

[MS-RPL]: Report Page Layout (RPL) Binary Stream Format Structure Specification

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft's Open Specification Promise (available here: <http://www.microsoft.com/interop/osp>) or the Community Promise (available here: <http://www.microsoft.com/interop/cp/default.mspx>). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplq@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.

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

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

Preliminary Documentation. This Open Specification is preliminary documentation for this technology. Since the documentation may change between this preliminary version and the final version, there are risks in relying on preliminary documentation. To the extent that you incur

additional development obligations or any other costs as a result of relying on this preliminary documentation, you do so at your own risk.

Revision Summary

| Date | Revision History | Revision Class | Comments |
|------------|------------------|----------------|--------------------------------|
| 08/07/2009 | 0.1 | Major | First release. |
| 11/06/2009 | 0.2 | Minor | Updated the technical content. |

Contents

| | |
|---|-----------|
| 1 Introduction | 9 |
| 1.1 Glossary..... | 9 |
| 1.2 References..... | 12 |
| 1.2.1 Normative References | 12 |
| 1.2.2 Informative References | 12 |
| 1.3 Structure Overview (Synopsis) | 12 |
| 1.3.1 Byte Ordering..... | 13 |
| 1.3.2 Organization of This Documentation | 13 |
| 1.4 Relationship to Protocols and Other Structures | 13 |
| 1.5 Applicability Statement..... | 14 |
| 1.6 Versioning and Localization | 14 |
| 1.7 Vendor-Extensible Fields | 14 |
| 1.8 Vendor-Unsupported Fields..... | 14 |
| 2 Structures | 15 |
| 2.1 Stream Structure | 15 |
| 2.1.1 RPL Stream..... | 15 |
| 2.1.2 Record..... | 15 |
| 2.1.3 Simple Data Type Structures | 15 |
| 2.1.3.1 Writing Strings to an RPL Stream | 16 |
| 2.1.3.2 Writing the Length Value of the String to an RPL Stream | 16 |
| 2.1.3.3 Reading Strings from an RPL Stream | 16 |
| 2.2 Report Structures..... | 16 |
| 2.2.1 Report..... | 16 |
| 2.2.2 Version | 18 |
| 2.2.3 ReportProperties | 18 |
| 2.2.4 OffsetsArrayElement | 19 |
| 2.2.5 PageContent..... | 20 |
| 2.2.5.1 PageContent (RPL 10.6)..... | 20 |
| 2.2.5.2 PageContent (RPL 10.4 and 10.5)..... | 21 |
| 2.2.5.3 PageContent (RPL 10.3) | 22 |
| 2.2.6 Page | 23 |
| 2.2.7 PageProperties | 24 |
| 2.2.8 PageLayout..... | 25 |
| 2.2.9 Section | 26 |
| 2.2.10 SimpleSection | 26 |
| 2.2.11 MixedSection..... | 28 |
| 2.2.12 SectionProperties | 29 |
| 2.2.13 BodyAreaElement..... | 30 |
| 2.2.13.1 BodyAreaElement (RPL 10.4)..... | 30 |
| 2.2.13.2 BodyAreaElement (RPL 10.3)..... | 31 |
| 2.2.14 PageHeaderElement | 32 |
| 2.2.15 PageFooterElement | 33 |
| 2.2.16 BodyElement..... | 34 |
| 2.2.17 ElementProperties | 36 |
| 2.2.18 SharedElementProperties | 36 |
| 2.2.19 UseSharedElementProperties | 36 |
| 2.2.20 InlineSharedElementProperties..... | 37 |
| 2.2.21 NonSharedElementProperties | 38 |
| 2.2.22 Style | 40 |

| | | |
|--------|---------------------------------|----|
| 2.2.23 | SharedStyleProperties | 42 |
| 2.2.24 | NonSharedStyleProperties | 42 |
| 2.2.25 | ActionInfo | 43 |
| 2.2.26 | ActionInfoContent | 44 |
| 2.2.27 | Action | 44 |
| 2.2.28 | ActionImageMapAreas | 45 |
| 2.2.29 | ActionInfoWithMaps | 46 |
| 2.2.30 | DynamicImageData | 47 |
| 2.2.31 | ReportItem | 47 |
| 2.2.32 | Line | 48 |
| 2.2.33 | Image | 49 |
| 2.2.34 | ImageDataProperties | 50 |
| 2.2.35 | UseSharedImageDataProperties | 50 |
| 2.2.36 | InlineSharedImageDataProperties | 51 |
| 2.2.37 | NonSharedImageDataProperties | 52 |
| 2.2.38 | ImageData | 53 |
| 2.2.39 | ImageMapAreas | 53 |
| 2.2.40 | ImageMapArea | 54 |
| 2.2.41 | Chart | 55 |
| 2.2.42 | GaugePanel | 56 |
| 2.2.43 | Map | 57 |
| 2.2.44 | Rectangle | 58 |
| 2.2.45 | SubReport | 60 |
| 2.2.46 | RichTextBox | 61 |
| 2.2.47 | ParagraphContent | 63 |
| 2.2.48 | TextRun | 63 |
| 2.2.49 | Paragraph | 65 |
| 2.2.50 | RichTextBoxStructure | 66 |
| 2.2.51 | Tablix | 67 |
| 2.2.52 | TablixContent | 68 |
| 2.2.53 | TablixStructure | 68 |
| 2.2.54 | TablixMeasurements | 69 |
| 2.2.55 | ColumnsWidths | 70 |
| 2.2.56 | ColumnInfo | 71 |
| 2.2.57 | Row Heights | 71 |
| 2.2.58 | Row Info | 72 |
| 2.2.59 | TablixRow | 73 |
| 2.2.60 | TablixRow Cell | 73 |
| 2.2.61 | TablixCorner | 74 |
| 2.2.62 | TablixColumnHeader | 75 |
| 2.2.63 | TablixRow Header | 76 |
| 2.2.64 | TablixBodyRow Cells | 77 |
| 2.2.65 | TablixBodyRow | 77 |
| 2.2.66 | TablixBodyCell | 78 |
| 2.2.67 | TablixRow MembersDef | 79 |
| 2.2.68 | TablixColMembersDef | 79 |
| 2.2.69 | TablixMemberDef | 80 |
| 2.2.70 | Measurements | 81 |
| 2.2.71 | Measurement | 82 |
| 2.2.72 | ReportElementEnd | 83 |
| 2.3 | Properties | 84 |
| 2.3.1 | ID | 85 |
| 2.3.2 | ColumnCount | 85 |

| | | |
|--------|---------------------------------|-----|
| 2.3.3 | ColumnSpacing | 86 |
| 2.3.4 | UniqueName | 87 |
| 2.3.5 | Name | 87 |
| 2.3.6 | Label | 88 |
| 2.3.7 | Bookmark | 89 |
| 2.3.8 | ToolTip | 89 |
| 2.3.9 | ToggleItem | 90 |
| 2.3.10 | Description | 90 |
| 2.3.11 | Location | 91 |
| 2.3.12 | Language | 91 |
| 2.3.13 | ExecutionTime | 92 |
| 2.3.14 | Author | 93 |
| 2.3.15 | AutoRefresh | 93 |
| 2.3.16 | ReportName | 94 |
| 2.3.17 | PageHeight | 94 |
| 2.3.18 | PageWidth | 95 |
| 2.3.19 | MarginTop | 96 |
| 2.3.20 | MarginLeft | 96 |
| 2.3.21 | MarginRight | 97 |
| 2.3.22 | MarginBottom | 97 |
| 2.3.23 | Columns | 98 |
| 2.3.24 | PageName (RPL 10.6) | 99 |
| 2.3.25 | Slant | 99 |
| 2.3.26 | CanGrow | 100 |
| 2.3.27 | CanShrink | 100 |
| 2.3.28 | Value | 101 |
| 2.3.29 | ToggleState | 102 |
| 2.3.30 | CanSort | 102 |
| 2.3.31 | SortState | 103 |
| 2.3.32 | Formula | 104 |
| 2.3.33 | IsToggleParent | 104 |
| 2.3.34 | TypeCode | 105 |
| 2.3.35 | OriginalValue | 105 |
| 2.3.36 | IsSimple | 106 |
| 2.3.37 | ContentOffset | 107 |
| 2.3.38 | StreamName | 107 |
| 2.3.39 | Sizing | 108 |
| 2.3.40 | LinkToChild | 109 |
| 2.3.41 | PrintOnFirstPage | 109 |
| 2.3.42 | PrintBetweenSections (RPL 10.4) | 110 |
| 2.3.43 | FormattedValueExpressionBased | 111 |
| 2.3.44 | ProcessedWithError | 111 |
| 2.3.45 | ImageMIMEType | 112 |
| 2.3.46 | ImageName | 113 |
| 2.3.47 | Width | 113 |
| 2.3.48 | Height | 114 |
| 2.3.49 | HorizontalResolution | 114 |
| 2.3.50 | VerticalResolution | 115 |
| 2.3.51 | RawFormat | 116 |
| 2.3.52 | Hyperlink | 116 |
| 2.3.53 | BookmarkLink | 117 |
| 2.3.54 | DrillthroughId | 117 |
| 2.3.55 | DrillthroughUrl | 118 |

| | | |
|---------|---------------------------|-----|
| 2.3.56 | BorderColor..... | 118 |
| 2.3.57 | BorderColorLeft..... | 119 |
| 2.3.58 | BorderColorRight..... | 120 |
| 2.3.59 | BorderColorTop..... | 120 |
| 2.3.60 | BorderColorBottom..... | 121 |
| 2.3.61 | BorderStyle..... | 122 |
| 2.3.62 | BorderStyleLeft..... | 122 |
| 2.3.63 | BorderStyleRight..... | 123 |
| 2.3.64 | BorderStyleTop..... | 123 |
| 2.3.65 | BorderStyleBottom..... | 124 |
| 2.3.66 | BorderWidth..... | 124 |
| 2.3.67 | BorderWidthLeft..... | 125 |
| 2.3.68 | BorderWidthRight..... | 126 |
| 2.3.69 | BorderWidthTop..... | 126 |
| 2.3.70 | BorderWidthBottom..... | 127 |
| 2.3.71 | PaddingLeft..... | 128 |
| 2.3.72 | PaddingRight..... | 128 |
| 2.3.73 | PaddingTop..... | 129 |
| 2.3.74 | PaddingBottom..... | 129 |
| 2.3.75 | FontStyle..... | 130 |
| 2.3.76 | FontFamily..... | 131 |
| 2.3.77 | FontSize..... | 131 |
| 2.3.78 | FontWeight..... | 132 |
| 2.3.79 | Format..... | 132 |
| 2.3.80 | TextDecoration..... | 133 |
| 2.3.81 | TextAlign..... | 133 |
| 2.3.82 | VerticalAlign..... | 134 |
| 2.3.83 | Color..... | 134 |
| 2.3.84 | LineHeight..... | 135 |
| 2.3.85 | Direction..... | 136 |
| 2.3.86 | WritingMode..... | 136 |
| 2.3.87 | UnicodeBiDi..... | 137 |
| 2.3.88 | BackgroundImage..... | 137 |
| 2.3.89 | BackgroundColor..... | 138 |
| 2.3.90 | BackgroundRepeat..... | 138 |
| 2.3.91 | NumeralLanguage..... | 139 |
| 2.3.92 | NumeralVariant..... | 140 |
| 2.3.93 | Calendar..... | 140 |
| 2.3.94 | ColumnHeaderRows..... | 141 |
| 2.3.95 | RowHeaderColumns..... | 142 |
| 2.3.96 | ColsBeforeRowHeader..... | 142 |
| 2.3.97 | LayoutDirection..... | 143 |
| 2.3.98 | DefinitionPath..... | 143 |
| 2.3.99 | Level..... | 144 |
| 2.3.100 | MemberCellIndex..... | 144 |
| 2.3.101 | CellItemOffset..... | 145 |
| 2.3.102 | ColSpan..... | 146 |
| 2.3.103 | RowSpan..... | 146 |
| 2.3.104 | DefIndex..... | 147 |
| 2.3.105 | ColumnIndex..... | 148 |
| 2.3.106 | RowIndex..... | 149 |
| 2.3.107 | GroupLabel..... | 149 |
| 2.3.108 | RecursiveToggleLevel..... | 150 |

| | | |
|----------|-------------------------------------|------------|
| 2.3.109 | ListStyle..... | 151 |
| 2.3.110 | ListLevel..... | 151 |
| 2.3.111 | ParagraphNumber..... | 152 |
| 2.3.112 | RightIndent..... | 152 |
| 2.3.113 | LeftIndent..... | 153 |
| 2.3.114 | HangingIndent..... | 154 |
| 2.3.115 | SpaceBefore..... | 154 |
| 2.3.116 | SpaceAfter..... | 155 |
| 2.3.117 | FirstLine..... | 155 |
| 2.3.118 | Markup..... | 156 |
| 2.3.119 | ContentTop..... | 156 |
| 2.3.120 | ContentLeft..... | 157 |
| 2.3.121 | ContentWidth..... | 158 |
| 2.3.122 | ContentHeight..... | 159 |
| 2.3.123 | State..... | 159 |
| 2.3.124 | CellItemState..... | 160 |
| 2.3.125 | MemberDefState..... | 161 |
| 2.4 | Enumerations..... | 161 |
| 2.4.1 | SortOptions..... | 161 |
| 2.4.2 | Sizings..... | 161 |
| 2.4.3 | ShapeType..... | 162 |
| 2.4.4 | ImageRawFormat..... | 162 |
| 2.4.5 | FontStyles..... | 162 |
| 2.4.6 | FontWeights..... | 163 |
| 2.4.7 | TextDecorations..... | 163 |
| 2.4.8 | TextAlignments..... | 163 |
| 2.4.9 | VerticalAlignments..... | 164 |
| 2.4.10 | Directions..... | 164 |
| 2.4.11 | WritingModes..... | 164 |
| 2.4.12 | UnicodeBiDiTypes..... | 164 |
| 2.4.13 | Calendars..... | 165 |
| 2.4.14 | BorderStyles..... | 165 |
| 2.4.15 | BackgroundRepeatTypes..... | 166 |
| 2.4.16 | ListStyles..... | 166 |
| 2.4.17 | MarkupStyles..... | 166 |
| 2.4.18 | TypeCode..... | 167 |
| 2.4.19 | StateValues..... | 167 |
| 2.4.20 | TablixMemberStateValues..... | 168 |
| 2.4.21 | TablixMemberDefStateValues..... | 169 |
| 2.4.22 | RPLSize..... | 170 |
| 3 | Structure Examples..... | 172 |
| 3.1 | Line..... | 172 |
| 3.2 | Image..... | 173 |
| 3.3 | Chart..... | 174 |
| 3.4 | GaugePanel..... | 176 |
| 3.5 | SubReport..... | 178 |
| 3.6 | RichTextBox..... | 180 |
| 3.7 | Tablix..... | 186 |
| 3.8 | Report..... | 193 |
| 3.9 | Report 2008..... | 209 |
| 4 | Security Considerations..... | 225 |

| | |
|---|------------|
| 5 Appendix A: Product Behavior | 226 |
| 6 Change Tracking | 231 |
| 7 Index..... | 233 |

Preliminary

1 Introduction

The Report Page Layout (RPL) Stream Format Structure Specification specifies the Report Page Layout stream format. The RPL stream format is an internal binary format used by Microsoft® SQL Server® Reporting Services when communicating with viewer controls, to offload some of the rendering work from the server to the client viewer control. By using RPL, implementers can create custom report designers that will create RPL, as well as custom report renderers that process and display RPL to display reports.

