1, the size of the array MUST be equal to the value of **cch** times 2.

### 2.5.28 TxOLastRun

This structure marks the end of the **formatting run** information in the **TxORuns** structure (section [2.5.29](#)).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
cchText																unused1															
unused2																															



**cchText (2 bytes):** An unsigned integer that specifies the number of characters in the preceding **TxO** record. The value **MUST** be the count of characters specified in the **cchText** field of the preceding **TxO** record.

**unused1 (2 bytes):** Undefined and **MUST** be ignored.

**unused2 (4 bytes):** Undefined and **MUST** be ignored.

### 2.5.29 TxORuns

This structure specifies the **formatting run** information for the **TxO** record and zero or more **Continue** records immediately following.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
rgTxoRuns (variable)																															
...																															
lastRun																															
...																															

**rgTxoRuns (variable):** An array of **Run** structures (section 2.5.26). Each **Run** structure specifies the formatting information for a **text run**. The **formatRun.ich** field **MUST** be less than the **cchText** field of the preceding **TxO** record. The number of elements in this array is calculated as follows:

$(\text{cbRuns of the preceding TxO record} / 8 - 1)$

**lastRun (8 bytes):** A **TxOLastRun** structure (section 2.5.28) that marks the end of the text run. This field is present only in the last **Continue** record following the **TxO** record.

### 2.5.30 XLUnicodeString

This structure specifies a **Unicode** string.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
cch																A	reserved							rgb (variable)								
...																																

**cch (2 bytes):** An unsigned integer that specifies the count of characters in the string.

**A - fHighByte (1 bit):** A bit that specifies whether the characters in **rgb** are double-byte characters. **MUST** be a value from the following table.

Value	Meaning
0x0	All the characters in the string have a high byte of 0x00, and only the low bytes are in <b>rgb</b> .
0x1	All the characters in the string are saved as 2-byte Unicode characters in <b>rgb</b> .

**reserved2 (7 bits):** **MUST** be set to zero, and **MUST** be ignored.

**rgb (variable):** An array of bytes that specifies the characters. If **fHighByte** is set to 0x0, the size of the array MUST be equal to **cch**. If **fHighByte** is set to 0x1, the size of the array MUST be equal to **cch** times 2.

### 2.5.31 XLUnicodeStringMin2

This structure specifies a **Unicode** string.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
cch																	st (variable)														
...																															

**cch (2 bytes):** An unsigned integer that specifies the count of characters in the string. MUST be equal to the number of characters in **st**.

**st (variable):** An optional **XLUnicodeStringNoCch** structure (section 2.5.32) that specifies the string. MUST exist if, and only if, **cch** is greater than zero.

### 2.5.32 XLUnicodeStringNoCch

This structure specifies a Unicode string. When an **XLUnicodeStringNoCch** structure (section 2.5.32) is used, the count of characters in the string MUST be specified in the structure that uses the **XLUnicodeStringNoCch**.

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
A	reserved							rgb (variable)																							
...																															

**A - fHighByte (1 bit):** A bit that specifies whether the characters in **rgb** are double-byte characters. MUST be a value from the following table.

Value	Meaning
0x0	All the characters in the string have a high byte of 0x00, and only the low bytes are in <b>rgb</b> .
0x1	All the characters in the string are saved as double-byte characters in <b>rgb</b> .

**reserved2 (7 bits):** MUST be set to zero, and MUST be ignored.

**rgb (variable):** An array of bytes that specifies the characters. If **fHighByte** is set to 0x0, the size of the array MUST be equal to the count of characters in the string. If **fHighByte** is set to 0x1, the size of the array MUST be twice the count of characters in the string.

### 2.5.33 Xnum

A 64-bit binary **floating-point number**, as defined in [\[IEEE754\]](#). This value MUST NOT be infinity, denormalized, not-a-number (NaN), or negative zero.

### 3 Structure Examples

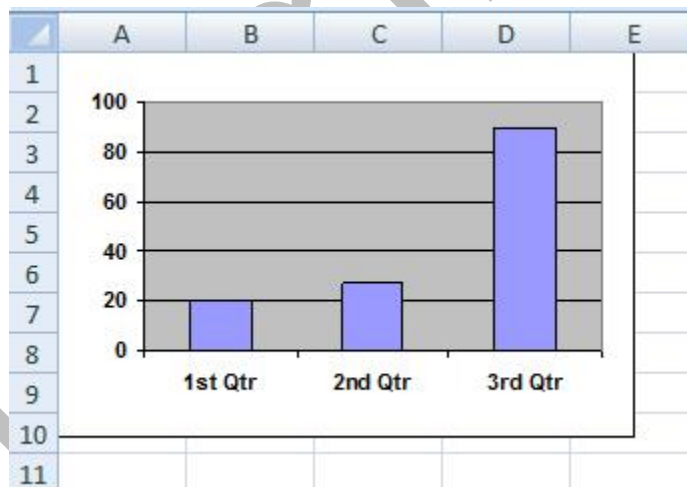
This section contains examples of some of the most commonly used data structures in files in this file format. The examples are meant to be a starting point for an implementer learning the file format. They are not meant to cover all records in the file format.

The following conventions are followed for all of the examples, unless noted otherwise:

- The order of the records, structures, and fields in the example match the corresponding order in the file format.
- The examples begin with the first record relevant to the example and end with the last record relevant to the example. An example cannot be used as a complete and stand-alone file.
- The examples are self-contained and contiguous. No records or structures are omitted in the middle of an example.
- Undefined and ignored fields are not included in the field explanations.
- Offsets for records and structures are omitted because these values vary, depending on how the files are created and on what optional records applications include in the files.

#### 3.1 Graph - Column Chart

This example shows a column chart created using a **graph object**. The set of records that specifies the column chart exists in the chart sheet substream of the file. For brevity, not all records in the chart sheet substream are included in the example. The column chart specified in this example has a single series with three data points, as shown in the following figure.



**Figure 3: Column chart in a sheet (1)**

This example includes the following records:

- The **Chart**, **Sci**, **PlotGrowth**, **ShtProps**, and **AxesUsed** records specify properties of the chart.
- The **Series**, **Begin**, **BRAI**, **SeriesText**, **DataFormat**, and **SerToCrt** records specify properties of the data series.
- The **AxisParent**, **Pos**, **Axis**, **CatSerRange**, **Tick**, **ValueRange**, **AxisLine**, and **LineFormat** records specify properties of the axis group in the chart.

- The **Frame**, **LineFormat**, and **AreaFormat** records specify the properties of the **plot area**.
- The **ChartFormat** and **Bar** records specify properties of the chart group.

### 3.1.1 Graph - Column Chart: Chart

The first record in this example, **Chart**, specifies the position and dimension of the **chart area** and specifies the beginning of a collection of records that specifies the chart.

Size	Structure	Value
0010	<a href="#">Chart</a> - <b>Chart</b>	
0004	FixedPoint - <b>x</b>	0x00000000
0004	FixedPoint - <b>y</b>	0x00000000
0004	FixedPoint - <b>dx</b>	0x00D80000
0004	FixedPoint - <b>dy</b>	0x00900000

**Figure 4: Structure of the Chart record**

**x:** 0x00000000 specifies that the horizontal position of the upper-left corner of the chart is zero **points**.

**y:** 0x00000000 specifies that the vertical position of the upper-left corner of the chart is zero points.

**dx:** 0x00D80000 specifies that the width of the chart is 14155776 points.

**dy:** 0x00900000 specifies that the height of the chart is 9437184 points.

Records following this record, and before the next **Scl** record, are omitted for brevity.

### 3.1.2 Graph - Column Chart: Scl

The next record in this example, **Scl**, specifies the **zoom level** of the current view in the window used to display the **chart window**. This record specifies that the zoom level is 100 percent.

Size	Structure	Value
0004	<a href="#">Scl</a> - <b>Scl</b>	
0002	SHORT - <b>nscl</b>	0x0001
0002	SHORT - <b>dscl</b>	0x0001

**Figure 5: Structure of the Scl record**

**nscl:** 0x0001 specifies that the numerator of the fraction is 1.

**dscl:** 0x0001 specifies that the denominator of the fraction is 1.

### 3.1.3 Graph - Column Chart: PlotGrowth

The next record in this example, **PlotGrowth**, specifies the scale factors for **font scaling** in the **plot area**.

Size	Structure	Value
0008	<a href="#">PlotGrowth</a> - PlotGrowth	
0004	FixedPoint - dxPlotGrowth	0x00010000
0004	FixedPoint - dyPlotGrowth	0x00010000

**Figure 6: Structure of the PlotGrowth record**

**dxPlotGrowth:** 0x00010000 specifies that the horizontal growth of the plot area is 0x00010000 **points**.

**dyPlotGrowth:** 0x00010000 specifies that the vertical growth of the plot area is 0x00010000 points.

### 3.1.4 Graph - Column Chart: Series

The next record in this example, **Series**, specifies the series of the chart and the type and size of the data that makes up the series.

Size	Structure	Value
000C	<a href="#">Series</a> - Series	
0002	USHORT - sdtX	0x0003
0002	USHORT - sdtY	0x0001
0002	USHORT - cValx	0x0003
0002	USHORT - cValy	0x0003
0002	USHORT - sdtBSize	0x0001
0002	USHORT - cValBSize	0x0000

**Figure 7: Structure of the Series record**

**sdtX:** 0x0003 specifies that the **categories** contain text information.

**sdtY:** 0x0001 specifies that the values contain numeric information

**cValx:** 0x0003 specifies that the count of categories is 3.

**cValy:** 0x0003 specifies that the count of values is 3.

### 3.1.5 Graph - Column Chart: Begin

The next record in this example, **Begin**, specifies the beginning of the collection of records that specifies properties in the series.

Size	Structure
0000	<a href="#">Begin</a> - Begin

**Figure 8: Structure of the Begin record**

### 3.1.6 Graph - Column Chart: BRAI

The next record in this example, **BRAI**, specifies a reference to a **cell** in the data sheet that specifies the series name.

Size	Structure	Value
0008	<a href="#">BRAI</a> - BRAI	
0001	BYTE - <b>id</b>	0x00
0001	BYTE - <b>rt</b>	0x01
1 bit	USHORT - <b>fUnlinkedIfmt</b>	0x0
1 bit	USHORT - <b>fRowCol</b>	0x1
14 bits	USHORT - <b>reserved</b>	0x0000
0002	USHORT - <b>ifmt</b>	0x0000
0002	USHORT - <b>rowcol</b>	0x0001

**Figure 9: Structure of the BRAI record**

**id:** 0x00 specifies that the referenced data is used for the series name.

**rt:** 0x01 specifies that the type of data that is being referenced is specified by **rowcol**.

**fUnlinkedIfmt:** 0x0 specifies that the series name uses the number formatting of the referenced data.

**rowcol:** 0x0001 specifies that the row number of the series name is 1.

### 3.1.7 Graph - Column Chart: SeriesText

The next record in this example, **SeriesText**, specifies the text for the series.