1.1 Glossary

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

Augmented Backus-Naur Form (ABNF)
big-endian
little-endian
stream
object
Unicode

The following terms are specific to this document:

absolute URL: A URL that specifies the exact [Location](#) of a [Report](#), and that takes the form protocol://hostname/other_information (ex: http://www.msn.com/myReport.rdl).

action: An interactivity event such as a hyperlink, bookmark link, or drillthrough link that is associated with an item in a **report**.

ancestor: A record that contains or is contained in another record.

body area: The area containing the detail and grouped data within a **page**.

body: The main component of the **report** within which the details and contents are included.

bookmark: An anchor that is used in a **report** to assist navigation, typically through the use of hyperlinks. Bookmarks are specified by the [Bookmark](#) element.

calendar: A property that allows the user to format dates based on the [Calendar](#) used.

chart: Displays data or the relationships between sets of data in a visual form. A chart instance is specified by using the [Chart](#) element.

column header: The header area of a column in a data region layout. The tablix column header is specified by the [TablixColumnHeader](#) element.

column member: An ordered list of members within a column **hierarchy** of a data region. **Tablix** column members are specified by the [TablixColMembersDef](#) element.

document map: A navigation pane that is generated at run time from labels specified in a report definition.

drillthrough report: A report that can be executed and viewed after clicking a **report item** that contains an action with a drillthrough link.

element: The data that makes up an array of structures. Each element inherits the attributes from the parent structure.

enumeration: A fixed set of integer values or **String** constants that are used to specify the value of certain properties.

footer area: The area at the bottom of each **page** of a **report** containing the layout of **report items**. The layout is specified by the [PageFooterElement](#) record.

gauge panel: The encompassing outer element that contains one or more radial gauges and linear gauges. Use the gauge panel to define properties that apply to all gauges in a gauge panel. A gauge panel instance is specified by using the [GaugePanel](#) element.

grandparent record: A type of **ancestor** record that contains child records. The record specifies the parent's **element** of the current record.

header area: The area at the top of each **page** of a **report** containing the layout of **report items**. The layout is specified by the [PageHeaderElement](#) record.

hierarchy: A logical tree structure that organizes a record such that each member has one parent member and zero or more child members.

horizontal resolution: The number of elements, dots, or columns from left to right in a fixed area—typically one inch. The resolution is specified by a [HorizontalResolution](#) element with a floating point value.

image: A rectangular grid of pixels that has a definite height and a definite width and is specified by the [Image](#) element.

image map: An image map is comprised of one or more areas that specify the coordinates of a clickable region of an **image**.

image map area: A set of coordinates in an **image map** making a polygon that can be assigned an **action**. The image map area is specified by the [ImageMapArea](#) element.

language: A text value that contains a language and culture code. The value of the language is specified by the [Language](#) element.

MIME type: A method that is used by protocol clients to associate files of a certain type (such as specific image formats) with applications that can open or access files of that type.

page: Specifies information pertaining to [PageContent](#). The details provided for a page include properties, header, and footer. A page is specified by the [Page](#) element.

parent record: An **ancestor** record. The record specifies the containing **element** that is a record.

physical-page oriented renderer: A renderer that generates page breaks based on physical measurements. The renderer can set properties to precisely control how the pages will break in reports.

raw format: Defines the contents of a raw image that has all the information needed to create an **image**. The format of an image is specified by the [RawFormat](#) element.

recursive toggle: Enables the user to toggle the display of the hidden data in a tablix **column header** and **row header**. The recursive toggling for the **tablix** is specified by the [RecursiveToggleLevel](#) element.

report item: An object that exists on a report layout. A report item is defined by the [ReportItem](#) element.

rich text box: A text box with advanced features, specified by the [RichTextBox](#) element.

report: A collection of information comprising data, layout, and properties. A report is specified by the Report element.

row header: Contains any **report item** that is set to be displayed in the row **header area** of a **tablix**. The row header area contains the headers of the visible row fields of a tablix data region. The row header of a tablix is specified by the [TablixRowHeader](#) element.

row member: A member in a data region layout that represents a row, including an optional header, an optional grouping of data, and an optional inner **hierarchy**.

RPLIF: An RPL Stamp, which is a **String** value that specifies the type of the stream. The current value of the RPL Stamp is "RPLIF" for an RPL stream.

section: Specifies the layout and structure information of a **report**. A **report section** is comprised of a **body**, a header, and a footer. A section is specified by the [Section](#) element.

sort state: Specifies how the data in a **report item** is organized. The sort state is defined by the [SortState](#) property.

stream name: A **String** value that uniquely identifies the name of a stream for an item. The stream name is specified by the [StreamName](#) element.

tablix: A data region that contains rows and columns resembling a table or matrix, possibly sharing characteristics of both. A tablix instance is specified by the [Tablix](#) element.

tablix body area: The area containing the detail and grouped data within a **tablix**.

tablix member: An ordered list of members for a tablix row **hierarchy** or a tablix column hierarchy in a **tablix** data region. The tablix member can contain a nested tablix members list.

text decoration: A visual ornamentation added to the text, specified by the [TextDecoration](#) element.

text run: The value and formatting of a contiguous span of text. A text run is specified by the [TextRun](#) element.

toggle state: Defines the value that specifies the current state of toggle of a control, which is either expanded or collapsed. The toggle state is defined by the [ToggleState](#) property.

token: The byte that specifies the start of a record.

version: A number that identifies the specific release and version of RPL. The version is specified by using major, minor, and build fields.

vertical resolution: The number of rows, dots, or lines from top to bottom on a fixed area—typically one inch. The vertical resolution is specified by a [VerticalResolution](#) element with a floating point value.

zIndex: Specifies the drawing order of an item within the containing item. Items are drawn in order according to the **zIndex element** for the item. Items with lower indices are drawn first, appearing behind items with higher indices.

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

1.2 References

1.2.1 Normative References

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

[ISO-639] International Organization for Standardization, "Codes for the Representation of Names of Languages", ISO 639, <http://www.loc.gov/standards/iso639-2/>

[ISO-3166-1] International Organization for Standardization, "[ISO 3166-1 decoding table](http://www.iso.org/iso/iso-3166-1_decoding_table)", ISO3166, http://www.iso.org/iso/iso-3166-1_decoding_table.

[MS-GLOS] Microsoft Corporation, "[Windows Protocols Master Glossary](#)", March 2007.

[MS-RDL200801] Microsoft Corporation, "Report Definition Language (RDL) 2008", <http://schemas.microsoft.com/sqlserver/reporting/2008/01/reportdefinition>

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.ietf.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.ietf.org/rfc/rfc5234.txt>

[RFC2781] Hoffman, P., and Yergeau, F., "UTF-16, an encoding of ISO 10646", RFC 2781, February 2000, <http://www.ietf.org/rfc/rfc2781.txt>

1.2.2 Informative References

[MS-RDL200501] Microsoft Corporation, "Report Definition Schema", <http://schemas.microsoft.com/sqlserver/reporting/2005/01/reportdefinition>

[MS-RDLS05SPEC] Microsoft Corporation, "Report Definition Language Specification", November 2005, <http://download.microsoft.com/download/c/2/0/c2091a26-d7bf-4464-8535-dbc31fb45d3c/rdlNov05.pdf>

[MS-RDLS08SPEC] Microsoft Corporation, "Report Definition Language Specification", July 2008, http://download.microsoft.com/download/6/5/7/6575f1c8-4607-48d2-941d-c69622e11c32/RDL_spec_08.pdf

[MS-RS2008WP] Microsoft Corporation, "[Microsoft SQL Server 2008 Reporting Services White Paper](#)," August 2007, <http://download.microsoft.com/download/a/c/d/acd8e043-d69b-4f09-bc9e-4168b65aaa71/RSinSQL2008.doc>.

1.3 Structure Overview (Synopsis)

This document specifies the Report Page Layout (RPL) format, a binary stream format that is used to represent metadata for a **page** of a **report**. The [stream](#) is described by a collection of nested records, each of which contains information about the type of the record, and where applicable, type-specific fields that further specify the record. Type-specific fields contain information that further specify the RPL format. Specifications of the fields for a given record type are in the [Record](#) section.

For a detailed overview of specific file architecture and content, see [Stream Structure](#).

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 stream format [enumerations](#), [objects](#), and records in RPL are stored in 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 |

1.3.2 Organization of This Documentation

Section [2](#) of this document is arranged with overviews of higher-level concepts, which are followed by more detailed concepts. Section 2.1, in particular, covers higher-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 an RPL stream file.

Section [2.2](#) through section [2.2.72](#) specifies the details of each of the structures that can be included in an RPL stream.

Section [2.3](#) through section [2.3.125](#) specifies the details of each of the properties that can be included in an RPL stream.

Section [2.4](#) through section [2.4.22](#) specifies the details of each of the enumerations that can be included in an RPL stream.

Section [3](#) provides specific examples intended to illustrate the concepts, records, and structures of this stream format.

Appendix [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 relevant items in the appendix.

1.4 Relationship to Protocols and Other Structures

The RPL binary stream format uses UTF-16 encoding of [Unicode](#) characters as specified in [\[RFC2781\]](#).

In addition to other file formats, RPL can be used as a protocol for reports that are created in Report Definition Language (RDL). The schema of the Report Definition Language is defined in [\[MS-RDL200801\]](#).

For more information about RDL, see [\[MS-RDL200501\]](#), [\[MS-RDLS05SPEC\]](#), and [\[MS-RDLS08SPEC\]](#).

The two protocols that carry RPL are HTTP and the Report Server ReportExecution Web service [\[MS-RWSRE2005\]](#)

1.5 Applicability Statement

This document specifies the format for an RPL stream.

The RPL stream is applicable for use as a protocol to allow custom report designers to create rendering instructions in RPL, and to allow custom report renderers to process and display these rendering instructions to display reports

1.6 Versioning and Localization

This document covers versioning issues in the following areas:

Structure Versions: There are four **versions** of RPL. The structures and properties in this document apply to all versions; if a particular structure or properties applies only to a certain version, the version is specified.

RPL version 10.3 is shipped with Microsoft® SQL Server® 2008. RPL versions 10.4, 10.5, and 10.6 are shipped with Microsoft® SQL Server® 2009.

Localization: There are no localization-dependent structures in the RPL stream format

1.7 Vendor-Extensible Fields

None.

1.8 Vendor-Unsupported Fields

If the RPL stream does not have well formatted grammar, ReportViewer controls will not read the stream. ReportViewer controls are the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Microsoft Visual Studio® 2010. The controls do not identify unknown spaces in structures. The controls do not keep track of fields that are not present in the grammar.

2 Structures

This section specifies all RPL structures, including report structures, report properties, and enumerations, in an RPL stream and their associated grammar. Every structure includes the following:

- A definition of the structure.
- The ABNF grammar for the structure.
- A bit diagram of the structure.
- Definitions of all fields contained within the structure, including the sizes of all related structures. Fields following a variable length field can be unaligned; that is, they can start on an even or an odd byte boundary from the beginning of the structure.

2.1 Stream Structure

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

The stream structure contains a series of [records](#). Each record contains zero or more structured fields that contain the report layout.

2.1.1 RPL Stream

This document specifies the structures and format required to create an RPL stream. The RPL stream **MUST** have only one [Report](#) record and the stream **MUST** be written as a series of binary records that preserve the report **hierarchy**.

The left and top properties of an item are relative to a parent. **Report item** measurements are saved at the parent level. The stream has a built-in offset-based hierarchy that allows a reader to jump from a parent to a child and read the rendering runtime sizes without having to read all the properties. All offsets are stored on 8 bytes. To optimize space, some structures are shared between report items so they will be written only once. All numeric values are declared as **Int32** unless otherwise specified. **Datetime** values are written as long values. **Boolean** values are defined as 0 for false, 1 for true.

This document uses Augmented Backus-Naur Form (ABNF) as specified in [\[RFC5234\]](#) to specify the sequence for RPL records.

This document uses information from Report Definition Language (RDL) as described in [\[MS-RDLS08SPEC\]](#) and [\[RS2008WP\]](#) to describe the RPL records.

2.1.2 Record

A record is the basic building block used to store information about a report. Each record is a variable-length sequence of bytes. A record consists of two components: a record type and the record data that is specific to that record type.

The record type is one byte that specifies what type of information is specified by the record and how the structure of the record data pertaining to the record is ordered and structured. The record value is dependent on the type of data that is specific to that record.

2.1.3 Simple Data Type Structures

The following table specifies the data types in an RPL stream.

| Data type | Description format |
|----------------|---|
| Int64 | Represents a 64-bit (8-byte) signed integer. |
| Int32 | Represents a 32-bit (4-byte) signed integer. |
| Float | Represents a 32-bit (4-byte) single-precision floating point value. |
| Boolean | Represents an 8-bit (1-byte) logical Boolean type value that MUST be true (1) or false (0). |
| Long | Represents a 64-bit (8-byte) signed integer. |
| String | All String values within the protocol MUST be UNICODE UTF-16. By default, all String values start with an integer that defines the length of the String . String values are represented in the protocol as an array of bytes; the number of bytes MUST be equal to the number of characters in the String multiplied by two. |

2.1.3.1 Writing Strings to an RPL Stream

Strings are written as a length-prefixed **String** to the RPL stream using UTF-16 encoding as specified in [\[RFC2781\]](#). After the length of the string is written, the current position of the stream is advanced in accordance with the encoding and the specific characters being written to the stream.

2.1.3.2 Writing the Length Value of the String to an RPL Stream

The length of a **String** is written to the RPL stream seven bits at a time, starting with the seven least-significant bits. The high bit of a byte indicates whether there are more bytes to be written after the current one.

If a value fits in seven bits, the value takes only one byte of space. If the value does not fit in seven bits, the high bit is set on the first byte and written out. The value is then shifted by seven bits and the next byte is written. This process is repeated until the entire integer has been written.

2.1.3.3 Reading Strings from an RPL Stream

Strings MUST be prefixed with an integer that defines the length of the **String** and is encoded using the rules specified in section [2.1.3.2](#).

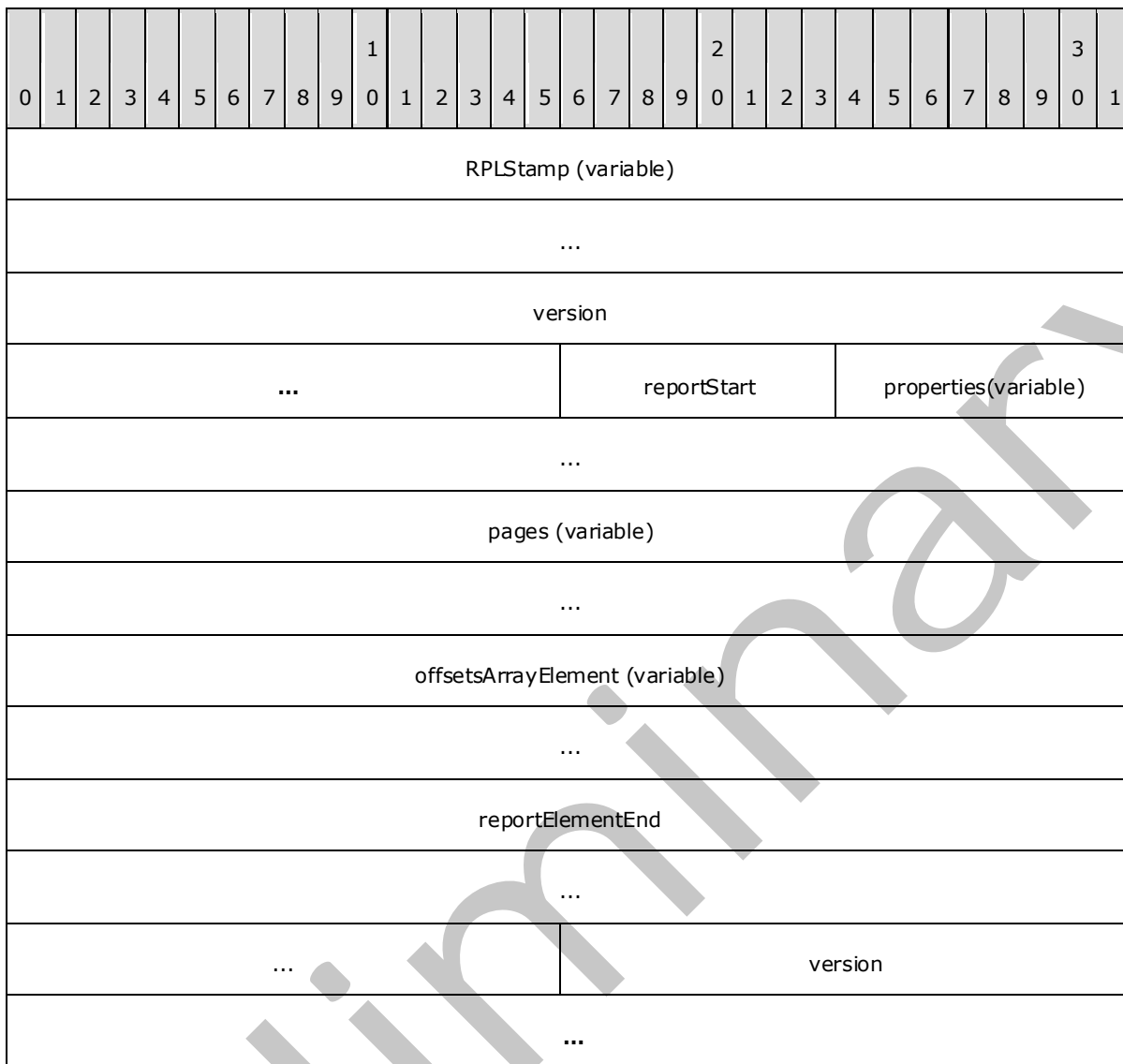
2.2 Report Structures

This section defines all the report structures along with their definitions, grammar, and a bit diagram. The structures include the definitions and sizes of their related structures and **elements**.

2.2.1 Report

The **Report** record is the root record of the [RPL Stream](#) and specifies the content of the stream. There MUST be one and only one record of this type in the stream.

```
Report = RPLStamp Version reportStart ReportProperties *PageContent OffsetsArrayElement
ReportElementEnd Version
```

RPLStamp (variable): A **String** that specifies the type of the stream. The value of the **RPLStamp** field MUST be "**RPLIF**".<1>

version (6 bytes): A [Version](#) record that specifies the version of RPL used to generate the RPL stream. The value of this **Version** record MUST be the same as the subsequent **Version** record in the stream.

reportStart (1 byte): A byte that specifies the start of the **Report** record content. The value of the **reportStart** field MUST be 0x00.

properties (variable): A [ReportProperties](#) record that specifies the properties applicable to the **Report** record.

pages (variable): An array of [PageContent](#) records. The number of elements in the array MUST be equal to the value of the **count** field of the [OffsetArrayElement](#) record specified by the **offsetsArrayElement** field.

| | |
|-----------|--|
| delimiter | |
|-----------|--|

reportPropertiesStart (1 byte): A byte that specifies the start of the **ReportProperties** record. The value of the **reportPropertiesStart** field MUST be 0x02.

itemProperties (variable): A collection of structures. The order of the structures in the collection is arbitrary. The collection can be empty. The structures contained in the **itemProperties** field MUST be one of the following.

| Property |
|-------------------------------|
| Description |
| Location |
| Language |
| ExecutionTime |
| Author |
| AutoRefresh |
| ReportName |

delimiter (1 byte): A byte that specifies the end of the **ReportProperties** record. The value of the **delimiter** field MUST be 0xFF.

2.2.4 OffsetsArrayElement

The **OffsetsArrayElement** record specifies information about the position of the **reportStart** field of the [Report](#) record in the RPL stream and the position of the **reportElementEnd** field of the [PageContent](#) records that are part of a **Report** record.

OffsetsArrayElement = offsetsArrayElementStart offset count offsetsPageContent

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------------|---|---|---|---|---|---|---|---|---|-------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| offsetsArrayElementStart | | | | | | | | | | offset | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | offsetsPageContent (variable) | | | | | | | | | | | | | | | | | | | | | | | |

...

offsetsArrayElementStart (1 byte): A byte that specifies the start of the **OffsetsArrayElement** record. The value of the **offsetsArrayElementStart** field MUST be 0x12.

offset (8 bytes): A signed integer that specifies the position of the **reportStart** field of the **parent record** of type **Report** in the RPL stream.

count (4 bytes): A signed integer that MUST be equal to the number of **PageContent** records in the parent **Report** record.

offsetsPageContent (variable): An array of 8-byte signed integers. The number of elements in the array MUST be equal to the value of the **count** field. Every element of the array MUST specify the position of the corresponding **reportElementEnd** field of the **PageContent** record in the RPL stream. The order of the elements in the array MUST match the order of the **PageContent** records in the parent **Report** record.

2.2.5 PageContent

The **PageContent** record specifies the content of a page.

2.2.5.1 PageContent (RPL 10.6)

Applies to RPL version 10.6 and later

The **PageContent** record specifies the content of a page of a report.

PageContent = pageContentStart PageLayout *Section Measurements [PageLayout] ReportElementEnd

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------------|---|---|---|---|---|---|---|---|---|----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| pageContentStart | | | | | | | | | | pageLayoutStart (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| sections (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| measurements (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| pageLayoutEnd (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| | |
|------------------|--|
| ... | |
| reportElementEnd | |
| ... | |
| ... | |

pageContentStart (1 byte): A byte that specifies the start of the **PageContent** record. The value of the **pageContentStart** field MUST be 0x13.

pageLayoutStart (variable): A [PageLayout](#) record that specifies properties that are applicable to the **PageContent** record.

sections (variable): An array of [Section](#) records. The number of records in the array MUST be equal to the value of the **count** field of the [Measurements](#) record specified by the **measurements** field in the **PageContent** record.

measurements (variable): A record of type **Measurements** that specifies the size, location, and state properties for every element of the sections array. The order of elements in the **measurementsContent** field array of the **Measurements** record MUST match the order of **Section** records in the sections array.

pageLayoutEnd (variable): A **PageLayout** record that specifies properties that are applicable to the **PageContent** record. If the same property is specified as part of the **pageLayoutStart** and **pageLayoutEnd** fields, the value that is specified as **pageLayoutEnd** MUST be used.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **PageContent** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **measurements** field of the **PageContent** record in the RPL stream.

2.2.5.2 PageContent (RPL 10.4 and 10.5)

Applies to RPL version 10.4 and 10.5

The **PageContent** record specifies the content of a page of a report.

PageContent = pageContentStart PageLayout *Section Measurements ReportElementEnd

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|-----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| pageContentStart | | | | | | | | | | pageLayout (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| |
|-------------------------|
| sections (variable) |
| ... |
| measurements (variable) |
| ... |
| reportElementEnd |
| ... |
| ... |

pageContentStart (1 byte): A byte that specifies the start of the **PageContent** record. The value of the **pageContentStart** field MUST be 0x13.

pageLayout (variable): A [PageLayout](#) record that specifies properties applicable to the **PageContent** record.

sections (variable): An array of [Section](#) records. The number of records in the array MUST be equal to the value of the **count** field of the [Measurements](#) record specified by the **measurements** field in the **PageContent** record.

measurements (variable): A record of type **measurements** that specifies the size, location, and state properties for every element of the **sections** array. The order of elements in the **measurementsContent** field array of the **measurements** record MUST match the order of **Section** records in the **sections** array.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **PageContent** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **measurements** field of the **PageContent** record in the RPL stream.

2.2.5.3 PageContent (RPL 10.3)

Applies to RPL version 10.3

The **PageContent** record specifies the content of a page of a report.

PageContent = pageContentStart BodyAreaElement Page Measurements ReportElementEnd

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|---------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| pageContentStart | | | | | | | | | | bodyArea (variable) | | | | | | | | | | | | | | | | | | | | | | | | |

| |
|-------------------------|
| ... |
| page (variable) |
| ... |
| measurements (variable) |
| ... |
| reportElementEnd |
| ... |
| ... |

pageContentStart (1 byte): A byte that specifies the start of the **PageContent** record. The value of the **pageContentStart** field MUST be 0x13.

bodyArea (variable): A [BodyAreaElement](#) record that specifies the content of the **body area** of the **PageContent** record.

page (variable): A [Page](#) record that specifies the properties, header, and footer of a page.

measurements (variable): A record of type [Measurements](#) that specifies the size, location, and state properties for the **bodyArea** record and for the **pageHeaderElement** and **pageFooterElement** fields of the **Page** record.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **PageContent** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **measurements** field of the **PageContent** record in the RPL stream.

2.2.6 Page

Applies to RPL version 10.3

The **Page** record specifies the properties, header, and footer that apply to a [PageContent](#) record.

Page = pageStart PageProperties [PageHeaderElement] [PageFooterElement] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| pageStart | | | | | | | | | | pageProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | | |

| |
|------------------------------|
| ... |
| pageHeaderElement (variable) |
| ... |
| pageFooterElement (variable) |
| ... |
| delimiter |

pageStart (1 byte): A byte that specifies the start of the **Page** record. The value of the **pageStart** field MUST be 0x01.

pageProperties (variable): A [PageProperties](#) record that specifies the properties applicable to the **Page** record.

pageHeaderElement (variable): A [PageHeaderElement](#) record that specifies the content of the **header area** of a **Page** record.

pageFooterElement (variable): A [PageFooterElement](#) record that specifies the content of the **footer area** of a **Page** record.

delimiter (1 byte): A byte that specifies the end of the **Page** record. The value of the **delimiter** field MUST be 0xFF.

2.2.7 PageProperties

Applies to RPL version 10.3

The **PageProperties** record specifies the properties applicable to a [Page](#) record.

PageProperties = pagePropertiesStart [itemProperties] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| pagePropertiesStart | | | | | | | | | | itemProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

pagePropertiesStart (1 byte): A byte that specifies the start of the **PageProperties** record. The value of the **pagePropertiesStart** field MUST be 0x03.

itemProperties (variable): A collection of structures. The order of the structures in the collection is arbitrary. The collection can be empty. The structures contained in the **itemProperties** field MUST be one of the following.

| Property |
|-------------------------------|
| ID |
| UniqueName |
| PageHeight |
| PageWidth |
| MarginTop |
| MarginLeft |
| MarginBottom |
| MarginRight |
| Columns |
| ColumnSpacing |
| Style |

delimiter (1 byte): A byte that specifies the end of the **PageProperties** record. The value of the **delimiter** field MUST be 0xFF.

2.2.8 PageLayout

Applies to RPL version 10.4 and later

The **PageLayout** record specifies the properties applicable to a [PageContent](#) record.

PageLayout = pagePropertiesStart [itemProperties] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| pagePropertiesStart | | | | | | | | | | itemProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| |
|------------------------------|
| ... |
| bodyAreaElement (variable) |
| ... |
| pageFooterElement (variable) |
| ... |
| pageHeaderElement (variable) |
| ... |
| measurements (variable) |
| ... |
| reportElementEnd |
| ... |
| ... |

sectionStart (1 byte): A byte that specifies the start of the **SimpleSection** record. The value of the **sectionStart** field MUST be 0x15.

sectionProperties (variable): A [SectionProperties](#) record that specifies the properties applicable to the **SimpleSection** record.

bodyAreaElement (variable): A [BodyAreaElement](#) record that specifies the content of the body area of the **SimpleSection** record.

pageFooterElement (variable): A [PageFooterElement](#) record that specifies the content of the footer area of the **SimpleSection** record.

pageHeaderElement (variable): A [PageHeaderElement](#) record that specifies the content of the header area of the **SimpleSection** record.

measurements (variable): A [Measurements](#) record that specifies the size, location, and state properties for **BodyAreaElement**, **PageFooterElement**, and **PageHeaderElement** records. The order of the elements in the **measurementsContent** field array of the **Measurements** record MUST match the order of the **BodyAreaElement**, **PageFooterElement**, and **PageHeaderElement** records in the **SimpleSection** record.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **SimpleSection** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **measurements** field of the **SimpleSection** record in the RPL stream.