Size	Structure	Value
000C	<a href="#">SeriesText</a> - SeriesText	
0002	USHORT - <b>reserved</b>	0x0000
000A	<a href="#">ShortXLUnicodeString</a> - <b>stText</b>	East

**Figure 10: Structure of the SeriesText record**

**stText:** The East record specifies the text for the series whose name is "East".

### 3.1.8 Graph - Column Chart: BRAI{1}

The next record in this example, **BRAI**, specifies a reference to a range in the data sheet that specifies the series values.

Size	Structure	Value
0008	<a href="#">BRAI</a> - BRAI	
0001	BYTE - <b>id</b>	0x01
0001	BYTE - <b>rt</b>	0x01
1 bit	USHORT - <b>fUnlinkedIfmt</b>	0x0
1 bit	USHORT - <b>fRowCol</b>	0x1

Size	Structure	Value
14 bits	USHORT - <b>reserved</b>	0x0000
0002	USHORT - <b>ifmt</b>	0x0000
0002	USHORT - <b>rowcol</b>	0x0001

**Figure 11: Structure of the BRAI record**

**id:** 0x01 specifies that the referenced data is used for series values.

**rt:** 0x01 specifies that the type of data that is being referenced is specified by **rowcol**.

**fUnlinkedIfmt:** 0x0 specifies that the series values use the number formatting of the referenced data.

**rowcol:** 0x0001 specifies that the row number of the series values is 1.

### 3.1.9 Graph - Column Chart: BRAI{2}

The next record in this example, **BRAI**, specifies a reference to a range in the data sheet that specifies the series **category** names.

Size	Structure	Value
0008	<a href="#">BRAI</a> - <b>BRAI</b>	
0001	BYTE - <b>id</b>	0x02
0001	BYTE - <b>rt</b>	0x01
1 bit	USHORT - <b>fUnlinkedIfmt</b>	0x0
1 bit	USHORT - <b>fRowCol</b>	0x1
14 bits	USHORT - <b>reserved</b>	0x0000
0002	USHORT - <b>ifmt</b>	0x0000
0002	USHORT - <b>rowcol</b>	0x0000

**Figure 12: Structure of the BRAI record**

**id:** 0x02 specifies that the referenced data is used for series category names.

**rt:** 0x01 specifies that the type of data that is being referenced is specified by **rowcol**.

**fUnlinkedIfmt:** 0x0 specifies that the series category names use the number formatting of the referenced data.

**rowcol:** 0x0000 specifies that the row number for series category names is zero.

### 3.1.10 Graph - Column Chart: DataFormat

The next record in this example, **DataFormat**, specifies the series to which the formatting information applies.

Size	Structure	Value
0008	<a href="#">DataFormat</a> - <b>DataFormat</b>	

Size	Structure	Value
0002	USHORT - <b>xi</b>	0xFFFF
0002	USHORT - <b>yi</b>	0x0000
0002	SHORT - <b>iss</b>	0x0000
0002	SHORT - <b>Reserved</b>	0x0000

**Figure 13: Structure of the DataFormat record**

**xi:** 0xFFFF specifies that this record determines the format of a series.

**yi:** 0x0000 specifies that the zero-based index into a **Series** record in the collection of **Series** records in the current chart sheet substream is zero.

**iss:** 0x0000 specifies that the plot order of the data series is zero.

Records following this record, and before the next **SerToCrt** record, are omitted for brevity.

### 3.1.11 Graph - Column Chart: SerToCrt

The next record in this example, **SerToCrt**, specifies the chart group for the series.

Size	Structure	Value
0002	<a href="#">SerToCrt</a> - <b>SerToCrt</b>	
0002	USHORT - <b>id</b>	0x0000

**Figure 14: Structure of the SerToCrt record**

**id:** 0x0000 specifies the first **ChartFormat** record in the collection of **ChartFormat** records in the current chart sheet substream. The referenced **ChartFormat** record specifies the chart group that contains the current series.

Records following this record, and before the next **ShtProps** record, are omitted for brevity.

### 3.1.12 Graph - Column Chart: ShtProps

The next record in this example, **ShtProps**, specifies properties of the chart.

Size	Structure	Value
0004	<a href="#">ShtProps</a> - <b>ShtProps</b>	
1 bit	USHORT - <b>fManSerAlloc</b>	0x1
1 bit	USHORT - <b>unused1</b>	0x0
1 bit	USHORT - <b>reserved1</b>	0x0
1 bit	USHORT - <b>fManPlotArea</b>	0x1
1 bit	USHORT - <b>fAlwaysAutoPlotArea</b>	0x0
11 bits	USHORT - <b>reserved2</b>	0x000
0001	BYTE - <b>mdBlank</b>	0x00



Size	Structure	Value
0001	BYTE - <b>reserved3</b>	0x00

**Figure 15: Structure of the ShtProps record**

**fManSerAlloc:** 0x0001 specifies that the series is automatically allocated for the chart.

**fManPlotArea:** 0x0001 is ignored because **fAlwaysAutoPlotArea** is set to 0x0000.

**fAlwaysAutoPlotArea:** 0x0000 specifies that the default **plot area** dimension is used, regardless of the **Pos** record information.

**mdBlank:** 0x00 specifies that empty **cells** are not plotted.

### 3.1.13 Graph - Column Chart: DefaultText

The next record in this example, **DefaultText**, specifies the text elements that are formatted using the information specified by the **Text** record immediately following this record.

Size	Structure	Value
0002	<a href="#">DefaultText</a> - <b>DefaultText</b>	
0002	USHORT - <b>id</b>	0x0002

**Figure 16: Structure of the DefaultText record**

**id:** 0x0002 specifies that all text in the chart that does not use scalable fonts is set by the **Text** record immediately following this record.

### 3.1.14 Graph - Column Chart: Text

The next record in this example, **Text**, specifies the position and appearance of text elements specified by the previous **DefaultText** record.

Size	Structure	Value
0020	<a href="#">Text</a> - <b>Text</b>	
0001	BYTE - <b>at</b>	0x02
0001	BYTE - <b>vat</b>	0x02
0002	WORD - <b>wBkgMode</b>	0x0001
0004	<a href="#">LongRGB</a> - <b>rgbText</b>	
0001	BYTE - <b>red</b>	0x00
0001	BYTE - <b>green</b>	0x00
0001	BYTE - <b>blue</b>	0x00
0001	BYTE - <b>reserved</b>	0x00
0004	LONG - <b>x</b>	0xFFFFFFFFB8
0004	LONG - <b>y</b>	0xFFFFFFFF92
0004	LONG - <b>dx</b>	0x00000000

Size	Structure	Value
0004	LONG - <b>dy</b>	0x00000000
1 bit	USHORT - <b>fAutoColor</b>	0x1
1 bit	USHORT - <b>fShowKey</b>	0x0
1 bit	USHORT - <b>fShowValue</b>	0x0
1 bit	USHORT - <b>unused1</b>	0x0
1 bit	USHORT - <b>fAutoText</b>	0x1
1 bit	USHORT - <b>fGenerated</b>	0x1
1 bit	USHORT - <b>fDeleted</b>	0x0
1 bit	USHORT - <b>fAutoMode</b>	0x1
3 bits	USHORT - <b>unused2</b>	0x0
1 bit	USHORT - <b>fShowLabelAndPerc</b>	0x0
1 bit	USHORT - <b>fShowPercent</b>	0x0
1 bit	USHORT - <b>fShowBubbleSizes</b>	0x0
1 bit	USHORT - <b>fShowLabel</b>	0x0
1 bit	USHORT - <b>reserved</b>	0x0
0002	<a href="#">Icv</a> - <b>icvText</b>	
0002	USHORT - <b>icv</b>	0x004D
4 bits	USHORT - <b>dlp</b>	0x0
10 bits	USHORT - <b>unused3</b>	0x150
2 bits	USHORT - <b>iReadingOrder</b>	0x0
0002	USHORT - <b>trot</b>	0x0000

**Figure 17: Structure of the Text record**

**at:** 0x02 specifies that the **horizontal alignment** of the text is set to center alignment.

**vat:** 0x02 specifies that the **vertical alignment** of the text is set to center alignment.

**wBkgMode:** 0x0001 specifies that the background of the text is transparent.

**rgbText:** Specifies that the color of the text is black.

**rgbText.red:** 0x00 specifies that the relative intensity of red is set to zero.

**rgbText.green:** 0x00 specifies that the relative intensity of green is set to zero.

**rgbText.blue:** 0x00 specifies that the relative intensity of blue is set to zero.

**x:** 0xFFFFFFFFB8 is ignored because this record is preceded by the **DefaultText** record.

**y:** 0xFFFFFFFF92 is ignored because this record is preceded by the **DefaultText** record.

**dx:** 0x00000000 is ignored because this record is preceded by the **DefaultText** record.

**dy:** 0x00000000 is ignored because this record is preceded by the **DefaultText** record

**fAutoColor:** 0x0001 specifies that the foreground color is determined automatically.

**fShowKey:** 0x0000 specifies that the text is not attached to a **legend key**.

**fShowValue:** 0x0000 specifies the text does not contain the value of a data point.

**fAutoText:** 0x0001 specifies that the text value of the text field is automatically generated and has not been changed.

**fGenerated:** 0x0001 specifies that the properties of the text field are automatically generated and have not been changed.

**fDeleted:** 0x0000 specifies that this data label has not been deleted by the user.

**fAutoMode:** 0x0001 specifies that the **background color** is determined automatically.

**icvText:** An **Icv** structure (section 2.5.16) that specifies the color of the text.

**icvText.icv:** 0x004D specifies the window text color in the chart display.

**iReadingOrder:** 0x0000 specifies that the **reading order** is set according to the **DefaultText** settings of the chart.

**trot:** 0x0000 specifies that the text is not rotated.

The **Begin** record following this record is omitted for brevity.

### 3.1.15 Graph - Column Chart: FontX

The next record in this example, **FontX**, specifies the font for the text element specified by the preceding **DefaultText** record.

Size	Structure	Value
0002	<a href="#">FontX</a> - <b>FontX</b>	
0002	USHORT - <b>iFont</b>	0x0000

**Figure 18: Structure of the FontX record**

**iFont:** 0x0000 specifies that the default **font** of the chart is used.

Records following this record, and before the next **AxesUsed** record, are omitted for brevity.

### 3.1.16 Graph - Column Chart: AxesUsed

The next record in this example, **AxesUsed**, specifies the number of axis groups on the chart.

Size	Structure	Value
0002	<a href="#">AxesUsed</a> - <b>AxesUsed</b>	
0002	USHORT - <b>cAxes</b>	0x0001

**Figure 19: Structure of the AxesUsed record**

**cAxes:** 0x0001 specifies that a single primary axis group is present on the chart.

### 3.1.17 Graph - Column Chart: AxisParent

The next record in this example, **AxisParent**, specifies properties of the primary axis group.