2.2.11 MixedSection

Applies to RPL version 10.4 and later

The **MixedSection** record specifies the structure and layout information for more than one report section.

```
MixedSection = sectionStart SectionProperties BodyAreaElement [PageFooterElement] Section
[PageHeaderElement] Measurements ReportElementEnd
```

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
|------------------------------|---|---|---|---|---|---|---|---|---|------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| sectionStart | | | | | | | | | | sectionProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| bodyAreaElement (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| pageFooterElement (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| section (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| pageHeaderElement (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| measurements (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| | |
|-----|--|
| ... | |
| ... | |

sectionStart (1 byte): A byte that specifies the start of the **MixedSection** record. The value of the **sectionStart** field MUST be 0x15.

sectionProperties (variable): A [SectionProperties](#) record that specifies the properties applicable to the **MixedSection** record.

bodyAreaElement (variable): A [BodyAreaElement](#) record that specifies the content of the body area of the **MixedSection** record.

pageFooterElement (variable): A [PageFooterElement](#) record that specifies the content of the footer area of the **MixedSection** record.

section: A [Section](#) record that specifies the content of the subsequent section in the sections field array of the [PageContent \(RPL 10.4\)](#) record.

pageHeaderElement (variable): A [PageHeaderElement](#) record that specifies the content of the header area of the **MixedSection** record.

measurements (variable): A [Measurements](#) record that specifies the size, location, and state properties for **BodyAreaElement**, **PageFooterElement**, and **PageHeaderElement** records. The order of the elements in the **measurementsContent** field array of the **Measurements** record MUST match the order of the **bodyAreaElement**, **pageFooterElement**, and **pageHeaderElement** records in the **MixedSection** record.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **MixedSection** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **measurements** field of the **MixedSection** record in the RPL stream.

2.2.12 SectionProperties

Applies to RPL version 10.4 and 10.5 (or later)

The **SectionProperties** record specifies the properties applicable to a [Section](#) record.

SectionProperties = sectionPropertiesStart [itemProperties] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 3 | | | | | |
| sectionPropertiesStart | | | | | | | | | | itemProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

sectionPropertiesStart (1 byte): A byte that specifies the start of the **SectionProperties** record. The value of the **sectionPropertiesStart** field MUST be 0x16.

itemProperties (variable): A collection of structures. The order of the structures in the collection is arbitrary. The collection can be empty. The structures contained in the **itemProperties** field MUST be one of the following.

| Property |
|-------------------------------|
| ID |
| ColumnCount |
| ColumnSpacing |

delimiter (1 byte): A byte that specifies the end of the **SectionProperties** record. The value of this field MUST be 0xFF.

2.2.13 BodyAreaElement

The **BodyAreaElement** record specifies the content of the body area of a [Section](#) or, in RPL 10.3, a [PageContent](#) record.

2.2.13.1 BodyAreaElement (RPL 10.4)

Applies to RPL version 10.4 and later

The **BodyAreaElement** record specifies the content of the body area of a section.

BodyAreaElement = columns *BodyElement Measurements ReportElementEnd

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
|-------------------------|---|---|---|---|---|---|---|---|---|---------------------|---|---|---|---|---|---|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| columns | | | | | | | | | | bodyArea (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| measurements (variable) | | | | | | | | | | | | | | | | | ... | | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

columns (1 byte): A byte that specifies the start of the **BodyAreaElement** record. The value of the **columns** field MUST be 0x14.

bodyArea (variable): An array of [BodyElement](#) records. The number of elements in the **bodyArea** array MUST be equal to the value of the **count** field of the [Measurements](#) record specified in the **measurements** field of the **BodyAreaElement** record. The number of elements MUST NOT be greater than the **ColumnCount** property specified in the [SectionProperties](#) record. <3>

measurements (variable): A **Measurements** record that specifies the size, location, and state properties for every element in the **bodyArea** array. The order of the elements in the **measurementsContent** field array of the **Measurements** record MUST match the order of the **BodyElement** records in the **bodyArea** array.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **BodyAreaElement** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **measurements** field of the **BodyAreaElement** record in the RPL stream.

2.2.13.2 BodyAreaElement (RPL 10.3)

Applies to RPL version 10.3

The **BodyAreaElement** record specifies the content of the body of a page.

BodyAreaElement = columns *BodyElement Measurements ReportElementEnd

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------------|---|---|---|---|---|---|---|---|---|---------------------|---|---|---|---|---|---|---|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| columns | | | | | | | | | | bodyArea (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| measurements (variable) | | | | | | | | | | | | | | | | | | ... | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

columns (1 byte): A byte that specifies the start of the **BodyAreaElement** record. The value of the **columns** field MUST be 0x14.

bodyArea (variable): An array of [BodyElement](#) records. The number of elements in the array MUST be equal to the value of the **count** field of the [Measurements](#) record specified by the **measurements** field. The number of elements MUST NOT be greater than the [Columns](#) property specified in the [Page](#) record. <4>

measurements (variable): A **Measurements** record that specifies the size, location, and state properties for every element of the **bodyArea** array. The order of the elements in the **measurementsContent** field array of the **Measurements** record MUST match the order of [BodyElement](#) records in the **bodyArea** array.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **BodyAreaElement** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **BodyAreaElement** record's **measurements** field in the RPL stream.

2.2.14 PageHeaderElement

The **PageHeaderElement** record specifies the content of the header area of a [Section](#) or [Page](#) record.

PageHeaderElement = pageHeader [ElementProperties] *ReportItems Measurements ReportElementEnd

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
|-------------------------|---|---|---|---|---|---|---|---|---|------------------------------|---|---|---|---|---|---|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| pageHeader | | | | | | | | | | elementProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportItems (variable) | | | | | | | | | | | | | | | | | ... | | | | | | | | | | | | | | | | |
| measurements (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | ... | | | | | | | | | | | | | | | | |

pageHeader (1 byte): A byte that specifies the start of the **PageHeaderElement** record. The value of the **pageHeader** field MUST be 0x04.

elementProperties (variable): A record of type [ElementProperties](#) that specifies a collection of properties applicable to the **PageHeaderElement** record. The order of the structures in the collection is arbitrary. The collection can be empty. The structures contained in the **elementProperties** field MUST be one of the following.

| Property |
|---|
| ID |
| Style |
| PrintOnFirstPage |
| PrintBetweenSections (RPL 10.4) |
| UniqueName |

reportItems (variable): An array of [ReportItem](#) records. The number of elements in the array MUST be equal to the value of the **count** field of the [Measurements](#) record specified by the **measurements** field.

measurements (variable): A **Measurements** record that specifies the size, location, and state properties for every element of the **reportItems** field array. The order of elements in the **measurementsContent** field array of the **Measurements** record MUST match the order of **ReportItem** records in the **reportItems** field array.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **PageHeaderElement** record. The **offset** of the **ReportElementEnd** record MUST specify the position of the **measurements** field in the RPL stream.

2.2.15 PageFooterElement

The **PageFooterElement** record specifies the content of the footer area of a [Section](#) or [Page](#) record.

```
PageFooterElement = pageFooter [ElementProperties] *ReportItems Measurements ReportElementEnd
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------------|---|---|---|---|---|---|---|---|---|------------------------------|---|---|---|---|---|---|---|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| pageFooter | | | | | | | | | | elementProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportItems (variable) | | | | | | | | | | | | | | | | | | ... | | | | | | | | | | | | | | | |
| measurements (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| | |
|------------------|--|
| ... | |
| reportElementEnd | |
| ... | |
| ... | |

pageFooter (1 byte): A byte that specifies the start of the **PageFooterElement** record. The value of the **pageFooter** field MUST be 0x05.

elementProperties (variable): A record of type [ElementProperties](#) that specifies a collection of properties applicable to the **PageFooterElement** record. The order of the structures in the collection is arbitrary. The structures contained in the **elementProperties** field MUST be one of the following.

| Property |
|---|
| ID |
| Style |
| PrintOnFirstPage |
| PrintBetweenSections (RPL 10.4) |
| UniqueName |

reportItems (variable): An array of [ReportItem](#) records. The number of elements in the array MUST be equal to the value of the **count** field of the [Measurements](#) record specified by the **measurements** field.

measurements (variable): A **Measurements** record that specifies the size, location, and state properties for every element of the **reportItems** field array. The order of elements in the array MUST match the order of **ReportItem** records in the **reportItems** field array.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **PageFooterElement** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **measurements** field of the **PageFooterElement** in the RPL stream.

2.2.16 BodyElement

The **BodyElement** record specifies the content of an element of the **bodyArea** field array of the **BodyAreaElement** record.

```
BodyElement = body [ElementProperties] *ReportItem ReportElementEnd
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------------|---|---|---|---|---|---|---|---|---|------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| body | | | | | | | | | | elementProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportItems (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| measurements (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

body (1 byte): A byte that specifies the start of a **BodyElement** record. The value of the **body** field MUST be 0x06.

elementProperties (variable): A record of type [ElementProperties](#) that specifies a collection of properties applicable to the **BodyElement** record. The order of the structures in the collection is arbitrary. The structures contained in the **elementProperties** field MUST be one of the following.

| Property |
|----------------------------|
| ID |
| Style |
| UniqueName |

reportItems (variable): An array of [ReportItem](#) records. The number of elements in the array MUST be equal to the value of the **count** field of the [Measurements](#) record specified by the **measurements** field.

measurements (variable): A **Measurements** record that specifies the size, location, and state properties for every element of the **reportItems** field array. The order of elements in the **measurementsContent** field array of the **Measurements** record MUST match the order of the **ReportItem** records in the **reportItems** field array.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **BodyElement** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **measurements** field of the **BodyElement** record in the RPL stream.

2.2.17 ElementProperties

The **ElementProperties** record specifies the properties applicable to the [BodyElement](#), [PageHeaderElement](#), [PageFooterElement](#), [ReportItem](#), [TextRun](#), and [Paragraph](#) records.

```
ElementProperties = elementPropertiesStart SharedElementProperties
[NonSharedElementProperties] delimiter
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------------------------|---|---|---|---|---|---|---|---|---|------------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| elementPropertiesStart | | | | | | | | | | sharedElementProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| nonSharedElementProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

elementPropertiesStart (1 byte): A byte that specifies the start of the **ElementProperties** record. The value of the **elementPropertiesStart** field MUST be 0x0F.

sharedElementProperties (variable): A [SharedElementProperties](#) record.

nonSharedElementProperties (variable): A [NonSharedElementProperties](#) record.

delimiter (1 byte): A byte that specifies the end of the **ElementProperties** record. The value of the **delimiter** field MUST be 0xFF.

2.2.18 SharedElementProperties

The **SharedElementProperties** record specifies the properties that can be shared by multiple [ElementProperties](#) records.

```
SharedElementProperties = UseSharedElementProperties / InlineSharedElementProperties
```

2.2.19 UseSharedElementProperties

The **UseSharedElementProperties** record specifies the location of shared properties applicable to a **grandparent record** in the RPL stream.

UseSharedElementProperties = useShared offset

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|---|---|---|---|---|---|---|---|---|--------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| useShared | | | | | | | | | | offset | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

useShared (1 byte): A byte that specifies the start of the **UseSharedElementProperties** record. The value of the **useShared** field MUST be 0x02.

offset (8 bytes): A signed integer that MUST specify the start of an [ElementProperties](#) record in the RPL stream, which MUST have an [InlineSharedElementProperties](#) record.

2.2.20 InlineSharedElementProperties

The **InlineSharedElementProperties** record specifies the properties that can be shared by multiple [ElementProperties](#) records.

InlineSharedElementProperties = shared [itemProperties] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| shared | | | | | | | | | | itemProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

shared (1 byte): A byte that specifies the start of the **InlineSharedElementProperties** record. The value of the **shared** field MUST be 0x00.

itemProperties (variable): A collection of structures. The order of the structures in the collection is arbitrary. If the [OriginalValue](#) property and the [TypeCode](#) property are in the collection, the **OriginalValue** property MUST appear in the collection after the **TypeCode** property. The structures contained in the **itemProperties** field MUST be one of the following.

| Property |
|---|
| ID |
| Name |
| Label |
| Bookmark |
| ToolTip |
| Style |
| ToggleItem |
| Slant |
| CanGrow |
| CanSort |
| CanShrink |
| Value |
| Formula |
| IsToggleParent |
| TypeCode |
| OriginalValue |
| IsSimple |
| Sizing |
| LinkToChild |
| PrintOnFirstPage |
| PrintBetweenSections (RPL 10.4) |
| FormattedValueExpressionBased |
| ReportName |

delimiter (1 byte): A byte that specifies the end of the **InlineSharedElementProperties** record. The value of the **delimiter** field MUST be 0xFF.

2.2.21 NonSharedElementProperties

The **NonSharedElementProperties** record specifies the properties that are specific to its parent [ElementProperties](#) record. These properties are not shared by multiple **ElementProperties** records.

```
NonSharedElementProperties = nonShared [itemProperties] delimiter
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| nonShared | | | | | | | | | | itemProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

nonShared (1 byte): A byte that specifies the start of the **NonSharedElementProperties** record. The value of the **nonShared** field MUST be 0x01.

itemProperties (variable): A collection of structures. The order of the structures in the collection is arbitrary. If the [OriginalValue](#) property and the [TypeCode](#) property are in the collection, the **OriginalValue** property MUST appear in the array after the **TypeCode** property. The structures contained in the **itemProperties** field MUST be one of the following.

| Property |
|-------------------------------------|
| UniqueName |
| Label |
| Bookmark |
| ToolTip |
| Style |
| ActionInfo |
| Value |
| ToggleState |
| SortState |
| IsToggleParent |
| TypeCode |
| OriginalValue |
| ProcessedWithError |
| ContentOffset |
| ContentHeight |
| ActionImageMapAreas |
| DynamicImageData |
| StreamName |

| Property |
|--------------------------|
| Image |
| Language |

delimiter (1 byte): A byte that specifies the end of the **NonSharedElementProperties** record. The value of the **delimiter** field MUST be 0xFF.

2.2.22 Style

The **Style** record specifies the properties associated with the style for the parent record. The parent record for this property MUST be an [InlineSharedElementProperties](#), [NonSharedElementProperties](#), [PageProperties](#), or [PageLayout](#) record.

Style = styleStart [SharedStyleProperties] [NonSharedStyleProperties] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------------------------|---|---|---|---|---|---|---|---|---|----------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| styleStart | | | | | | | | | | sharedStyleProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| nonSharedStyleProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

styleStart (1 byte): A byte that specifies the start of the **Style** record. The value of the **styleStart** field MUST be 0x06.

sharedStyleProperties (variable): A [SharedStyleProperties](#) record. This field MUST NOT be present if the parent record is **NonSharedElementProperties**.<5>

nonSharedStyleProperties (variable): A [NonSharedStyleProperties](#) record. This field MUST NOT be present if the parent record is **InlineSharedElementProperties**.<6>

The structures specified by the **sharedStyleProperties** and the **nonSharedStyleProperties** fields MUST be on the following list.

| Property |
|---------------------------------|
| BorderColor |
| BorderColorLeft |

| Property |
|-----------------------------------|
| BorderColorRight |
| BorderColorTop |
| BorderColorBottom |
| BorderStyle |
| BorderStyleLeft |
| BorderStyleRight |
| BorderStyleTop |
| BorderStyleBottom |
| BorderWidth |
| BorderWidthLeft |
| BorderWidthRight |
| BorderWidthTop |
| BorderWidthBottom |
| PaddingLeft |
| PaddingRight |
| PaddingTop |
| PaddingBottom |
| FontStyle |
| FontFamily |
| FontSize |
| FontWeight |
| Format |
| TextDecoration |
| TextAlign |
| VerticalAlign |
| Color |
| LineHeight |
| Direction |
| WritingMode |
| UnicodeBiDi |

| |
|----------------------------------|
| Property |
| Language |
| BackgroundImage |
| BackgroundColor |
| BackgroundRepeat |
| NumeralLanguage |
| NumeralVariant |
| Calendar |

delimiter (1 byte): A byte that specifies the end of the **Style** record. The value of the **delimiter** field MUST be 0xFF.

2.2.23 SharedStyleProperties

The **SharedStyleProperties** record specifies the style properties that can be shared by multiple records.

```
SharedStyleProperties = shared [styleProperties] delimiter
```



shared (1 byte): A byte that specifies the start of the **SharedStyleProperties** record. The value of the **shared** field MUST be 0x00.

styleProperties (variable): A collection of structures that specifies style properties that are shared by multiple records. The order of the structures in the collection is arbitrary. The collection can be empty.

delimiter (1 byte): A byte that specifies the end of the **SharedStyleProperties** record. The value of the **delimiter** field MUST be 0xFF.

2.2.24 NonSharedStyleProperties

The **NonSharedStyleProperties** record specifies the style properties that are specific to a record.

NonSharedStyleProperties = nonShared [styleProperties] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|---|---|---|---|---|---|---|---|---|----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| nonShared | | | | | | | | | | styleProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

nonshared (1 byte): A byte that specifies the start of the **NonSharedStyleProperties** record. The value of the **nonshared** field MUST be 0x01.

styleProperties (variable): A collection of structures that specifies style properties that are specific to a record. The order of the structures in the collection is arbitrary. The collection can be empty.

delimiter (1 byte): A byte that specifies the end of the **NonSharedStyleProperties** record. The value of the **delimiter** field MUST be 0xFF.

2.2.25 ActionInfo

The **ActionInfo** property specifies a structure that defines the actions for an [Image](#), [RichTextBox](#), or [TextRun](#) record. If this property is specified, it MUST be part of one of the following records:

- [Image.ElementProperties.NonSharedElementProperties](#)
- **RichTextBox.ElementProperties.NonSharedElementProperties**
- **TextRun.ElementProperties.NonSharedElementProperties**

ActionInfo = actionInfoStart [ActionInfoContent] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|---|---|---|---|---|---|---|---|---|------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| actionInfoStart | | | | | | | | | | actionInfoContent (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

actionInfoStart (1 byte): A byte that specifies the start of the **ActionInfo** record. The value of the **actionInfoStart** field MUST be one of the following.

| Value | Parent record |
|-------|--|
| 0x07 | ElementProperties.NonSharedElementProperties for Image ElementProperties.NonSharedElementProperties for RichTextBox |
| 0x0B | ElementProperties.NonSharedElementProperties for TextRun |

actionInfoContent (variable): An [ActionInfoContent](#) record.

delimiter (1 byte): A byte that specifies the end of the **ActionInfo** record. The value of the **delimiter** field MUST be 0xFF.

2.2.26 ActionInfoContent

The **ActionInfoContent** record specifies an array of [Action](#) records.

```
ActionInfoContent = actionsStart count actions
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------|---|---|---|---|---|---|---|---|---|--------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| actionsStart | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | actions (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

actionsStart (1 byte): A byte that specifies the start of the **ActionInfoContent** record. The value of the **actionsStart** field MUST be 0x02.

count (4 bytes): A signed integer that specifies the number of **Action** record elements in the array.

actions (variable): An array of **Action** records. The number of elements in the array MUST be equal to the value of the **count** field.

2.2.27 Action

The **Action** record specifies the properties of an **action**.

```
Action = actionStart actionInfoProperties delimiter
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|---|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| actionStart | | | | | | | | | | actionInfoProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

actionStart (1 byte): A byte that specifies the start of the **Action** record. The value of the **actionStart** field MUST be 0x03.

actionInfoProperties (variable): A collection of structures that specifies the properties applicable to the **Action** record. The order of the structures in the collection is arbitrary. The **actionInfoProperties** collection can be empty. If the **actionInfoProperties** collection is specified, it MUST contain one and only one of the following structures:

| Property |
|---------------------------------|
| Label |
| HyperLink |
| BookmarkLink |
| DrillthroughId |
| DrillthroughUrl |

delimiter (1 byte): A byte that specifies the end of the **Action** record. The value of the **delimiter** field MUST be 0xFF.

2.2.28 ActionImageMapAreas

The **ActionImageMapAreas** property specifies a structure that defines a set of actions associated with areas of a [Chart](#), [Image](#), [Map](#), or [GaugePanel](#) record. If the **ActionImageMapAreas** is specified, it MUST be part of one of the following records:

- [Chart.ElementProperties.NonSharedElementProperties](#)
- **Image.ElementProperties.NonSharedElementProperties**
- **Map.ElementProperties.NonSharedElementProperties**
- **GaugePanel.ElementProperties.NonSharedElementProperties**

```
ActionImageMapAreas = actionImageMapAreasStart count *ActionInfoWithMaps
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------------|---|---|---|---|---|---|---|---|---|-------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| actionImageMapAreasStart | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | actionInfoWithMaps (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

actionImageMapAreasStart (1 byte): A byte that specifies the start of the **ActionImageMapAreas** record. The value of the **actionImageMapAreasStart** field MUST be 0x26.

count (4 bytes): A signed integer that specifies the number of [ActionInfoWithMaps](#) record elements in the array.

actionInfoWithMaps (variable): An array of **ActionInfoWithMaps** records. The number of elements in the array MUST be equal to the value of the **count** field.

2.2.29 ActionInfoWithMaps

The **ActionInfoWithMaps** record specifies an array of image areas for a [Chart](#), [Image](#), [Map](#), or [GaugePanel](#) record and the actions associated with these image areas.

ActionInfoWithMaps = actionInfoStart [ActionInfoContent] [ImageMapAreas] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|---|---|---|---|---|---|---|---|---|------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| actionInfoStart | | | | | | | | | | actionInfoContent (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | imageMapAreas (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

actionInfoStart (1 byte): A byte that specifies the start of the **ActionInfoWithMaps** record. The value of the **actionInfoStart** field MUST be 0x07.

actionInfoContent (variable): An [ActionInfoContent](#) record.

imageMapAreas (variable): An [ImageMapAreas](#) record.

delimiter (1 byte): A byte that specifies the end of the **ActionInfoWithMaps** record. The value of the **delimiter** field MUST be 0xFF.

2.2.30 DynamicImageData

The **DynamicImageData** property specifies a structure that defines the dynamically generated **images** for a [Chart](#), [Map](#), or [GaugePanel](#) record. If the **DynamicImageData** property is specified, it MUST be part of one of the following records:

- [Chart.ElementProperties.NonSharedElementProperties](#)
- **Map.ElementProperties.NonSharedElementProperties**
- **GaugePanel.ElementProperties.NonSharedElementProperties**

```
DynamicImageData = dynamicImageDataStart count dynamicImageContent
```

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
|-----------------------|---|---|---|---|---|---|---|---|---|--------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| dynamicImageDataStart | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | dynamicImageContent (variable) | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

dynamicImageDataStart (1 byte): A byte that specifies the start of the **DynamicImageData** record. The value of the **dynamicImageDataStart** field MUST be 0x27.

count (4 bytes): A signed integer that specifies the number of bytes that comprise the image content.

dynamicImageContent (variable): An array of bytes that comprise the a **Chart**, **Map**, or **GaugePanel**. The number of bytes in the array MUST be equal to the value of the **count** field.

2.2.31 ReportItem

The **ReportItem** record specifies the structure types used to define the content of [BodyElement](#), [PageHeaderElement](#), and [PageFooterElement](#) records. Report items include [Line](#), [Image](#), [Chart](#), [Map](#), [GaugePanel](#), [Rectangle](#), [SubReport](#), [RichTextBox](#), and [Tablix](#). **Chart**, **Map**, **GaugePanel**, **SubReport**, or **Tablix** records MUST NOT be part of a **PageHeaderElement** or **PageFooterElement** record. <7>

```
ReportItem = Line / Image / Chart / GaugePanel / Map / Rectangle / SubReport / RichTextBox / Tablix
```

2.2.32 Line

The **Line** record specifies the properties applicable to a **Line** in a [Report](#).

```
Line = lineToken lineProperties ReportElementEnd
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| lineToken | | | | | | | | | | lineProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

lineToken (1 byte): A byte that specifies the start of the **Line** record. The value of the **lineToken** field MUST be 0x08.

lineProperties (variable): A record of type [ElementProperties](#) that specifies the properties applicable to the **Line** record. The structures contained in this field MUST be one of the following.

| Property |
|----------------------------|
| UniqueName |
| Slant |
| ID |
| Name |
| Tooltip |
| Bookmark |
| Label |
| ToggleItem |
| Style |

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **Line** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **lineToken** field of the **Line** record in the RPL stream.

2.2.33 Image

The **Image** record specifies the properties applicable to an **Image** within a [Report](#).

Image = imageStart imageProperties ReportElementEnd

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| imageStart | | | | | | | | | | imageProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

imageStart (1 byte): A byte that specifies the start of the **Image** record. The value of the **imageStart** field MUST be 0x09.

imageProperties (variable): A record of type [ElementProperties](#) that specifies the properties applicable to the **Image** record. The structures contained in the **imageProperties** field MUST be one of the following.

| Property |
|-------------------------------------|
| UniqueName |
| ID |
| Name |
| Tooltip |
| Bookmark |
| Label |
| ToggleItem |
| Style |
| Sizing |
| ActionInfo |
| ActionImageMapAreas |

| |
|-------------------------------------|
| Property |
| ImageDataProperties |

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **Image** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **imageToken** field of the **Image** record in the RPL stream.

2.2.34 ImageDataProperties

The **ImageDataProperties** record specifies a collection of properties applicable to the [Image](#) record and the [BackgroundImage](#) style property. The parent record for the **ImageDataProperties** record MUST be one of the following:

- [Image.ElementProperties.NonSharedElementProperties](#)
- [Style.SharedStyleProperties](#)
- [Style.NonSharedStyleProperties](#)

`ImageDataProperties = imageStart (UseSharedImageDataProperties / InlineSharedImageDataProperties / NonSharedImageDataProperties) delimiter`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------|---|---|---|---|---|---|---|---|---|--------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| imageStart | | | | | | | | | | imageDataProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

imageStart (1 byte): A byte that specifies the start of the **ImageDataProperties** record. The value of the **imageStart** field MUST be 0x2A.

imageDataProperties (variable): A [UseSharedImageDataProperties](#), [InlineSharedImageDataProperties](#), or [NonSharedImageDataProperties](#) record. When the parent record is **Style.SharedStyleProperties**, the **imageDataProperties** MUST be **UseSharedImageDataProperties** or **InlineSharedImageDataProperties**. The **imageDataProperties** MUST be **NonSharedImageDataProperties** when the parent record is **Style.NonSharedStyleProperties**.

delimiter (1 byte): A byte that specifies the end of **ImageDataProperties** record. The value of the **delimiter** field MUST be 0xFF.

2.2.35 UseSharedImageDataProperties

The **UseSharedImageDataProperties** record specifies the location of shared properties applicable to an [Image](#) record or [BackgroundImage](#) style property.

UseSharedImageDataProperties = useShared offset

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|---|---|---|---|---|---|---|---|---|--------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| useShared | | | | | | | | | | offset | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

useShared (1 byte): A byte that specifies the start of the **UseSharedImageDataProperties** record. The value of the **useShared** field MUST be 0x02.

offset (8 bytes): A signed integer that specifies the start of the [ImageDataProperties](#) record in the RPL stream; the **ImageDataProperties** record MUST have the [InlineSharedImageDataProperties](#) record.

2.2.36 InlineSharedImageDataProperties

The **InlineSharedImageDataProperties** record specifies the properties that can be shared by multiple [ImageDataProperties](#) records.