Size	Structure	Value
0012	<a href="#">AxisParent</a> - <b>AxisParent</b>	
0002	SHORT - <b>iax</b>	0x0000
0010	unused - <b>unused</b>	23 02 00 00 B8 01 00 00 ED 0C 00 00 BB 0A 00 00

**Figure 20: Structure of the AxisParent record**

**iax:** 0x0000 specifies that the axis group is primary.

The **Begin** record following this record is omitted for brevity.

### 3.1.18 Graph - Column Chart: Pos

The next record in this example, **Pos**, specifies the size and position for the primary axis group of the **plot area**.

Size	Structure	Value
0014	<a href="#">Pos</a> - <b>PosCh</b>	
0002	<a href="#">PositionMode</a> - <b>mdTopLt</b>	0x0002
0002	PositionMode - <b>mdBotRt</b>	0x0002
0002	SHORT - <b>x1</b>	0x0090
0002	SHORT - <b>unused1</b>	0x0000
0002	SHORT - <b>y1</b>	0x00DC
0002	SHORT - <b>unused2</b>	0x0000
0002	SHORT - <b>x2</b>	0x0E80
0002	SHORT - <b>unused3</b>	0x0000
0002	SHORT - <b>y2</b>	0x0DE8
0002	SHORT - <b>unused4</b>	0x0000

**Figure 21: Structure of the PosCh record**

**mdTopLt:** 0x0002, combined with the value of **mdBotRt**, determines that **x1** and **y1** specify the horizontal and vertical offsets of the upper-left corner of the primary axis group, relative to the upper-left corner of the **chart area**, in SPRC, and that **x2** and **y2** specify the width and height of the primary axis group, in SPRC.

**mdBotRt:** 0x0002, combined with the value of **mdTopLt**, determines that **x1** and **y1** specify the horizontal and vertical offsets of the upper-left corner of the primary axis group, relative to the upper-left corner of the chart area, in SPRC, and that **x2** and **y2** specify the width and height of the primary axis group, in SPRC.

**x1:** 0x0090 specifies the horizontal offsets of the upper-left corner of the primary axis group, relative to the upper-left corner of the chart area, as 144 SPRC.

**y1:** 0x00DC specifies the vertical offsets of the upper-left corner of the primary axis group, relative to the upper-left corner of the chart area, as 220 SPRC.

**x2:** 0x0E80 specifies that the width of the primary axis group is 3712 SPRC.

**y2:** 0x0DE8 specifies that the height of the primary axis group is 3560 SPRC.

### 3.1.19 Graph - Column Chart: Axis

The next record in this example, **Axis**, specifies properties of the **category** axis.

Size	Structure	Value
0012	<a href="#">Axis</a> - <b>Axis</b>	
0002	WORD - <b>wType</b>	0x0000
0004	ULONG - <b>reserved1</b>	0x00000000
0004	ULONG - <b>reserved2</b>	0x00000000
0004	ULONG - <b>reserved3</b>	0x00000000
0004	ULONG - <b>reserved4</b>	0x00000000

**Figure 22: Structure of the Axis record**

**wType:** 0x0000 specifies that the axis type is category axis.

The **Begin** record following this record is omitted for brevity.

### 3.1.20 Graph - Column Chart: CatSerRange

The next record in this example, **CatSerRange**, specifies the properties of a **category** axis.

Size	Structure	Value
0008	<a href="#">CatSerRange</a> - <b>XRange</b>	
0002	SHORT - <b>catCross</b>	0x0001
0002	SHORT - <b>catLabel</b>	0x0001
0002	SHORT - <b>catMark</b>	0x0001
1 bit	USHORT - <b>fBetween</b>	0x1
1 bit	USHORT - <b>fMaxCross</b>	0x0
1 bit	USHORT - <b>fReverse</b>	0x0
13 bits	USHORT - <b>reserved</b>	0x0000

**Figure 23: Structure of the XRange record**

**catCross:** 0x0001 specifies that the category axis is crossed by the value axis at the first category.

**catLabel:** 0x0001 specifies that the interval between axis labels on this axis is 1.

**catMark:** 0x0001 specifies that the interval at which **major tick marks** and **minor tick marks** are displayed on the axis is 1.

**fBetween:** 0x0001 specifies that the value axis crosses the category axis between major tick marks.

**fMaxCross:** 0x0000 specifies that the value axis crosses the category axis at the value specified by **catCross**.

**fReverse:** 0x0000 specifies that the category axis is displayed in order

Records following this record, and before the next **Tick** record, are omitted for brevity.

### 3.1.21 Graph - Column Chart: Tick

The next record in this example, **Tick**, specifies the properties of **tick marks** associated with the **category** axis.

Size	Structure	Value
001E	<a href="#">Tick</a> - <b>Tick</b>	
0001	BYTE - <b>tkMajor</b>	0x02
0001	BYTE - <b>tkMinor</b>	0x00
0001	BYTE - <b>tlt</b>	0x03
0001	BYTE - <b>wBkgMode</b>	0x01
0004	<a href="#">LongRGB</a> - <b>rgb</b>	
0001	BYTE - <b>red</b>	0x00
0001	BYTE - <b>green</b>	0x00
0001	BYTE - <b>blue</b>	0x00
0001	BYTE - <b>reserved</b>	0x00
0004	LONG - <b>reserved1</b>	0x00000000
0004	LONG - <b>reserved2</b>	0x00000000
0004	LONG - <b>reserved3</b>	0x00000000
0004	LONG - <b>reserved4</b>	0x00000000
1 bit	USHORT - <b>fAutoCo</b>	0x1
1 bit	USHORT - <b>fAutoMode</b>	0x1
3 bits	USHORT - <b>rot</b>	0x0
1 bit	USHORT - <b>fAutoRot</b>	0x1
8 bits	USHORT - <b>unused</b>	0x00
2 bits	USHORT - <b>iReadingOrder</b>	0x0
0002	<a href="#">IcvChart</a> - <b>icv</b>	
0002	USHORT - <b>icv</b>	0x004D
0002	SHORT - <b>trot</b>	0x0000

**Figure 24: Structure of the Tick record**

**tktMajor:** 0x02 specifies that the location of the **major tick marks** is outside, which indicates that the major tick marks are drawn away from the **plot area**.

**tktMinor:** 0x00 specifies that no **minor tick marks** are present on the axis.

**tl:** 0x03 specifies that the tick mark labels are placed next to the axis.

**wBkgMode:** 0x01 specifies that a transparent background is drawn for the text of the tick mark labels.

**rgb:** Ignored because **fAutoCo** is 1.

**fAutoCo:** 0x0001 specifies that the text uses an automatically selected foreground color, based on the display settings of the computer.

**fAutoMode:** 0x0001 specifies that the background mode is set according to the preceding **DefaultText** record in the chart.

**rot:** 0x0000 is ignored because **fAutoRot** is 1.

**fAutoRot:** 0x0001 specifies that the text rotation of the tick mark labels is determined automatically.

Records following this record, and before the next **Axis** record, are omitted for brevity.

### 3.1.22 Graph - Column Chart: Axis{1}

The next record in this example, **Axis**, specifies properties of the value axis.

Size	Structure	Value
0012	<a href="#">Axis</a> - <b>Axis</b>	
0002	WORD - <b>wType</b>	0x0001
0004	ULONG - <b>reserved1</b>	0x00000000
0004	ULONG - <b>reserved2</b>	0x00000000
0004	ULONG - <b>reserved3</b>	0x00000000
0004	ULONG - <b>reserved4</b>	0x00000000

**Figure 25: Structure of the Axis record**

**wType:** 0x0001 specifies that the axis type is value axis.

The **Begin** record following this record is omitted for brevity.

### 3.1.23 Graph - Column Chart: ValueRange

The next record in this example, **ValueRange**, specifies properties of the value axis.

Size	Structure	Value
002A	<a href="#">ValueRange</a> - <b>ValueRange</b>	
0008	<a href="#">Xnum</a> - <b>numMin</b>	0x0000000000000000
0008	Xnum - <b>numMax</b>	0x0000000000000000
0008	Xnum - <b>numMajor</b>	0x0000000000000000

Size	Structure	Value
0008	Xnum - <b>numMinor</b>	0x0000000000000000
0008	Xnum - <b>numCross</b>	0x0000000000000000
1 bit	USHORT - <b>fAutoMin</b>	0x1
1 bit	USHORT - <b>fAutoMax</b>	0x1
1 bit	USHORT - <b>fAutoMajor</b>	0x1
1 bit	USHORT - <b>fAutoMinor</b>	0x1
1 bit	USHORT - <b>fAutoCross</b>	0x1
1 bit	USHORT - <b>fLog</b>	0x0
1 bit	USHORT - <b>fReversed</b>	0x0
1 bit	USHORT - <b>fMaxCross</b>	0x0
8 bits	USHORT - <b>unused</b>	0x01

**Figure 26: Structure of the ValueRange record**

**numMin:** 0x0000000000000000 is ignored because **fAutoMin** is set to 1.

**numMax:** 0x0000000000000000 is ignored because **fAutoMax** is set to 1.

**numMajor:** 0x0000000000000000 is ignored because **fAutoMajor** is set to 1.

**numMinor:** 0x0000000000000000 is ignored because **fAutoMinor** is set to 1.

**numCross:** 0x0000000000000000 is ignored because **fAutoCross** is set to 1.

**fAutoMin:** 0x0001 specifies that **numMin** is calculated automatically.

**fAutoMax:** 0x0001 specifies that **numMax** is calculated automatically.

**fAutoMajor:** 0x0001 specifies that **numMajor** is calculated automatically.

**fAutoMinor:** 0x0001 specifies that **numMinor** is calculated automatically.

**fAutoCross:** 0x0001 specifies that **numCross** is calculated automatically.

**fLog:** 0x0000 specifies that the scale of the value axis is linear.

**fReversed:** 0x0000 specifies that values are displayed from left to right.

Records following this record, and before the next **Tick** record, are omitted for brevity.

### 3.1.24 Graph - Column Chart: Tick{1}

The next record in this example, **Tick**, specifies the properties of **tick marks** associated with the value axis.

Size	Structure	Value
001E	<a href="#">Tick</a> - <b>Tick</b>	
0001	BYTE - <b>tkMajor</b>	0x02



Size	Structure	Value
0001	BYTE - <b>tkMinor</b>	0x00
0001	BYTE - <b>tl</b>	0x03
0001	BYTE - <b>wBkgMode</b>	0x01
0004	<a href="#">LongRGB</a> - <b>rgb</b>	
0001	BYTE - <b>red</b>	0x00
0001	BYTE - <b>green</b>	0x00
0001	BYTE - <b>blue</b>	0x00
0001	BYTE - <b>reserved</b>	0x00
0004	LONG - <b>reserved1</b>	0x00000000
0004	LONG - <b>reserved2</b>	0x00000000
0004	LONG - <b>reserved3</b>	0x00000000
0004	LONG - <b>reserved4</b>	0x00000000
1 bit	USHORT - <b>fAutoCo</b>	0x1
1 bit	USHORT - <b>fAutoMode</b>	0x1
3 bits	USHORT - <b>rot</b>	0x0
1 bit	USHORT - <b>fAutoRot</b>	0x1
8 bits	USHORT - <b>unused</b>	0x00
2 bits	USHORT - <b>iReadingOrder</b>	0x0
0002	<a href="#">IcvChart</a> - <b>icv</b>	
0002	USHORT - <b>icv</b>	0x004D
0002	SHORT - <b>trot</b>	0x0000

**Figure 27: Structure of the Tick record**

**tktMajor:** 0x02 specifies that the location of the **major tick marks** is outside, which indicates that the major tick marks are drawn away from the **plot area**.

**tktMinor:** 0x00 specifies that no **minor tick marks** are present on the axis.

**tl:** 0x03 specifies that the tick mark labels are placed next to the axis.

**wBkgMode:** 0x01 specifies that the background of the text of the axis labels is transparent.

**rgb:** is ignored because **fAutoCo** is set to 1.

**fAutoCo:** 0x0001 specifies that the text of the axis labels uses an automatically selected foreground color, based on the display settings of the computer.

**fAutoMode:** 0x0001 specifies that the background color of the axis label is set according to the preceding **DefaultText** record in the chart.

**rot:** 0x0000 is ignored because **fAutoRot** is set to 1.

**fAutoRot:** 0x0001 specifies that the text rotation of the axis labels is determined automatically.

### 3.1.25 Graph - Column Chart: AxisLine

The next record in this example, **AxisLine**, specifies which part of the value axis is specified by the **LineFormat** record that follows this record.

Size	Structure	Value
0002	<a href="#">AxisLine</a> - AxisLine	
0002	USHORT - id	0x0001

**Figure 28: Structure of the AxisLine record**

**id:** 0x0001 specifies that the **major gridlines** along the value axis are defined by the **LineFormat** record that follows this record.

### 3.1.26 Graph - Column Chart: LineFormat

The next record in this example, **LineFormat**, specifies the appearance of the value axis line.

Size	Structure	Value
000C	<a href="#">LineFormat</a> - LineFormat	
0004	<a href="#">LongRGB</a> - rgb	
0001	BYTE - red	0x00
0001	BYTE - green	0x00
0001	BYTE - blue	0x00
0001	BYTE - reserved	0x00
0002	USHORT - <b>Ins</b>	0x0000
0002	SHORT - <b>we</b>	0x0000
1 bit	USHORT - <b>fAuto</b>	0x1
1 bit	USHORT - <b>reserved1</b>	0x0
1 bit	USHORT - <b>fAxisOn</b>	0x0
1 bit	USHORT - <b>fAutoCo</b>	0x1
12 bits	USHORT - <b>reserved2</b>	0x000
0002	<a href="#">IcvChart</a> - icv	
0002	USHORT - icv	0x004D

**Figure 29: Structure of the LineFormat record**

**rgb:** This field is ignored because **fAuto** is set to 1.

**Ins:** 0x0000 is ignored because **fAuto** is set to 1.

**we:** 0x0000 is ignored because **fAuto** is set to 1.

**fAuto:** 0x0001 specifies that the line has default formatting.

**fAxisOn:** 0x0000 specifies that the axis line is displayed.

**fAutoCo:** 0x0001 specifies that **icv** is set to 0x004D.

**icv:** Ignored because **fAuto** is set to 1.

Records following this record, and before the next **Frame** record, are omitted for brevity.

### 3.1.27 Graph - Column Chart: Frame

The next record in this example, **Frame**, specifies the type, size, and position of the frame around the chart.

Size	Structure	Value
0004	<a href="#">Frame</a> - <b>Frame</b>	
0002	USHORT - <b>frt</b>	0x0000
1 bit	USHORT - <b>fAutoSize</b>	0x1
1 bit	USHORT - <b>fAutoPosition</b>	0x1
14 bits	USHORT - <b>reserved</b>	0x0000

**Figure 30: Structure of the Frame record**

**frt:** 0x0000 specifies that a frame is drawn around the chart element.

**fAutoSize:** 0x0001 specifies that the size of the frame is automatically calculated.

**fAutoPosition:** 0x0001 specifies that the position of the frame is automatically calculated.

Records following this record, and before the next **LineFormat** record, are omitted for brevity.

### 3.1.28 Graph - Column Chart: LineFormat{1}

The next record in this example, **LineFormat**, specifies the appearance of the outline of the **plot area**.

Size	Structure	Value
000C	<a href="#">LineFormat</a> - <b>LineFormat</b>	
0004	<a href="#">LongRGB</a> - <b>rgb</b>	
0001	BYTE - <b>red</b>	0x80
0001	BYTE - <b>green</b>	0x80
0001	BYTE - <b>blue</b>	0x80
0001	BYTE - <b>reserved</b>	0x00
0002	USHORT - <b>Ins</b>	0x0000
0002	SHORT - <b>we</b>	0x0000
1 bit	USHORT - <b>fAuto</b>	0x0

Size	Structure	Value
1 bit	USHORT - <b>reserved1</b>	0x0
1 bit	USHORT - <b>fAxisOn</b>	0x0
1 bit	USHORT - <b>fAutoCo</b>	0x0
12 bits	USHORT - <b>reserved2</b>	0x000
0002	<a href="#">IcvChart</a> - <b>icv</b>	
0002	USHORT - <b>icv</b>	0x0017

**Figure 31: Structure of the LineFormat record**

**rgb:** A **LongRGB** structure (section 2.5.20) that specifies that the color of the line is gray.

**rgb.red:** 0x80 specifies that the relative intensity of red is set to 0x80.

**rgb.green:** 0x80 specifies that the relative intensity of green is set to 0x80.

**rgb.blue:** 0x80 specifies that the relative intensity of blue is set to 0x80.

**Ins:** 0x0000 specifies that the style of the line is solid.

**we:** 0x0000 specifies that the thickness of the line is narrow (single).

**fAuto:** 0x0000 specifies that this line has formatting as specified by **Ins**, **we**, and **icv**.

**fAxisOn:** 0x0000 specifies that the axis line is displayed.

**fAutoCo:** 0x0000 specifies that **icv.icv** is not set to 0x004D.

**icv:** Specifies that the color of the line is gray.

**icv.icv:** 0x0017 specifies that the color of the line has the following composition: red is set to 128, green is set to 128, and blue is set to 128.

### 3.1.29 Graph - Column Chart: AreaFormat

The next record in this example, **AreaFormat**, specifies the patterns and colors used in the **plot area** of the chart.

Size	Structure	Value
0010	<a href="#">AreaFormat</a> - <b>AreaFormat</b>	
0004	<a href="#">LongRGB</a> - <b>rgbFore</b>	
0001	BYTE - <b>red</b>	0xC0
0001	BYTE - <b>green</b>	0xC0
0001	BYTE - <b>blue</b>	0xC0
0001	BYTE - <b>reserved</b>	0x00
0004	LongRGB - <b>rgbBack</b>	
0001	BYTE - <b>red</b>	0x00

Size	Structure	Value
0001	BYTE - <b>green</b>	0x00
0001	BYTE - <b>blue</b>	0x00
0001	BYTE - <b>reserved</b>	0x00
0002	SHORT - <b>fls</b>	0x0001
1 bit	WORD - <b>fAuto</b>	0x0
1 bit	WORD - <b>fInvertNeg</b>	0x0
14 bits	WORD - <b>reserved</b>	0x0000
0002	<a href="#">IcvChart</a> - <b>icvFore</b>	
0002	USHORT - <b>icv</b>	0x0016
0002	IcvChart - <b>icvBack</b>	
0002	USHORT - <b>icv</b>	0x004F

**Figure 32: Structure of the AreaFormat record**

**rgbFore:** A **LongRGB** structure (section 2.5.20) that specifies that the **foreground color** of the **fill pattern** is gray.

**rgbFore.red:** 0xC0 specifies that the relative intensity of red is set to 0xC0.

**rgbFore.green:** 0xC0 specifies that the relative intensity of green is set to 0xC0.

**rgbFore.blue:** 0xC0 specifies that the relative intensity of blue is set to 0xC0.

**rgbBack:** A **LongRGB** structure (section 2.5.20) that specifies that the **background color** of the fill pattern is black.

**rgbBack.red:** 0x00 specifies that the relative intensity of red is set to 0x00.

**rgbBack.green:** 0x00 specifies that the relative intensity of green is set to 0x00.

**rgbBack.blue:** 0x00 specifies that the relative intensity of blue is set to 0x00.

**fls:** 0x0001 specifies that the fill pattern is solid.

**fAuto:** 0x0000 specifies that the **fill** colors are not set automatically.

**fInvertNeg:** 0x0000 specifies that the foreground and background are not swapped when the data value of the filled area is negative.

**icvFore:** Specifies that the foreground color of the fill pattern is gray.

**icvFore.icv:** 0x0016 specifies that the color of the line has the following composition: red is set to 192, green is set to 192, and blue is set to 192.

**icvBack:** Specifies that the background color of the fill pattern is black.

**icvBack.icv:** 0x004F specifies that the color of the line has the following composition: red is set to zero, green is set to zero, and blue is set to zero.

### 3.1.30 Graph - Column Chart: ChartFormat

The next record in this example, **ChartFormat**, specifies properties of the chart group for the data series.

Size	Structure	Value
0014	<a href="#">ChartFormat</a> - <b>ChartFormat</b>	
0004	LONG - <b>reserved1</b>	0x00000000
0004	LONG - <b>reserved2</b>	0x00000000
0004	LONG - <b>reserved3</b>	0x00000000
0004	LONG - <b>reserved4</b>	0x00000000
1 bit	WORD - <b>fVaried</b>	0x0
15 bits	WORD - <b>reserved5</b>	0x0000
0002	SHORT - <b>icrt</b>	0x0000

**Figure 33: Structure of the ChartFormat record**

**fVaried:** 0x0000 specifies that the color for each data point and the color and type for each **data marker** do not vary.

**icrt:** 0x0000 specifies that the drawing order of the chart group, relative to the other chart groups, is at the bottom of the **z-order**.

The **Begin** record following this record is omitted for brevity.

### 3.1.31 Graph - Column Chart: Bar

The next record in this example, **Bar**, specifies that the chart group is a bar chart group and specifies the chart group attributes.

Size	Structure	Value
0006	<a href="#">Bar</a> - <b>Bar</b>	
0002	SHORT - <b>pcOverlap</b>	0x0000
0002	USHORT - <b>pcGap</b>	0x0096
1 bit	USHORT - <b>fTranspose</b>	0x0
1 bit	USHORT - <b>fStacked</b>	0x0
1 bit	USHORT - <b>f100</b>	0x0
1 bit	USHORT - <b>fHasShadow</b>	0x0
12 bits	USHORT - <b>reserved</b>	0x000

**Figure 34: Structure of the Bar record**

**pcOverlap:** 0x0000 specifies that there is no overlap between data points.

**pcGap:** 0x0096 specifies that the width of the gap between the **categories** and the left and right edges of the **plot area** is 150 percent of the data point width, divided by 2.