InlineSharedImageDataProperties = shared [itemProperties] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| shared | | | | | | | | | | itemProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

shared (1 byte): A byte that specifies the start of the **InlineSharedImageDataProperties** record. The value of the **shared** field MUST be 0x00.

itemProperties (variable): A collection of structures. The order of the elements in the collection is arbitrary. The structures contained in the **itemProperties** field MUST be one of the following.

| Property |
|-------------------------------|
| ImageMimeType |

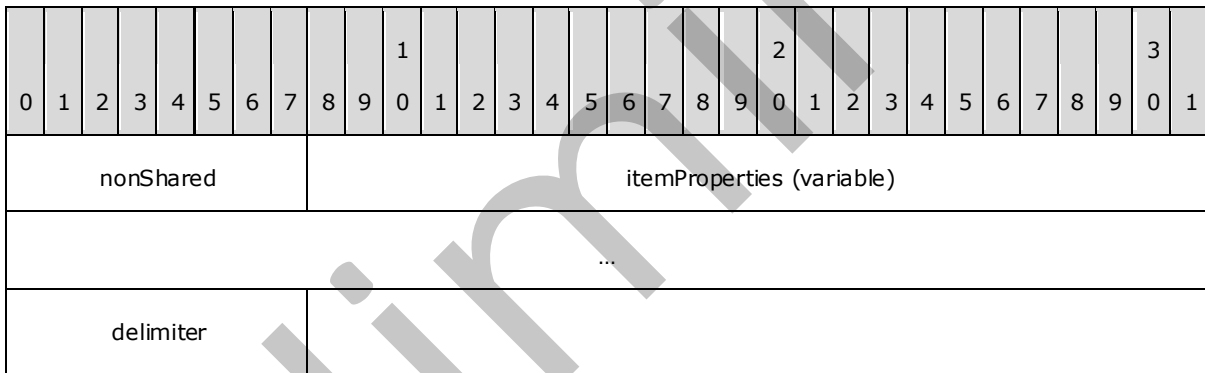
| Property |
|--------------------------------------|
| ImageName |
| ImageData |
| Width |
| Height |
| HorizontalResolution |
| VerticalResolution |
| RawFormat |

delimiter (1 byte): A byte that specifies the end of the **InlineSharedImageDataProperties** record. The value of the **delimiter** field MUST be 0xFF.

2.2.37 NonSharedImageDataProperties

The **NonSharedImageDataProperties** record specifies the properties that are specific to a record. These properties are not shared by multiple [ImageDataProperties](#) records.

`NonSharedImageDataProperties = nonShared [itemProperties] delimiter`



nonShared (1 byte): A byte that specifies the start of the **NonSharedImageDataProperties** record. The value of the **nonShared** field MUST be 0x01.

itemProperties (variable): A collection of structures. The order of the elements in the collection is arbitrary. The structures contained in the **itemProperties** field MUST be one of the following.

| Property |
|-------------------------------|
| ImageMimeType |
| ImageName |
| ImageData |
| Width |

| |
|--------------------------------------|
| Property |
| Height |
| HorizontalResolution |
| VerticalResolution |
| RawFormat |

delimiter (1 byte): A byte that specifies the end of the **NonSharedImageDataProperties** record. The value of the **delimiter** field MUST be 0xFF.

2.2.38 ImageData

The **ImageData** property specifies a structure that defines the content for an image. If this property is specified, it MUST be part of one of the following records:

- [ImageDataProperties.InlineSharedImageDataProperties](#)
- [ImageDataProperties.NonSharedImageDataProperties](#)

ImageData = imageDataStart count imageDataContent

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| imageDataStart | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | imageDataContents (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

imageDataStart (1 byte): A byte that specifies the start of the **ImageData** record. The value of the **imageDataStart** field MUST be 0x02.

count (4 bytes): A signed integer that specifies the number of bytes that comprise the image content.

imageDataContents (variable): An array of bytes that comprise the image. The number of bytes in the array MUST be equal to the value of the **count** field.

2.2.39 ImageMapAreas

The **ImageMapAreas** record specifies an array of **image map areas**.

ImageMapAreas = imageMapAreasStart count imageMapAreas

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------|---|---|---|---|---|---|---|---|---|--------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| imageMapAreasStart | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | imageMapAreas (variable) | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

imageMapAreasStart (1 byte): A byte that specifies the start of the **ImageMapAreas** record. The value of the **imageMapAreasStart** field MUST be 0x0A.

count (4 bytes): A signed integer that specifies the number of [ImageMapArea](#) record elements in the array.

imageMapAreas (variable): An array of **ImageMapArea** records. The number of elements in the array MUST be equal to the value of the **count** field.

2.2.40 ImageMapArea

The **ImageMapArea** record specifies an area in an **image map**.

ImageMapArea = shapeType count coordinates (ToolTip / delimiter)

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|------------------------|---|---|---|---|---|-----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| shapeType | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | coordinates (variable) | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| toolTip (variable) | | | | | | | | | | | | | | | | ... | | | | | | | | | | | | | | | |
| delimiter (optional) | | | | | | | | | | ... | | | | | | | | | | | | | | | | | | | | | |

shapeType (1 byte): A byte that specifies the shape of the area. The value of the **shapeType** field MUST be specified by the [ShapeType](#) enumeration.

count (4 bytes): A signed integer that specifies the number of coordinates record elements in the array.

coordinates (variable): An array of floating point values that specifies the coordinates of the shape, as a percentage of the image size. The number of elements in the array MUST be equal to the value of the **count** field. <8>

For a polygon shape, the **coordinates** array MUST have an even length and consist of X, Y coordinate pairs. Every even array element is a horizontal X coordinate and every odd array element is a vertical Y coordinate.

For a circle shape, there are exactly three elements in the **coordinates** array; these elements MUST occur in this order: the first two are the X and Y coordinates of the center of the circle, and the third element is the radius of the circle.

For a rectangle shape, there are four elements in the **coordinates** array; these elements MUST occur in this order: left, top, width, and height.

toolTip (variable): A [ToolTip](#) property that specifies the textual label for the image map area. This is an optional field that MUST be present only if the **delimiter** field is not specified.

delimiter (1 byte): A byte that specifies the end of an **ImageMapArea** record. The value of the **delimiter** field MUST be 0xFF. This is an optional field that MUST be present only if the **toolTip** field is not specified.

2.2.41 Chart

The **Chart** record specifies the properties applicable to a **chart** within a [Report](#).

Chart = chartToken chartProperties ReportElementEnd

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | | |
| chartToken | | | | | | | | | | chartProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

chartToken (1 byte): A byte that specifies the start of the **Chart** record. The value of the **chartToken** field MUST be 0x0B.

chartProperties (variable): A record of type [ElementProperties](#) that specifies the properties applicable to the **Chart** record. The structure contained in this field MUST be one of the following.

| Property |
|-------------------------------------|
| ActionImageMapAreas |
| DynamicImageData |
| StreamName |
| UniqueName |
| Label |
| Bookmark |
| Tooltip |
| Style |
| ID |
| Name |
| ToggleItem |

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **Chart** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **chartToken** field of the **Chart** record in the RPL stream.

2.2.42 GaugePanel

The **GaugePanel** record specifies the properties applicable to a **gauge panel** within a [Report](#).

GaugePanel = gaugePanelToken gaugePanelProperties ReportElementEnd

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| gaugePanelToken | | | | | | | | | | gaugePanelProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

gaugePanelToken (1 byte): A byte that specifies the start of the **GaugePanel** record. The value of the **gaugePanelToken** field MUST be 0x0E.

gaugePanelProperties (variable): A record of type [ElementProperties](#) that specifies the properties applicable to the **GaugePanel** record. The structures contained in this field **MUST** be one of the following.

| Property |
|-------------------------------------|
| ActionImageMapAreas |
| DynamicImageData |
| StreamName |
| UniqueName |
| Label |
| Bookmark |
| Tooltip |
| Style |
| ID |
| Name |
| ToggleItem |

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **GaugePanel** record. The **offset** field of the **ReportElementEnd** record **MUST** specify the position of a **gaugePanelToken** field of the **GaugePanel** record in the RPL stream.

2.2.43 Map

Applies to RPL version 10.5 and later

The **Map** record specifies the properties applicable to a map within a [Report](#).

Map = mapToken mapProperties ReportElementEnd

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|--------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| mapToken | | | | | | | | | | mapProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| |
|-----|
| ... |
|-----|

mapToken (1 byte): A byte that specifies the start of the **Map** record. The value of the **mapToken** field MUST be 0x15.

mapProperties (variable): A record of type [ElementProperties](#) that specifies the properties applicable to the **Map** record. The structure contained in this field MUST be one of the following.

| Property |
|-------------------------------------|
| ActionImageMapAreas |
| DynamicImageData |
| StreamName |
| UniqueName |
| Label |
| Bookmark |
| Tooltip |
| Style |
| ID |
| Name |
| ToggleItem |

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **Map** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **mapToken** field of the **ReportElementEnd** record in the RPL stream.

2.2.44 Rectangle

The **Rectangle** record specifies the properties applicable to a rectangle within a [Report](#) record.

Rectangle = rectangleToken rectangleProperties *ReportItem Measurements ReportElementEnd

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|--------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | | | |
| rectangleToken | | | | | | | | | | rectangleProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| |
|-------------------------|
| reportItems (variable) |
| ... |
| measurements (variable) |
| ... |
| reportElementEnd |
| ... |
| ... |

rectangleToken (1 byte): A byte that specifies the start of the **Rectangle** record. The value of the **rectangleToken** field MUST be 0x0A.

rectangleProperties (variable): A record of type [ElementProperties](#) that specifies the properties applicable to the **Rectangle** record. The structures contained in this field MUST be one of the following.

| Property |
|-----------------------------|
| LinkToChild |
| ID |
| Name |
| Tooltip |
| Bookmark |
| Label |
| ToggleItem |
| Style |
| UniqueName |

reportItems (variable): An array of [ReportItem](#) records. The number of elements in the array MUST be equal to the value of the **count** field of the [Measurements](#) record specified in the **measurements** field.

measurements (variable): A **Measurements** record that specifies the size, location, and state properties for every element of the **reportItems** field array. The order of elements in the array MUST match the order of the **ReportItem** records in the **reportItems** field array.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **Rectangle** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **measurements** field of the **Rectangle** record in the RPL stream.

2.2.45 SubReport

The **SubReport** record specifies the properties applicable to a subreport within a [Report](#).

SubReport = subReportToken subReportProperties *ReportItem Measurements ReportElementEnd

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------------|---|---|---|---|---|---|---|---|---|--------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| subReportToken | | | | | | | | | | subReportProperties (variable) | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportItems (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| measurements (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

subReportToken (1 byte): A byte that specifies the start of the **SubReport** record. The value of the **subReportToken** field MUST be 0x0C.

subReportProperties (variable): An [ElementProperties](#) record that specifies the properties applicable to **Report** records. The structure contained in this field MUST be one of the following.

| Property |
|----------------------------|
| Language |
| UniqueName |
| Label |
| Bookmark |

| Property |
|----------------------------|
| Tooltip |
| Style |
| ReportName |
| ID |
| Name |
| ToggleItem |

reportItems (variable): An array of [ReportItem](#) records. The number of elements in the array MUST be equal to the value of the **count** field of the [Measurements](#) record specified in the **measurements** field.

measurements (variable): A **Measurements** record that specifies the size, location, and state properties for every element of the **reportItems** field array. The order of elements in the array MUST match the order of **ReportItem** records in the **reportItems** field array.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **SubReport** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of the **measurements** field of the **SubReport** record in the RPL stream.

2.2.46 RichTextBox

The **RichTextBox** record specifies the properties applicable to a **rich text box** within a [Report](#) record.

```
RichTextBox = richTextBoxToken richTextBoxProperties *ParagraphContent RichTextBoxStructure
ReportElementEnd
```

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
|---------------------------------|---|---|---|---|---|---|---|---|---|----------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| richTextBoxToken | | | | | | | | | | richTextBoxProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| paragraphs (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| richTextBoxStructure (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| | |
|------------------|--|
| reportElementEnd | |
| ... | |
| ... | |

richTextBoxToken (1 byte): A byte that specifies the start of the **RichTextBox** record. The value of the **richTextBoxToken** field MUST be 0x07.

richTextBoxProperties (variable): A [ElementProperties](#) record that specifies the properties applicable to the **RichTextBox** record. If the [OriginalValue](#) property and the [TypeCode](#) property are both present, the **OriginalValue** property MUST appear in the array after the **TypeCode** property. The structure contained in the **richTextBoxProperties** field MUST be one of the following.

| Property |
|---|
| ActionInfo |
| Value |
| ToggleState |
| SortState |
| IsToggleParent |
| ContentHeight |
| ContentOffset |
| ProcessedWithError |
| TypeCode |
| OriginalValue |
| CanGrow |
| CanShrink |
| CanSort |
| Formula |
| IsSimple |
| FormattedValueExpressionBased |
| UniqueName |
| Label |
| Bookmark |
| Tooltip |

| |
|----------------------------|
| Property |
| Style |
| ID |
| Name |
| ToggleItem |

paragraphs (variable): An array of [ParagraphContent](#) records. The number of elements in the array MUST be equal to the value of the **count** field of the [RichTextBoxStructure](#) record defined by the **richTextBoxStructure** field.

richTextBoxStructure (variable): A **RichTextBoxStructure** record.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **RichTextBox** record. The **offset** field for the **ReportElementEnd** record specifies the position of a **richTextBoxStructure** field in the RPL stream.

2.2.47 ParagraphContent

The **ParagraphContent** record specifies a paragraph of text and properties applicable to the entire paragraph. The text is specified through a contiguous collection of [TextRun](#) records.

ParagraphContent = *TextRun Paragraph

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| textRuns (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| paragraph (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

textRuns (variable): An array of **TextRun** records. The number of elements in the array MUST be equal to the value of the **count** field of a [Paragraph](#) record defined by the **paragraph** field.

paragraph (variable): A **Paragraph** record.

2.2.48 TextRun

The **TextRun** record specifies information about a span of text that is part of a [ParagraphContent](#) record.

TextRun = textRunStart textRunProperties delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------|---|---|---|---|---|---|---|---|---|------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| textRunStart | | | | | | | | | | textRunProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

textRunStart (1 byte): A byte that specifies the start of the **TextRun** record. The value of the **textRunStart** field MUST be 0x14.

textRunProperties (variable): A record of type [ElementProperties](#) that specifies the properties applicable to the **TextRun** record. The structure contained in the **textRunProperties** field MUST be one of the following.

| Property |
|------------------------------------|
| UniqueName |
| ID |
| Style |
| Markup |
| Label |
| Tooltip |
| Value |
| ActionInfo |
| Formula |
| ProcessedWithError |
| ContentTop |
| ContentLeft |
| ContentWidth |
| ContentHeight |

delimiter (1 byte): A byte that specifies the end of the **TextRun** record. The value of the **delimiter** field MUST be 0xFF.

2.2.49 Paragraph

The **Paragraph** record specifies a paragraph element of a [ParagraphContent](#) record.