**fTranspose:** 0x0000 specifies that the data points and the value axis are vertical.

**fStacked:** 0x0000 specifies that the data points in the chart group that share the same category are not stacked.

**f100:** 0x0000 specifies that the data points in the chart group are not displayed as a percentage of the sum of all data points in the chart group that share the same category.

**fHasShadow:** 0x0000 specifies that none of the data points in the chart group have shadows.

### 3.2 Graph - Pie Chart

This example illustrates a pie chart created using a **graph object** and has three data points. Records already covered in the preceding column chart example and other records not relevant to this example are omitted for brevity.

This example begins with the **Orient** record.

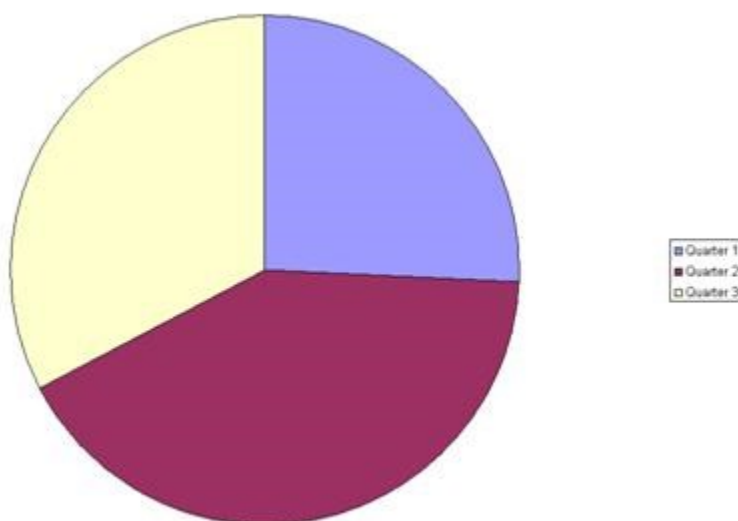


Figure 4: Pie chart in a sheet (1)

#### 3.2.1 Graph - Pie Chart: Orient

The first record in this example, **Orient**, specifies how the series data of the chart is arranged.

Size	Structure	Value
0006	<a href="#">Orient</a> - <b>Orient</b>	
0001	BYTE - <b>fSeriesInRows</b>	0x01
0002	USHORT - <b>rowSeriesX</b>	0x0000
0002	USHORT - <b>colSeriesX</b>	0x0000
0001	BYTE - <b>Reserved3</b>	0x01

Figure 36: Structure of the Orient record

**fSeriesInRows:** 0x01 specifies that the series is arranged by rows.

**rowSeriesX:** 0x0000 is required in this field because this is a pie chart.

**colSeriesX:** 0x0000 is required in this field because this is a pie chart.

**Reserved3:** 0x01 is required in this field because this is reserved.

Records following this record, and before the next **Chart** record, are omitted for brevity.

### 3.2.2 Graph - Pie Chart: Chart

The next record in this example, **Chart**, specifies the beginning of the collection of records for this chart and the position and dimension of the **chart area**.

Size	Structure	Value
0010	<a href="#">Chart</a> - Chart	
0004	FixedPoint - <b>x</b>	0x00000000
0004	FixedPoint - <b>y</b>	0x00000000
0004	FixedPoint - <b>dx</b>	0x00D80000
0004	FixedPoint - <b>dy</b>	0x00900000

**Figure 37: Structure of the Chart record**

**x:** 0x00000000 specifies that the horizontal position of the upper-left corner of the chart is zero points. This value is required to be zero.

**y:** 0x00000000 specifies that the vertical position of the upper-left corner of the chart is zero points. This value is required to be zero.

**dx:** 0x00D80000 specifies that the width of the chart is 0x00D80000 (14155776) points.

**dy:** 0x00900000 specifies that the height of the chart is 0x00900000 (9437184) points.

Records following this record, and before the next **ShtProps** record, are omitted for brevity.

### 3.2.3 Graph - Pie Chart: ShtProps

The next record in this example, **ShtProps**, specifies **chart sheet** properties.

Size	Structure	Value
0004	<a href="#">ShtProps</a> - ShtProps	
1 bit	USHORT - <b>fManSerAlloc</b>	0x1
1 bit	USHORT - <b>unused1</b>	0x0
1 bit	USHORT - <b>reserved1</b>	0x0
1 bit	USHORT - <b>fManPlotArea</b>	0x1
1 bit	USHORT - <b>fAlwaysAutoPlotArea</b>	0x0
11 bits	USHORT - <b>reserved2</b>	0x000



Size	Structure	Value
0001	BYTE - <b>mdBlank</b>	0x00
0001	BYTE - <b>reserved3</b>	0x00

**Figure 38: Structure of the ShtProps record**

**fManSerAlloc:** 0x1 specifies that the series is automatically allocated for the chart.

**fManPlotArea:** 0x1 is ignored in this field because **fAlwaysAutoPlotArea** is set to zero.

**fAlwaysAutoPlotArea:** 0x0 specifies that the default **plot area** dimension is used, regardless of the **Pos** record information.

**mdBlank:** 0x00 specifies that empty cells are not plotted.

Records following this record, and before the next **AxesUsed** record, are omitted for brevity.

### 3.2.4 Graph - Pie Chart: AxesUsed

The next record in this example, **AxesUsed**, specifies the number of axis groups on the chart.

Size	Structure	Value
0002	<a href="#">AxesUsed</a> - <b>AxesUsed</b>	
0002	USHORT - <b>cAxes</b>	0x0001

**Figure 39: Structure of the AxesUsed record**

**cAxes:** 0x0001 specifies that a single primary value axis is present and used on the chart.

### 3.2.5 Graph - Pie Chart: AxisParent

The next record in this example, **AxisParent**, specifies properties of an axis group.

Size	Structure	Value
0012	<a href="#">AxisParent</a> - <b>AxisParent</b>	
0002	<a href="#">Boolean</a> - <b>iax</b>	0x0000
0010	DWORD - <b>unused</b>	23 02 00 00 B8 01 00 00 FB 07 00 00 31 0C 00 00

**Figure 40: Structure of the AxisParent record**

**iax:** 0x0000 specifies that the axis group is primary.

Records following this record, and before the next **ChartFormat** record, are omitted for brevity.

### 3.2.6 Graph - Pie Chart: ChartFormat

The next record in this example, **ChartFormat**, specifies properties of a chart group.

Size	Structure	Value
0014	<a href="#">ChartFormat</a> - <b>ChartFormat</b>	
0004	LONG - <b>reserved1</b>	0x00000000

Size	Structure	Value
0004	LONG - <b>reserved2</b>	0x00000000
0004	LONG - <b>reserved3</b>	0x00000000
0004	LONG - <b>reserved4</b>	0x00000000
1 bit	WORD - <b>fVaried</b>	0x1
15 bits	WORD - <b>reserved5</b>	0x0000
0002	SHORT - <b>icrt</b>	0x0000

**Figure 41: Structure of the ChartFormat record**

**fVaried:** 0x1 specifies that the color for each data point and the color and type for each **data marker** vary.

**icrt:** 0x0000 specifies the drawing order of the chart group relative to the other chart groups; 0x0000 is the bottom of the **z-order**.

Records following this record, and before the next **Pie** record, are omitted for brevity.

### 3.2.7 Graph - Pie Chart: Pie

The next record in this example, **Pie**, specifies that this is a pie chart group, and specifies chart group properties.

Size	Structure	Value
0006	<a href="#">Pie</a> - <b>Pie</b>	
0002	USHORT - <b>anStart</b>	0x0000
0002	USHORT - <b>pcDonut</b>	0x0000
1 bit	USHORT - <b>fHasShadow</b>	0x0
1 bit	USHORT - <b>fShowLdrLines</b>	0x1
14 bits	USHORT - <b>reserved</b>	0x0000

**Figure 42: Structure of the Pie record**

**anStart:** 0x0000 specifies that the starting angle of the first data point, clockwise from the top of the circle, is zero degrees.

**pcDonut:** 0x0000 specifies that the chart group is a pie chart group.

**fHasShadow:** 0x0 specifies that no data points in the chart group have shadows.

**fShowLdrLines:** 0x1 specifies that the **leader lines** to the data labels are shown.

Records following this record, and before the next **Legend** record, are omitted for brevity.

### 3.2.8 Graph - Pie Chart: Legend

The next record in this example, **Legend**, specifies properties of the legend, including the location on the display and its overall size.

Size	Structure	Value
0014	<a href="#">Legend</a> - Legend	
0004	ULONG - <b>x</b>	0x00000C5D
0004	ULONG - <b>y</b>	0x00000553
0004	ULONG - <b>dx</b>	0x00000309
0004	ULONG - <b>dy</b>	0x000004E5
0001	BYTE - <b>unused</b>	0x03
0001	BYTE - <b>wSpace</b>	0x01
1 bit	WORD - <b>fAutoPosition</b>	0x1
1 bit	WORD - <b>reserved1</b>	0x1
1 bit	WORD - <b>fAutoPosX</b>	0x1
1 bit	WORD - <b>fAutoPosY</b>	0x1
1 bit	WORD - <b>fVert</b>	0x1
1 bit	WORD - <b>fWasDataTable</b>	0x0
10 bits	WORD - <b>reserved2</b>	0x000

**Figure 43: Structure of the Legend record**

Fields **x**, **y**, **dx**, and **dy** are ignored in this record because a **Pos** record follows this record.

**wSpace:** 0x01 is required in this field and specifies that the space between **legend entries** is set to 1, which represents 40 **twips** between legend entries.

**fAutoPosition:** 0x1 specifies that the legend is automatically positioned.

**fAutoPosX:** 0x1 is required in this field because **fAutoPosition** is set to 1 and specifies that the x-positioning of the legend is automatic.

**fAutoPosY:** 0x1 is required in this field because **fAutoPosition** is set to 1 and specifies that the y-positioning of the legend is automatic.

**fVert:** 0x1 specifies that the layout of the legend entries contains a single column of entries.

**fWasDataTable:** 0x0 specifies that the chart does not show the legend in the data table.

### 3.2.9 Graph - Pie Chart: Begin

The next record in this example, **Begin**, specifies the beginning of a collection of records that specifies properties of the **Legend**.

Size	Structure
0000	<a href="#">Begin</a> - Begin

**Figure 44: Structure of the Begin record**

### 3.2.10 Graph - Pie Chart: Pos

The next record in this example, **Pos**, specifies the size and position for the legend.