Paragraph = paragraphStart paragraphProperties count offsetsTextRun delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------------|---|---|---|---|---|---|---|---|---|--------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| paragraphStart | | | | | | | | | | paragraphProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| count | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| offsetsTextRun (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

paragraphStart (1 byte): A byte that specifies the start of the **Paragraph** record. The value of the **paragraphStart** field MUST be 0x13.

paragraphProperties (variable) : A record of type [ElementProperties](#) that specifies the properties applicable to the **Paragraph** record. The structure contained in the **paragraphProperties** field MUST be one of the following.

| Property |
|-------------------------------|
| UniqueName |
| ID |
| Style |
| ListStyle |
| ListLevel |
| LeftIndent |
| RightIndent |
| HangingIndent |
| SpaceBefore |
| SpaceAfter |

| Property |
|---------------------------------|
| ParagraphNumber |
| FirstLine |
| ContentTop |
| ContentLeft |
| ContentWidth |
| ContentHeight |

count (4 bytes): A signed integer that MUST be equal to the number of [TextRun](#) records present in the **textRuns** field array of the parent [ParagraphContent](#) record.

offsetsTextRun (variable): An array of 8-byte signed integers. The number of elements in the array MUST be equal to the value of the **count** field. Every element of the array specifies the position of a **TextRun** record in the RPL stream. The order of elements in the array MUST match the order of **TextRun** records in the **textRuns** field array of the parent **ParagraphContent** record.

delimiter (1 byte): A byte that specifies the end of the **Paragraph** record. The value of the **delimiter** field MUST be 0xFF.

2.2.50 RichTextBoxStructure

The **RichTextBoxStructure** record specifies the structure of a [RichTextBox](#) record.

```
RichTextBoxStructure = richTextBoxStructureStart offset count offsetsParagraphContent
delimiter
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------------|---|---|---|---|---|---|---|---|---|------------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| richTextBoxStructureStart | | | | | | | | | | offset | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | offsetsParagraphContent (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

richTextBoxStructureStart (1 byte): A byte that specifies the start of the [RichTextBoxStructure](#) record. The value of the **richTextBoxStructureStart** field MUST be 0x12.

offset (8 bytes): A signed integer that specifies the position of the **richTextBoxToken** field of the parent record of type **RichTextBox** in the RPL stream.

count (4 bytes): A signed integer value that MUST be equal to the number of [ParagraphContent](#) records present in the parent [RichTextBox](#) record.

offsetsParagraphContent (variable): An array of 8-byte signed integers. The number of elements in the array MUST be equal to the value of the **count** field. Every element of the array specifies the position of a **Paragraph** as specified by the **paragraph** field of the **ParagraphContent** record in the RPL stream. The order of elements in the array MUST match the order of the **ParagraphContent** records in a **paragraphs** field array of the parent **RichTextBox** record.

delimiter (1 byte): A byte that specifies the end of the **RichTextBox** record. The value of the **delimiter** field MUST be 0xFF.

2.2.51 Tablix

The **Tablix** record specifies the properties, structures, and layout information applicable to a **tablix** within a [Report](#). The **Tablix** is a flexible grid layout that consists of cells that are organized in rows. A tablix layout is comprised of four areas: corner, **column headers**, **row headers**, and body.

Tablix = tablixToken tablixProperties [TablixContent] TablixStructure ReportElementEnd

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------------|---|---|---|---|---|---|---|---|---|-----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| tablixToken | | | | | | | | | | tablixProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| tablixContent (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| tablixStructure (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| reportElementEnd | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| | |
|-----|--|
| ... | |
|-----|--|

tablixToken (1 byte): A byte that specifies the start of the **Tablix** record. The value of the **tablixToken** field MUST be 0x0D.

tablixProperties (variable): A record of type [ElementProperties](#) that specifies the properties applicable to the **Tablix** record. The structures contained in the **tablixProperties** field MUST be one of the following.

| Property |
|----------------------------|
| UniqueName |
| ID |
| Name |
| Tooltip |
| Bookmark |
| Label |
| ToggleItem |
| Style |

tablixContent (variable): A [TablixContent](#) record that specifies the content of grid cells and layout information for the cells of a **tablix body area**.

tablixStructure (variable): A [TablixStructure](#) record.

reportElementEnd (10 bytes): A [ReportElementEnd](#) record that marks the end of the **Tablix** record. The **offset** field of the **ReportElementEnd** record MUST specify the position of a **tablixStructure** field of the **Tablix** record in the RPL stream.

2.2.52 TablixContent

The **TablixContent** record specifies the content of a [Tablix](#) record. The child records MUST be of type [ReportItem](#) or [TablixBodyRow](#).

```
TablixContent = *(ReportItem / TablixBodyRow)
```

2.2.53 TablixStructure

The **TablixStructure** record specifies information about the layout of a [Tablix](#) record and the position of the **Tablix** record in the RPL stream.

```
TablixStructure = tablixStructureStart offset TablixMeasurements delimiter
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|-------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| tablixStructureStart | | | | | | | | | | offset | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | tablixMeasurements (variable) | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

tablixStructureStart (1 byte): A byte that specifies the start of the **TablixStructure** record. The value of the **tablixStructureStart** field MUST be 0x11.

offset (8 bytes): A signed integer that specifies the position of the **tablixStart** field of the parent record of type **Tablix** in the RPL stream.

tablixMeasurements (variable): A record of type [TablixMeasurements](#).

delimiter (1 byte): A byte that specifies the end of the **TablixStructure** record. The value of the **delimiter** field MUST be 0xFF.

2.2.54 TablixMeasurements

The **TablixMeasurements** record specifies the layout information of a [Tablix](#) record.

TablixMeasurements = tablixMeasurementsProperties *TablixRow delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| tablixMeasurementsProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| tablixRows (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

tablixMeasurementsProperties (variable): A collection of structures. The order of the structures in the collection is arbitrary. The structures in the collection MUST be one of the following.

| Property |
|-------------------------------------|
| ColumnHeaderRows |
| RowHeaderColumns |
| ColsBeforeRowHeader |
| LayoutDirection |
| ColumnsWidths |
| RowHeights |
| ContentTop |
| ContentLeft |
| TablixRowMembersDef |
| TablixColMembersDef |

tablixRows (variable): An array of [TablixRow](#) records.

delimiter (1 byte): A byte that specifies the end of the **TablixMeasurements** record. The value of the **delimiter** field MUST be 0xFF.

2.2.55 ColumnsWidths

The **ColumnsWidths** record specifies the number of columns a [Tablix](#) record contains and the properties associated to each of these columns. The columns MUST be ordered according to the [LayoutDirection](#) and [ColsBeforeRowHeader](#) properties.

ColumnsWidths = columnsWidthsStart count *ColumnsInfo

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------|---|---|---|---|---|---|---|---|---|--------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| columnsWidthsStart | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | columns (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

columnsWidthsStart (1 byte): A byte that specifies the start of the **ColumnsWidths** record. The value of the **columnsWidthsStart** field MUST be 0x04.

count (4 bytes): A signed integer that specifies the number of columns contained in the corresponding **Tablix** record. The value of this field MUST be greater than or equal to the value of the [RowHeaderColumns](#) property of the **Tablix**.

columns (variable): An array of [ColumnInfo](#) records. The number of elements in the array MUST be equal to the value of the **count** field.

2.2.56 ColumnInfo

The **ColumnInfo** record specifies the properties applicable to a column of a [Tablix](#) record.

`ColumnInfo = width fixed`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| width | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| fixed | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

width (4 bytes): A floating point value that specifies the width of the column in millimeters. The value of the **width** field MUST be greater than or equal to zero. [<9>](#)

fixed (1 byte): A **Boolean** value that indicates if the column should be displayed on the page when the user scrolls part of the tablix off the page. The value of the **fixed** field MUST be one of the following.

| Value | Meaning |
|-------|---------------------------------|
| 0x00 | False, column is not displayed. |
| 0x01 | True, column is displayed. |

2.2.57 RowHeights

The **RowHeights** record specifies the number of rows a [Tablix](#) record contains and the properties associated to each of these rows.

`RowHeights = rowHeightsStart count *RowInfo`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|-------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| rowHeightsStart | | | | | | | | | | | | | | | | count | | | | | | | | | | | | | | | |

| | |
|-----|-----------------|
| ... | rows (variable) |
| ... | |

rowHeightsStart (1 byte): A byte that specifies the start of the **RowHeights** record. The value of the **rowHeightsStart** field MUST be 0x05.

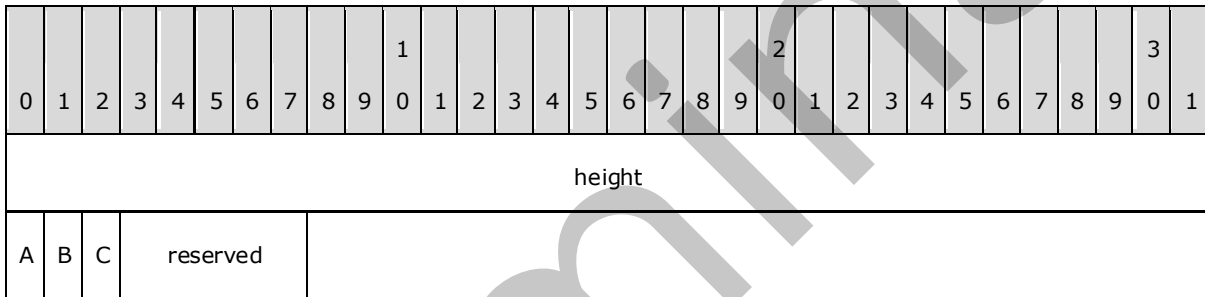
count (4 bytes): A signed integer that specifies the number of rows contained in the corresponding **Tablix** record. The value of this field MUST be greater than or equal to the value of the [ColumnHeaderRows](#) property of the **Tablix**.

rows (variable): An array of [RowInfo](#) records. The number of elements in the array MUST be equal to the value of the **count** field.

2.2.58 RowInfo

The **RowInfo** record specifies the properties applicable to a row of a [Tablix](#) record.

RowInfo = height byte



height (4 bytes): A floating point value that specifies the height of the row in millimeters. This MUST be a value greater than or equal to zero. [<10>](#)

A - FixedRows (1 bit): A bit field that specifies if the row should be displayed on the page when the user scrolls part of the tablix off the page.

The value of the **FixedRows** bit field MUST be one of the following.

| Value | Meaning |
|-------|---------|
| 0x0 | False |
| 0x1 | True |

B - SharedLayoutRow (1 bit): A bit field that specifies if the tablix row has a layout that can be shared by other rows that follow this one. The rows that share the same layout MUST be consecutive entries in the **tablixRows** field array of [TablixMeasurements](#) record. The **UseSharedLayoutRow** and **SharedLayoutRow** fields MUST NOT be set at the same time. [<11>](#)
The value of this bit field MUST be one of the following.

| Value | Meaning |
|-------|---------|
| 0x0 | False |
| 0x1 | True |

C - UseSharedLayoutRow (1 bit): A bit that specifies if the tablix row uses the same layout as a previous row in the **tablixRows** field array of **TablixMeasurements** record. The **UseSharedLayoutRow** and **SharedLayoutRow** MUST NOT be set at the same time. <12> The value of the **UseSharedLayoutRow** bit field MUST be one of the following.

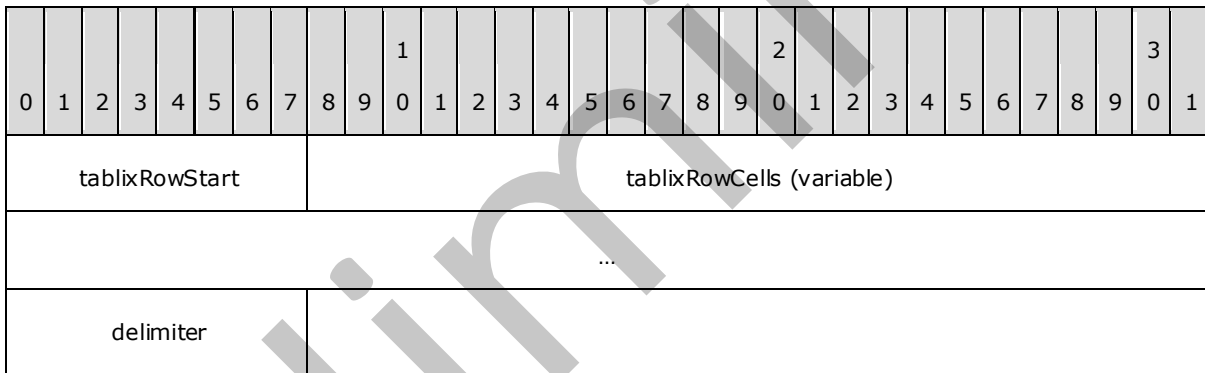
| Value | Meaning |
|-------|---------|
| 0x0 | False |
| 0x1 | True |

reserved (5 bits): The value of the **reserved** field MUST be zero and MUST be ignored.

2.2.59 TablixRow

The **TablixRow** record specifies a row of a [Tablix](#) record.

```
TablixRow = tablixRowStart *TablixRowCell delimiter
```



tablixRowStart (1 byte): A byte that specifies the start of the **TablixRow** record. The value of the **tablixRowStart** byte field MUST be 0x08.

tablixRowCells (variable) : An array of [TablixRowCell](#) records. The elements MUST be ordered according to the [LayoutDirection](#) property of the **Tablix**. A [TablixCorner](#) or a [TablixColumnHeader](#) structure MUST NOT be part of a row that contains a [TablixRowHeader](#) or [TablixBodyRowCells](#) structure.

delimiter (1 byte): A byte that specifies the end of the **TablixRow** record. The value of the **delimiter** field MUST be 0xFF.

2.2.60 TablixRowCell

The **TablixRowCell** record specifies the content of a cell in a [Tablix](#) record.

TablixRowCell = TablixCorner / TablixColumnHeader / TablixRowHeader / TablixBodyRowCells

2.2.61 TablixCorner

The **TablixCorner** record specifies a cell of the corner area of a [Tablix](#) record.

TablixCorner = tablixCornerStart [tablixCellProperties] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------|---|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| tablixCornerStart | | | | | | | | | | tablixCellProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

tablixCornerStart (1 byte): A byte that specifies the start of the **TablixCorner** record. The value of the **tablixCornerStart** field MUST be 0x0A.

tablixCellProperties (variable): A collection of structures. The order of the structures in the collection is arbitrary. The structures in the collection MUST be one of the following.

| Property |
|--------------------------------|
| CellItemOffset |
| ColSpan |
| RowSpan |
| ColumnIndex |
| RowIndex |
| CellItemState |
| ContentTop |
| ContentLeft |
| ContentWidth |
| ContentHeight |

delimiter (1 byte): A byte that specifies the end of the **TablixCorner** record. The value of the **delimiter** field MUST be 0xFF.