Size	Structure	Value
0014	<a href="#">Pos</a> - <b>Pos</b>	
0002	<a href="#">PositionMode</a> - <b>mdTopLt</b>	0x0005
0002	PositionMode - <b>mdBotRt</b>	0x0002
0002	SHORT - <b>x1</b>	0x0C5D
0002	SHORT - <b>unused1</b>	0x0000
0002	SHORT - <b>y1</b>	0x0553
0002	SHORT - <b>unused2</b>	0x0000
0002	SHORT - <b>x2</b>	0x0000
0002	SHORT - <b>unused3</b>	0x0000
0002	SHORT - <b>y2</b>	0x0000
0002	SHORT - <b>unused4</b>	0x0000

**Figure 45: Structure of the Pos record**

Fields **x2** and **y2** are ignored in this record because of the combination of values **mdTopLt** and **mdBotRt**.

**mdTopLt:** 0x0005 specifies that the positioning mode of the upper-left corner for this **Legend** record is set as a relative position to the chart, in SPRC. The position information is saved in a **Pos** record.

**mdBotRt:** 0x0002 specifies that the preceding **Pos** record determines how to interpret the position data for the lower-right corner of a **Legend**.

**x1:** 0x0C5D specifies that the horizontal offset of the upper-left corner of the legend, relative to the upper-left corner of the **chart area**, is set to 0x0C5D (3165) SPRC.

**y1:** 0x0553 specifies that the vertical offset of the upper-left corner of the legend, relative to the upper-left corner of the chart area, is set to 0x0553 (1363) SPRC.

### 3.2.11 Graph - Pie Chart: Text

The next record in this example, **Text**, specifies the position and appearance of text fields that appear on the chart.

Size	Structure	Value
0020	<a href="#">Text</a> - <b>Text</b>	
0001	BYTE - <b>at</b>	0x02
0001	BYTE - <b>vat</b>	0x02
0002	WORD - <b>wBkgMode</b>	0x0001
0004	<a href="#">LongRGB</a> - <b>rgbText</b>	

Size	Structure	Value
0001	BYTE - <b>red</b>	0x00
0001	BYTE - <b>green</b>	0x00
0001	BYTE - <b>blue</b>	0x00
0001	BYTE - <b>reserved</b>	0x00
0004	LONG - <b>x</b>	0xFFFFFFFFB8
0004	LONG - <b>y</b>	0xFFFFFFFF92
0004	LONG - <b>dx</b>	0x00000000
0004	LONG - <b>dy</b>	0x00000000
1 bit	USHORT - <b>fAutoColor</b>	0x1
1 bit	USHORT - <b>fShowKey</b>	0x0
1 bit	USHORT - <b>fShowValue</b>	0x0
1 bit	USHORT - <b>unused1</b>	0x0
1 bit	USHORT - <b>fAutoText</b>	0x1
1 bit	USHORT - <b>fGenerated</b>	0x1
1 bit	USHORT - <b>fDeleted</b>	0x0
1 bit	USHORT - <b>fAutoMode</b>	0x1
3 bits	USHORT - <b>unused2</b>	0x0
1 bit	USHORT - <b>fShowLabelAndPerc</b>	0x0
1 bit	USHORT - <b>fShowPercent</b>	0x0
1 bit	USHORT - <b>fShowBubbleSizes</b>	0x0
1 bit	USHORT - <b>fShowLabel</b>	0x0
1 bit	USHORT - <b>reserved</b>	0x0
0002	<b>icv</b> - <b>icvText</b>	
0002	USHORT - <b>icv</b>	0x004D
4 bits	USHORT - <b>dlp</b>	0x0
10 bits	USHORT - <b>unused3</b>	0x3F9
2 bits	USHORT - <b>iReadingOrder</b>	0x0
0002	USHORT - <b>trot</b>	0x0000

**Figure 46: Structure of the Text record**

Fields **x**, **y**, **dx**, and **dy** are ignored in this record because a **Pos** record follows this record.

**at**: 0x02 specifies that the text fields that appear in the legend are center-aligned.

**vat**: 0x02 specifies that the text fields that appear in the legend are center-aligned.

**wBkgMode:** 0x0001 specifies that the display mode of the background of the text is transparent.

**rgbText:** A **LongRGB** structure (section 2.5.20) that specifies that the color of the text is black.

**rgbText.red:** 0x00 specifies that the relative intensity of red is set to zero.

**rgbText.green:** 0x00 specifies that the relative intensity of green is set to zero.

**rgbText.blue:** 0x00 specifies that the relative intensity of blue is set to zero.

**fAutoColor:** 0x1 specifies that the foreground text color is determined automatically.

**fShowKey:** 0x0 specifies that the text is not attached to a **legend key**.

**fShowValue:** 0x0 specifies that the text does not contain the value of a data point.

**fAutoText:** 0x1 specifies that the text value of the text field is automatically generated and unchanged.

**fGenerated:** 0x1 specifies that the properties of the text field are automatically generated and unchanged.

**fDeleted:** 0x0 specifies that no automatic text field that is displayed by default has been deleted by the user.

**fAutoMode:** 0x1 specifies that the background color is determined automatically.

**fShowLabelAndPerc:** 0x0 specifies that this text field of the pie chart is not shown.

**fShowPercent:** 0x0 specifies that this text field of the pie chart is not shown.

**fShowBubbleSizes:** 0x0 specifies that this text field of the pie chart is not shown.

**fShowLabel:** 0x0 specifies that this text field of the pie chart is not shown.

**icvText:** An **Icv** structure (section 2.5.16) that specifies the color of the text.

**icvText.icv:** 0x004D specifies that the default chart foreground color is used.

**dIpl:** 0x0 specifies that the data label positioning of the text, relative to the chart item to which the text is attached, is set to "Auto".

**iReadingOrder:** 0x0 specifies that the **reading order** of the text is based on context (the parent of the text label).

**trot:** 0x0000 specifies that the text rotation is zero degrees counter-clockwise.

### 3.2.12 Graph - Pie Chart: Begin{1}

The next record in this example, **Begin**, specifies the beginning of a collection of records that specifies properties of the data labels.

Size	Structure
0000	<a href="#">Begin</a> - <b>Begin</b>

**Figure 47: Structure of the Begin record**

Records following this record, and before the next **BRAI** record, are omitted for brevity.

### 3.2.13 Graph - Pie Chart: BRAI

The next record in this example, **BRAI**, specifies a reference to data in the **chart sheet** that is used by the **legend entry**.

Size	Structure	Value
0008	<a href="#">BRAI</a> - <b>BRAI</b>	
0001	BYTE - <b>id</b>	0x00
0001	BYTE - <b>rt</b>	0x01
1 bit	USHORT - <b>fUnlinkedIfmt</b>	0x0
1 bit	USHORT - <b>fRowCol</b>	0x1
14 bits	USHORT - <b>reserved</b>	0x0000
0002	<a href="#">IFmt</a> - <b>ifmt</b>	0x0000
0002	USHORT - <b>rowcol</b>	0x0000

**Figure 48: Structure of the BRAI record**

**id:** 0x00 specifies that the referenced data is used for the text of a legend entry.

**rt:** 0x01 specifies the type of data that is being referenced as specified by the **rowcol** field.

**fUnlinkedIfmt:** 0x0 specifies that the part of the chart specified by the **id** field uses the number formatting of the referenced data.

**fRowCol:** 0x1 is required in this field because **rt** is set to 1, and specifies that the reference is to a row or column. The referenced data is contained by the row or column stored in **rowcol**.

**ifmt:** 0x0000 specifies that the ID for number formatting information is set to "General", which means automatic.

**rowcol:** 0x0000 specifies that the referenced information is placed on the first row.

Records following this record, and before the next **End** record, are omitted for brevity.

### 3.2.14 Graph - Pie Chart: End

The last record in this example, **End**, specifies the end of the collection of records for this example.

Size	Structure
0000	<a href="#">End</a> - <b>End</b>

**Figure 49: Structure of the End record**

## 4 Security Considerations

None.

Preliminary



## 5 Appendix A: 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 Office 97
- Microsoft Office 2000
- Microsoft Office XP
- Microsoft Office 2003
- the 2007 Microsoft Office system
- Microsoft Office 2010 suites
- Microsoft Office 2013
- Microsoft Office 2016
- Microsoft Office 2019
- Microsoft Office 2021

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.

[<1> Section 2.1.7.3.1](#): When saving unsupported future records, the Graph component for Office 97, for Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 write those records at the end of the substream in which they were encountered when loading, in the order in which they were encountered.

[<2> Section 2.1.7.3.2](#): When saving unsupported future records, the Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 write those records at the end of the substream in which they were encountered when loading, in the order in which they were encountered.

[<3> Section 2.2.1.10](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can create files that do not conform to these rules.

[<4> Section 2.2.1.10](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can create files that do not conform to these rules.

[<5> Section 2.2.1.10](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can create files that do not conform to these rules.

[<6> Section 2.2.1.10](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can create files that do not conform to these rules.

[<7> Section 2.2.1.10](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can create files that do not conform to these rules.

<8> [Section 2.2.1.10](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can create files that do not conform to these rules.

<9> [Section 2.2.1.10](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can create files that do not conform to these rules.

<10> [Section 2.2.1.10](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can create files that do not conform to these rules.

<11> [Section 2.2.1.10](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can create files that do not conform to these rules.

<12> [Section 2.4.3](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 write a value for this field, but ignore it when loading a graph or chart.

<13> [Section 2.4.3](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 write a value for this field, but ignore it when loading a graph or chart.

<14> [Section 2.4.3](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can write out **Icv** values of 0x0040 and 0x0041 or **Icv** values greater than or equal to 0x0000 and less than or equal to 0x0007.

<15> [Section 2.4.3](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can write out **Icv** values of 0x0040 and 0x0041 or **Icv** values greater than or equal to 0x0000 and less than or equal to 0x0007.

<16> [Section 2.4.5](#): If the base unit is not days and the data is near the base value of the **date system** specified by the **Date1904** record, the **catMin** value can underflow and be saved as a large number.

<17> [Section 2.4.5](#): When the base unit is not days and the data is near the maximum value of the date system specified by the **Date1904** record, the **catMax** value can overflow and be saved as a small number.

<18> [Section 2.4.13](#): The Graph component for Office 97 sets **rupYear** to 0x07CC (1996).

<19> [Section 2.4.13](#): The Graph component for Office 97 shipped in the alpha version.

<20> [Section 2.4.13](#): This occurs only in the Graph component for Office 97.

<21> [Section 2.4.13](#): Specifies the Graph component for Office 97.

<22> [Section 2.4.13](#): Specifies the Graph component for Office 2000.

<23> [Section 2.4.13](#): Specifies the Graph component for Office XP.

<24> [Section 2.4.13](#): Specifies the Graph component for Office 2003.

<25> [Section 2.4.13](#): Specifies the Graph component for the 2007 Office system.

<26> [Section 2.4.13](#): Specifies the Graph component for Office 2010.

<27> [Section 2.4.13](#): Specifies the Graph component for Office 97.

<28> [Section 2.4.13](#): Specifies the Graph component for Office 2000.

<29> [Section 2.4.13](#): Specifies the Graph component for Office XP.

<30> [Section 2.4.13](#): Specifies the Graph component for Office 2003.

- [<31> Section 2.4.13](#): Specifies the Graph component for the 2007 Office system.
- [<32> Section 2.4.13](#): Specifies the Graph component for Office 2010.
- [<33> Section 2.4.15](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 support values less than or equal to 32000. A series cannot have more than 4000 data points. Values greater than 3999 are considered equal to 3999.
- [<34> Section 2.4.22](#): The Graph component for the 2007 Office system and Office 2010 ignores this bit and sets it back to zero if the file is resaved.
- [<35> Section 2.4.26](#): The Graph component for Office 2000.
- [<36> Section 2.4.26](#): The Graph component for Office 2000, Office XP, Office 2003, the 2007 Office system, or Office 2010.
- [<37> Section 2.4.26](#): The Graph component for the 2007 Office system and Office 2010.
- [<38> Section 2.4.26](#): The Graph component for Office 97.
- [<39> Section 2.4.26](#): The Graph component for Office 97.
- [<40> Section 2.4.26](#): Specifies the Graph component for Office 2000.
- [<41> Section 2.4.26](#): Specifies the Graph component for Office XP or Office 2003.
- [<42> Section 2.4.26](#): Specifies the Graph component for the 2007 Office system.
- [<43> Section 2.4.26](#): Specifies the Graph component for Office 2010.
- [<44> Section 2.4.26](#): Specifies the Graph component for Office 2000.
- [<45> Section 2.4.26](#): Specifies the Graph component for Office XP or Office 2003.
- [<46> Section 2.4.26](#): Specifies the Graph component for the 2007 Office system.
- [<47> Section 2.4.26](#): Specifies the Graph component for Office 2010.
- [<48> Section 2.4.35](#): The Graph component for the 2007 Office system or Office 2010 can sometimes save values greater than 254.
- [<49> Section 2.4.35](#): Although the maximum zero-based series number is 254, the Graph component for the 2007 Office system or Office 2010 sometimes saves values greater than 254.
- [<50> Section 2.4.48](#): This record is ignored in the Graph component for the 2007 Office system and Office 2010.
- [<51> Section 2.4.49](#): This record is ignored in the Graph component for the 2007 Office system and Office 2010.
- [<52> Section 2.4.50](#): The Graph component for Office 2003, the 2007 Office system, or Office 2010 can also write out zero.
- [<53> Section 2.4.54](#): Specifies the Graph component for Office 2000.
- [<54> Section 2.4.54](#): Specifies the Graph component for Office XP.
- [<55> Section 2.4.56](#): The Graph component for Office 97 does not save **OPT2**.
- [<56> Section 2.4.62](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can sometimes write out **Icv** values (section [2.5.16](#)) 0x0040 and 0x0041.

<57> [Section 2.4.65](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can sometimes write out **Icv** values 0x0040 and 0x0041 and can sometimes write out **Icv** values that are greater than or equal to 0x0000 and less than or equal to 0x0007.

<58> [Section 2.4.65](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can sometimes write out **Icv** values 0x0040 and 0x0041 and can sometimes write out **Icv** values that are greater than or equal to 0x0000 and less than or equal to 0x0007.

<59> [Section 2.4.96](#): Specifies the Graph component for Office 97.

<60> [Section 2.4.96](#): Specifies the Graph component for Office 2000.

<61> [Section 2.4.96](#): Specifies the Graph component for Office XP.

<62> [Section 2.4.96](#): Specifies the Graph component for Office 2003.

<63> [Section 2.4.96](#): Specifies the Graph component for the 2007 Office system.

<64> [Section 2.4.96](#): Specifies the Graph component for Office 2010.

<65> [Section 2.4.98](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can write numbers larger than 4000 due to rounding while editing small charts.

<66> [Section 2.4.98](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can write numbers larger than 4000 due to rounding while editing small charts.

<67> [Section 2.4.98](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can write numbers larger than 4000 due to rounding while editing small charts.

<68> [Section 2.4.98](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can write numbers larger than 4000 due to rounding while editing small charts.

<69> [Section 2.4.99](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can sometimes write out **Icv** values that are greater than or equal to 0x0000 and less than or equal to 0x0007.

<70> [Section 2.4.100](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 also set this field to 2 to specify justify low alignment.

<71> [Section 2.4.100](#): Justified alignment is not supported in the Graph component for the 2007 Office system and Office 2010 and is treated as middle alignment.

<72> [Section 2.4.100](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, , and Office 2010, East Asian version.

<73> [Section 2.4.104](#): When edited in the Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, or Office 2010, this value can be negative. If so, the **chart window** is repositioned to its default location.

<74> [Section 2.4.104](#): When edited in the Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, or Office 2010, this value can be negative. If so, the chart window is repositioned to its default location.

<75> [Section 2.5.5](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003, the 2007 Office system, and Office 2010 can sometimes save values 511 through 1022.

[<76> Section 2.5.6](#): This value is ignored in the Graph component for the 2007 Office system and Office 2010

[<77> Section 2.5.19](#): The Graph component for Office 97, Office 2000, Office XP, Office 2003 the 2007 Office system, and Office 2010 can write values greater than or equal to 0x017F and less than or equal to 0x0188, but they are not loaded.

Preliminary

## 6 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
<a href="#">5</a> Appendix A: Product Behavior	Updated list of supported products.	major

## 7 Index

### A

[AIRuns record](#) 49  
[Applicability](#) 16  
[Area record](#) 49  
[AreaFormat record](#) 49  
[Attached labels in charts](#) 41  
[AttachedLabel record](#) 55  
[AxcExt record](#) 56  
[AxesUsed record](#) 58  
[Axis group](#) 25  
[Axis in charts](#) 27  
[Axis record](#) 59  
[AxisLine record](#) 59  
[AxisParent record](#) 60

### B

[Bar record](#) 60  
[Begin record](#) 61  
[Blank record](#) 61  
[BOF record](#) 62  
[BOFDatasheet record](#) 64  
[Boolean structure](#) 138  
[BopPop record](#) 64  
[BopPopCustom record](#) 66  
[BoundSheet8 record](#) 67  
[BRAI record](#) 68  
[Byte ordering overview](#) 15

### C

[CatLab record](#) 69  
[CatSerRange record](#) 70  
[CFrtId structure](#) 138  
[Change tracking](#) 190  
Chart  
    [attached labels](#) 41  
    [axis components](#) 27  
    [axis groups](#) 25  
    [chart groups](#) 28  
    [chart sheets](#) 23  
    [data labels](#) 33  
    [data points](#) 32  
    [data sheets](#) 24  
    [data tables](#) 40  
    [elements and components](#) 24  
    [error bars](#) 40  
    [legends](#) 30  
    [series](#) 31  
    [SPRC measurements](#) 41  
    [trendlines](#) 39  
[Chart elements and components](#) 24  
[Chart group](#) 28  
[Chart record](#) 71  
[Chart sheet](#) 23  
[Chart sheet substream](#) 21  
[Chart3d record](#) 71  
[Chart3DBarShape record](#) 73  
[ChartColors record](#) 74  
[ChartFormat record](#) 74

[ChartFrtInfo record](#) 75  
[ChartNumNillable structure](#) 139  
Charts 23  
[Chart-specific future record](#) 19  
[ClrtClient record](#) 76  
[CodePage record](#) 77  
[Collection of records file structure](#) 19  
[ColumnWidth record](#) 77  
[Component object stream](#) 20  
[Compound file structure](#) 18  
[Conceptual overview](#) 22  
[Continue record](#) 77  
[Country record](#) 78  
[CrtLine record](#) 78  
[CrtLink record](#) 79

### D

[Dat record](#) 79  
[Data labels](#) 33  
[Data points in charts](#) 32  
[Data sheet](#) 24  
[Data tables](#) 40  
[DataFormat record](#) 79  
[DataLabExt record](#) 80  
[DataLabExtContents record](#) 81  
[Date1904 record](#) 82  
[DateUnit structure](#) 139  
[DefaultText record](#) 82  
Details  
    [charts](#) 23  
    [collection of records file structure](#) 19  
    [compound file structure](#) 18  
    [conceptual overview](#) 22  
    [file structure](#) 18  
    [future record file structure](#) 19  
    [record](#) 49  
    [record enumerations](#) 42  
    [record file structure](#) 18  
    [storages and streams file structure](#) 20  
    [stream file structure](#) 18  
    [substream file structure](#) 18  
[Dimensions record](#) 83  
[Documentation overview](#) 16  
[DropBar record](#) 83

### E

[End record](#) 84  
[EndBlock record](#) 84  
[EndObject record](#) 85  
[EOF record](#) 85  
[Error bars in charts](#) 40  
Example  
    Graph-Column chart  
        [AreaFormat](#) 172  
        [AxesUsed](#) 163  
        Axis ([section 3.1.19](#) 165, [section 3.1.22](#) 167)  
        [AxisLine](#) 170  
        [AxisParent](#) 164  
        [Bar](#) 174

[begin](#) 157  
[BRAI](#) 157  
[BRAI \(1\)](#) 158  
[BRAI \(2\)](#) 159  
[CatSerRange](#) 165  
[chart](#) 156  
[ChartFormat](#) 174  
[DataFormat](#) 159  
[DefaultText](#) 161  
[FontX](#) 163  
[Frame](#) 171  
[LineFormat for plot area outline](#) 171  
[LineFormat for value axis line](#) 170  
[PlotGrowth](#) 156  
[Pos](#) 164  
[Scl](#) 156  
[series](#) 157  
[SeriesText](#) 158  
[SerToCrt](#) 160  
[ShtProps](#) 160  
[text](#) 161  
[Tick](#) 166  
[Tick \(1\)](#) 168  
[ValueRange](#) 167  
 Graph-Pie chart  
[AxesUsed](#) 177  
[AxisParent](#) 177  
[Begin](#) 179  
[Begin \(1\)](#) 182  
[BRAI](#) 183  
[Chart](#) 176  
[ChartFormat](#) 177  
[End](#) 183  
[Legend](#) 178  
[Orient](#) 175  
[Pie](#) 178  
[Pos](#) 180  
[ShtProps](#) 176  
[Text](#) 180  
 Examples 155  
[Graph - Column Chart](#) 155  
[graph - column chart](#) 155  
[Graph - Pie Chart](#) 175  
[graph - pie chart](#) 175  
[ExcludeColumns record](#) 86  
[ExcludeRows record](#) 86  
**F**  
[Fbi record](#) 86  
[Fbi2 record](#) 87  
[Fields - vendor-extensible](#) 17  
[File structure](#) 18  
[Font record](#) 87  
[FontIndex structure](#) 139  
[FontInfo structure](#) 140  
[FontX record](#) 89  
[Format record](#) 89  
[FormatRun structure](#) 140  
[Frame record](#) 90  
[FrtFlags structure](#) 140  
[FrtFontList record](#) 91  
[FrtHeader structure](#) 141  
[FrtHeaderOld structure](#) 141  
[FrtWrapper record](#) 91  
[FtCf structure](#) 142  
[FtCmo structure](#) 142  
[FtGmo structure](#) 144  
[Future record file structure](#) 19  
[chart](#) 19  
**G**  
[GelFrame record](#) 92  
[Global substream](#) 20  
[Glossary](#) 10  
[Graph - Column Chart example](#) 155  
[Graph - column chart examples](#) 155  
[Graph - Pie Chart example](#) 175  
[Graph - pie chart examples](#) 175  
[Graph Col structure](#) 144  
[Graph Rw structure](#) 144  
 Graph-Column Chart  
[AreaFormat example](#) 172  
[AxesUsed example](#) 163  
 Axis example ([section 3.1.19](#) 165, [section 3.1.22](#) 167)  
[AxisLine example](#) 170  
[AxisParent example](#) 164  
[Bar example](#) 174  
[begin example](#) 157  
[BRAI \(1\) example](#) 158  
[BRAI \(2\) example](#) 159  
[BRAI example](#) 157  
[CatSerRange example](#) 165  
[chart example](#) 156  
[ChartFormat example](#) 174  
[DataFormat example](#) 159  
[DefaultText example](#) 161  
[FontX example](#) 163  
[Frame example](#) 171  
[LineFormat for plot area outline](#) 171  
[LineFormat for value axis line example](#) 170  
[PlotGrowth example](#) 156  
[Pos example](#) 164  
[Scl example](#) 156  
[series example](#) 157  
[SeriesText example](#) 158  
[SerToCrt example](#) 160  
[ShtProps example](#) 160  
[text example](#) 161  
[Tick \(1\) example](#) 168  
[Tick example](#) 166  
[ValueRange example](#) 167  
 Graph-Pie Chart  
[AxesUsed example](#) 177  
[AxisParent example](#) 177  
[Begin \(1\) example](#) 182  
[Begin example](#) 179  
[BRAI example](#) 183  
[Chart example](#) 176  
[ChartFormat example](#) 177  
[End example](#) 183  
[Legend example](#) 178  
[Orient example](#) 175  
[Pie example](#) 178  
[Pos example](#) 180  
[ShtProps example](#) 176  
[Text example](#) 180



**I**

[Icv structure](#) 144  
[IcvChart structure](#) 147  
[IcvFont structure](#) 148  
[IFmt structure](#) 148  
[IFmtRecord record](#) 94  
[Implementer - security considerations](#) 184  
[Informative references](#) 15  
[Introduction](#) 10

**L**

[Label record](#) 94  
[Legend in charts](#) 30  
[Legend record](#) 95  
[LegendException record](#) 96  
[Line record](#) 96  
[LineFormat record](#) 97  
[LinkedSelection record](#) 98  
[Localization](#) 16  
[LongRGB structure](#) 148

**M**

[MainWindow record](#) 99  
[MarkerFormat record](#) 99  
[MaxStatus record](#) 101  
[MsoDrawing record](#) 101  
[MsoDrawingGroup record](#) 101  
[MsoDrawingSelection record](#) 102

**N**

[NilChartNum structure](#) 149  
[Normative references](#) 14  
[Number record](#) 102

**O**

[Obj record](#) 102  
[ObjectLink record](#) 103  
[OfficeArtClientAnchorChart structure](#) 149  
[OfficeArtClientData structure](#) 150  
[OfficeArtClientTextbox structure](#) 151  
[OLE stream](#) 20  
[Orient record](#) 104  
 Overview  
   [byte ordering](#) 15  
   [documentation](#) 16  
[Overview \(synopsis\)](#) 15

**P**

[Palette record](#) 105  
[PicF record](#) 105  
[Pie record](#) 106  
[PieFormat record](#) 107  
[PlotArea record](#) 107  
[PlotGrowth record](#) 107  
[Pos record](#) 108  
[PositionMode structure](#) 151  
[Product behavior](#) 185

**R**

[Radar area record](#) 110  
[Radar record](#) 110  
 Record  
   [AIRuns](#) 49  
   [Area](#) 49  
   [AreaFormat](#) 49  
   [AttachedLabel](#) 55  
   [AxcExt](#) 56  
   [AxesUsed](#) 58  
   [Axis](#) 59  
   [AxisLine](#) 59  
   [AxisParent](#) 60  
   [Bar](#) 60  
   [Begin](#) 61  
   [Blank](#) 61  
   [BOF](#) 62  
   [BOFDatasheet](#) 64  
   [BopPop](#) 64  
   [BopPopCustom](#) 66  
   [BoundSheets](#) 67  
   [BRAI](#) 68  
   [CatLab](#) 69  
   [CatSerRange](#) 70  
   [Chart](#) 71  
   [Chart3d](#) 71  
   [Chart3DBarShape](#) 73  
   [ChartColors](#) 74  
   [ChartFormat](#) 74  
   [ChartFrtInfo](#) 75  
   [ClrtClient](#) 76  
   [CodePage](#) 77  
   [ColumnWidth](#) 77  
   [Continue](#) 77  
   [Country](#) 78  
   [CrtLine](#) 78  
   [CrtLink](#) 79  
   [Dat](#) 79  
   [DataFormat](#) 79  
   [DataLabExt](#) 80  
   [DataLabExtContents](#) 81  
   [Date1904](#) 82  
   [DefaultText](#) 82  
   [Dimensions](#) 83  
   [DropBar](#) 83  
   [End](#) 84  
   [EndBlock](#) 84  
   [EndObject](#) 85  
   [EOF](#) 85  
   [ExcludeColumns](#) 86  
   [ExcludeRows](#) 86  
   [Fbi](#) 86  
   [Fbi2](#) 87  
   [Font](#) 87  
   [FontX](#) 89  
   [Format](#) 89  
   [Frame](#) 90  
   [FrtFontList](#) 91  
   [FrtWrapper](#) 91  
   [GelFrame](#) 92  
   [IFmtRecord](#) 94  
   [Label](#) 94  
   [Legend](#) 95  
   [LegendException](#) 96

[Line](#) 96  
[LineFormat](#) 97  
[LinkedSelection](#) 98  
[MainWindow](#) 99  
[MarkerFormat](#) 99  
[MaxStatus](#) 101  
[MsoDrawing](#) 101  
[MsoDrawingGroup](#) 101  
[MsoDrawingSelection](#) 102  
[Number](#) 102  
[Obj](#) 102  
[ObjectLink](#) 103  
[Orient](#) 104  
[Palette](#) 105  
[PicF](#) 105  
[Pie](#) 106  
[PieFormat](#) 107  
[PlotArea](#) 107  
[PlotGrowth](#) 107  
[Pos](#) 108  
[Radar](#) 110  
[Radar area](#) 110  
[Scatter](#) 110  
[Scl](#) 111  
[Selection](#) 112  
[SerAuxErrBar](#) 112  
[SerAuxTrend](#) 113  
[SerFmt](#) 115  
[Series](#) 115  
[SeriesList](#) 116  
[SeriesText](#) 116  
[SerParent](#) 116  
[SerToCrt](#) 117  
[ShtProps](#) 117  
[StartBlock](#) 118  
[StartObject](#) 120  
[Surf](#) 122  
[Text](#) 122  
[Tick](#) 127  
[TxO](#) 130  
[Units](#) 132  
[ValueRange](#) 132  
[WinDoc](#) 135  
[Window1](#) 135  
[Window1\\_10](#) 136  
[Window2Graph](#) 136  
[YMult](#) 137  
Record enumeration  
  [by name](#) 42  
  [by number](#) 42  
[Record file structure](#) 18  
[References](#) 14  
  [informative](#) 15  
  [normative](#) 14  
[Relationship to protocols and other structures](#) 16  
[Run structure](#) 151

## S

[Scatter record](#) 110  
[Scl record](#) 111  
[Security - implementer considerations](#) 184  
[Selection record](#) 112  
[SerAuxErrBar record](#) 112  
[SerAuxTrend record](#) 113

[SerFmt record](#) 115  
[Series in charts](#) 31  
[Series record](#) 115  
[SeriesList](#) 116  
[SeriesText record](#) 116  
[SerParent record](#) 116  
[SerToCrt record](#) 117  
[ShortXLUnicodeString structure](#) 152  
[ShtProps record](#) 117  
[SPRC measurements in charts](#) 41  
[StartBlock record](#) 118  
[StartObject record](#) 120  
[Storages and streams file structure](#) 20  
  [chart sheet substream](#) 21  
  [component object stream](#) 20  
  [globals substream](#) 20  
  [OLE stream](#) 20  
  [workbook stream](#) 20  
[Stream file structure](#) 18  
Structure  
  [Boolean](#) 138  
  [CFrtId](#) 138  
  [ChartNumNillable](#) 139  
  [DateUnit](#) 139  
  [FontIndex](#) 139  
  [FontInfo](#) 140  
  [FormatRun](#) 140  
  [FrtFlags](#) 140  
  [FrtHeader](#) 141  
  [FrtHeaderOld](#) 141  
  [FtCr](#) 142  
  [FtCmo](#) 142  
  [FtGmo](#) 144  
  [Graph\\_Col](#) 144  
  [Graph\\_Rw](#) 144  
  [Icv](#) 144  
  [IcvChart](#) 147  
  [IcvFont](#) 148  
  [IFmt](#) 148  
  [LongRGB](#) 148  
  [NilChartNum](#) 149  
  [OfficeArtClientAnchorChart](#) 149  
  [OfficeArtClientData](#) 150  
  [OfficeArtClientTextbox](#) 151  
  [PositionMode](#) 151  
  [Run](#) 151  
  [ShortXLUnicodeString](#) 152  
  [TxOLastRun](#) 152  
  [TxORuns](#) 153  
  [XLUnicodeString](#) 153  
  [XLUnicodeStringMin2](#) 154  
  [XLUnicodeStringNoCch](#) 154  
  [Xnum](#) 154  
[Substream file structure](#) 18  
[Surf record](#) 122

## T

[Text record](#) 122  
[Tick record](#) 127  
[Tracking changes](#) 190  
[Trendlines in charts](#) 39  
[TxO record](#) 130  
[TxOLastRun structure](#) 152  
[TxORuns structure](#) 153

## U

[Units record](#) 132

## V

[ValueRange record](#) 132

[Vendor-extensible fields](#) 17

[Versioning](#) 16

## W

[WinDoc record](#) 135

[Window1 record](#) 135

[Window1\\_10 record](#) 136

[Window2Graph record](#) 136

[Workbook stream](#) 20

## X

[XLUnicodeString structure](#) 153

[XLUnicodeStringMin2 structure](#) 154

[XLUnicodeStringNoCch structure](#) 154

[Xnum structure](#) 154

## Y

[YMult record](#) 137

Preliminary