2.2.62 TablixColumnHeader

The **TablixColumnHeader** record specifies a cell of the column headers area of a [Tablix](#) record.

TablixColumnHeader = tablixColumnHeaderStart [tablixCellProperties] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------------|---|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| tablixColumnHeaderStart | | | | | | | | | | tablixCellProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

tablixColumnHeaderStart (1 byte): A byte that specifies the start of the **TablixColumnHeader** record. The value of the **tablixColumnHeaderStart** field MUST be 0x0B.

tablixCellProperties (variable): A collection of structures. The order of the structures in the collection is arbitrary. The structures in the collection MUST be one of the following.

| Property |
|--------------------------------------|
| CellItemOffset |
| ColSpan |
| RowSpan |
| DefIndex |
| ColumnIndex |
| RowIndex |
| GroupLabel |
| UniqueName |
| State |
| CellItemState |
| RecursiveToggleLevel |
| ContentTop |
| ContentLeft |
| ContentWidth |
| ContentHeight |

delimiter (1 byte): A byte that specifies the end of the **TablixColumnHeader** record. The value of the **delimiter** field MUST be 0xFF.

2.2.63 TablixRowHeader

The **TablixRowHeader** record specifies a cell of the row headers area of a [Tablix](#) record.

TablixRowHeader = tablixRowHeaderStart [tablixCellProperties] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| tablixRowHeaderStart | | | | | | | | | | tablixCellProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

tablixRowHeaderStart (1 byte): A byte that specifies the start of the **TablixRowHeader** record. The value of the **tablixRowHeaderStart** field MUST be 0x0C.

tablixCellProperties (variable): A collection of structures. The order of the structures in the collection is arbitrary. The structures in the collection MUST be one of the following.

| Property |
|--------------------------------------|
| CellItemOffset |
| ColSpan |
| RowSpan |
| DefIndex |
| ColumnIndex |
| RowIndex |
| GroupLabel |
| UniqueName |
| State |
| CellItemState |
| RecursiveToggleLevel |
| ContentTop |
| ContentLeft |

| Property |
|-------------------------------|
| ContentWidth |
| ContentHeight |

delimiter (1 byte): A byte that specifies the end of the **TablixRowHeader** record. The value of the **delimiter** field MUST be 0xFF.

2.2.64 TablixBodyRowCells

The **TablixBodyRowCells** record specifies the position in the RPL stream of the [TablixBodyRow](#) record that is part of the parent [TablixRow](#) record.

`TablixBodyRowCells = tablixBodyRowCellsStart offset`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------------|---|---|---|---|---|---|---|---|---|--------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| tablixBodyRowCellsStart | | | | | | | | | | offset | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

tablixBodyRowCellsStart (1 byte): A byte that specifies the start of the **TablixBodyRowCells** record. The value of the **tablixBodyRowCellsStart** field MUST be 0x09.

offset (8 bytes): A signed integer that specifies the position of the **TablixBodyRow** record in the RPL stream.

2.2.65 TablixBodyRow

The **TablixBodyRow** record specifies the content of a row in the body area of a [Tablix](#) record.

`TablixBodyRow = tablixBodyRowStart rowIndex *TablixBodyCell delimiter`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------|---|---|---|---|---|---|---|---|---|----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| tablixBodyRowStart | | | | | | | | | | rowIndex | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | tablixBodyCells (variable) | | | | | | | | | | | | | | | | | | | | | | | |

| | |
|-----------|--|
| ... | |
| delimiter | |

tablixBodyRowStart (1 byte): A byte that specifies the start of the **TablixBodyRow** record. The value of the **tablixBodyRowStart** field MUST be 0x12.

RowIndex (4 bytes): A signed integer that specifies the index of the row. The value of this field identifies the corresponding [RowInfo](#) structure in a **rows** field array of a [TablixMeasurements.RowHeights](#) record.

tablixBodyCells (variable) : An array of [TablixBodyCell](#) records. The elements in the array MUST be ordered according to the **TablixLayout** property of a tablix.

delimiter (1 byte): A byte that specifies the end of the **TablixBodyRow** record. The value of the **delimiter** field MUST be 0xFF.

2.2.66 TablixBodyCell

The **TablixBodyCell** record specifies a cell of a body area of a [Tablix](#) record.

```
TablixBodyCell = tablixBodyCellStart [tablixCellProperties] delimiter
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|--|
| | 1 | | 2 | | 3 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | | | |
| tablixBodyCellStart | | | | | | | | | | tablixCellProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

tablixBodyCellStart (1 byte): A byte that specifies the start of the **TablixBodyCell** record. The value of the **tablixBodyCellStart** field MUST be 0x0D.

tablixCellProperties (variable): A collection of structures. The order of the structures in the collection is arbitrary. The structures in the collection MUST be one of the following.

| Property |
|--------------------------------|
| CellItemOffset |
| ColSpan |
| RowSpan |
| ColumnIndex |

| Property |
|-------------------------------|
| CellItemState |
| ContentTop |
| ContentLeft |
| ContentWidth |
| ContentHeight |

delimiter (1 byte): A byte that specifies the end of the **TablixBodyCell** record. The value of the **delimiter** byte field MUST be 0xFF.

2.2.67 TablixRowMembersDef

The **TablixRowMembersDef** record specifies a list of [TablixMemberDef](#) records that are referenced by [TablixRowHeader](#) records.

TablixRowMembersDef = tablixRowMembersDefStart count *TablixMemberDef

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------------|---|---|---|---|---|---|---|---|---|-------------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| tablixRowMembersDefStart | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | tablixMembersDefContents (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

tablixRowMembersDefStart (1 byte): A byte that specifies the start of the **TablixRowMemberDef** record. The value of the **tablixRowMembersDefStart** field MUST be 0x0E.

count (4 bytes): A signed integer that specifies the number of **TablixMemberDef** records associated with elements of the row headers area of the [Tablix](#) record.

tablixMembersDefContents (variable) : An array of **TablixMemberDef** records. The number of elements in the array MUST be equal to the value of the **count** field.

2.2.68 TablixColMembersDef

The **TablixColMembersDef** record specifies a list of [TablixMemberDef](#) records that are referenced by [TablixColumnHeader](#) records.

TablixColMembersDef = tablixColMembersDefStart count *TablixMemberDef

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------------|---|---|---|---|---|---|---|---|---|-------------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| tablixColMembersDefStart | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | tablixMembersDefContents (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

tablixColMembersDefStart (1 byte): A byte that specifies the start of the **TablixColMembersDef** record. The value of the **tablixColMembersDefStart** field MUST be 0x0F.

count (4 bytes): A signed integer that specifies the number of **TablixMemberDef** records associated with elements of associated with elements of the column headers area of the [Tablix](#) record.

tablixMembersDefContents (variable): An array of **TablixMemberDef** records. The number of elements in the array MUST be equal to the value of the **count** field.

2.2.69 TablixMemberDef

The **TablixMemberDef** record specifies the definition of the **column members** and **row members** of a [Tablix](#) record.

TablixMemberDef = tablixMemberDefStart [tablixMemberDefProperties] delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|--------------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| tablixMemberDefStart | | | | | | | | | | tablixMemberDefProperties (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| delimiter | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

tablixMemberDefStart (1 byte): A byte that specifies the start of the **TablixMemberDef** record. The value of the **tablixMemberDefStart** field MUST be 0x10.

tablixMemberDefProperties (variable): A collection of structures. The order of the structures in the collection is arbitrary. The structures in the collection MUST be one of the following.

| Property |
|--------------------------------|
| DefinitionPath |

| Property |
|---------------------------------|
| Level |
| MemberCellIndex |
| MemberDefState |

delimiter (1 byte): A byte that specifies the end of **TablixMemberDef** record. The value of the **delimiter** field MUST be 0xFF.

2.2.70 Measurements

The **Measurements** record specifies the properties of an element in the parent record.

Measurements = measurementsStart offset count measurementsContent

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------|---|---|---|---|---|---|---|---|---|--------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| measurementsStart | | | | | | | | | | offset | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | count | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | measurementsContent (variable) | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

measurementsStart (1 byte): A byte that specifies the start of the **Measurements** record. The value of the **measurementsStart** field MUST be 0x10.

offset (8 bytes): A signed integer that MUST specify the position of the parent record in the RPL stream. The parent record MUST be one of the following.

| Parent record |
|-----------------------------------|
| PageContent |
| SimpleSection |
| MixedSection |
| BodyAreaElement |
| PageHeaderElement |
| PageFooterElement |

| |
|-----------------------------|
| Parent record |
| BodyElement |
| Rectangle |
| SubReport |

count (4 bytes): A signed integer that specifies the number of elements the parent record contains.

measurementsContent (variable): An array of [Measurement](#) structures. The number of elements in the array MUST be equal to the value of the **count** field.

2.2.71 Measurement

The **Measurement** record specifies the properties of an element in the grandparent record.

```
Measurement = (left, top, width, height, zIndex, state, reportElementEndOffset)
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------|---|---|---|---|---|---|---|---|---|---|---|---|---|------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| left | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| top | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| width | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| height | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| zIndex | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| state | | | | | | | | | | | | | | reportElementEndOffset | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

left (4 bytes): A field that specifies the left position, expressed in millimeters, of the element relative to its parent record.

top (4 bytes): A field that specifies the top position, expressed in millimeters, of the element relative to its parent record.

width (4 bytes): A field that specifies the width, expressed in millimeters, of the element.

height (4 bytes): A field that specifies the height, expressed in millimeters, of the element.

zIndex (4 bytes): A field that specifies the **zIndex** property. The **zIndex** is the drawing order of the items within the panel.

state (1 byte): A field of type byte that specifies information for the element. The meaning of this field value is derived from the [StateValues](#) enumeration.

reportElementEndOffset (8 bytes) : A signed integer that specifies the position of the [ReportElementEnd](#) record in the RPL stream for an element in the grandparent record. The **offset** MUST point to a corresponding item (identified by the same index in the collection) in the grandparent record.

| Record | Corresponding element type |
|-----------------------------------|---|
| PageContent | Section |
| SimpleSection | BodyAreaElement PageHeaderElement PageFooterElement |
| MixedSection | BodyAreaElement PageHeaderElement PageFooterElement |
| BodyAreaElement | BodyElement |
| PageHeaderElement | ReportItems |
| PageFooterElement | ReportItems |
| BodyElement | ReportItems |
| Rectangle | ReportItems |
| SubReport | ReportItems |

2.2.72 ReportElementEnd

The **ReportElementEnd** record specifies the end of a variable size record.

ReportElementEnd = elementEnd offset delimiter

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------|---|---|---|---|---|---|---|---|---|--------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| elementEnd | | | | | | | | | | offset | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

| | | |
|-----|-----------|--|
| ... | delimiter | |
|-----|-----------|--|

elementEnd (1 byte): A byte that specifies the start of the **ReportElementEnd** record. The value of this byte field MUST be 0xFE.

offset (8 bytes): A signed integer that points to the start of a record in the RPL stream. The record identified by the **offset** field depends on the record type that the **ReportElementEnd** record ends. The variable size records MUST be one of the following.

| Record type | Offset points to |
|-----------------------------------|---------------------------|
| Report | offsetsArrayElement |
| PageContent | measurements |
| SimpleSection | measurements |
| MixedSection | measurements |
| BodyAreaElement | measurements |
| PageHeaderElement | measurements |
| PageFooterElement | measurements |
| BodyElement | measurements |
| Line | lineToken |
| Image | imageToken |
| Chart | chartToken |
| GaugePanel | gaugePanelToken |
| Map | mapToken |
| Rectangle | measurements |
| SubReport | measurements |
| RichTextBox | richTextBoxStructureStart |
| Tablix | tablixStructureStart |

delimiter (1 byte): A byte that specifies the end of **ReportElementEnd** record. The value of the **delimiter** field MUST be 0xFF.

2.3 Properties

This section specifies the properties that can be used in an RPL stream.

2.3.1 ID

The **ID** property specifies a value that uniquely identifies the parent record of this property. If this property is specified, it MUST be part of a [SectionProperties](#), [PageProperties](#), or [ElementProperties.InlineSharedElementProperties](#) record.

ID = idStart idValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------|---|---|---|---|---|---|---|---|---|--------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| idStart | | | | | | | | | | idValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

idStart (1 byte): A byte that specifies the start of the **ID** property. The value of the **idStart** field MUST be one of the following values.

| Value | Parent |
|-------|--|
| 0x00 | SectionProperties |
| 0x01 | PageProperties ElementProperties.InlineSharedElementProperties for ReportItem ElementProperties.InlineSharedElementProperties for PageHeaderElement ElementProperties.InlineSharedElementProperties for PageFooterElement ElementProperties.InlineSharedElementProperties for BodyElement |
| 0x05 | ElementProperties.InlineSharedElementProperties for Paragraph ElementProperties.InlineSharedElementProperties for TextRun |

idValue (variable): A **String** variable that specifies the unique identifier of a structure specified in the table above.

2.3.2 ColumnCount

The **ColumnCount** property specifies the number of columns in a [Section](#) record. If this property is specified, it MUST be part of a [SectionProperties](#) record. If the **ColumnCount** property is not present, the default value is used.

ColumnCount = columnCountStart columnCountValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| columnCountStart | | | | | | | | | | columnCountValue | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

columnCountStart (1 byte): A byte that specifies the start of the **ColumnCount** property. The value of the **columnCountStart** field MUST be the following value.

| Value | Parent |
|-------|--------------------------|
| 0x01 | SectionProperties |

columnCountValue (4 bytes): An **integer** that specifies the number of columns in a section. The default value is 0x00000000.

2.3.3 ColumnSpacing

The **ColumnSpacing** property specifies the spacing between each column within a single section in multicolumn renderings. If this property is specified, it MUST be part of a [SectionProperties](#) or [PageProperties](#) record. If the **ColumnSpacing** property is not specified, the default value is used.

ColumnSpacing = columnSpacingStart columnSpacingValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------|---|---|---|---|---|---|---|---|---|--------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| columnSpacingStart | | | | | | | | | | columnSpacingValue | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

columnSpacingStart (1 byte): A byte that specifies the start of the **ColumnSpacing** property. The value of **columnSpacingStart** field MUST be one of the following values.

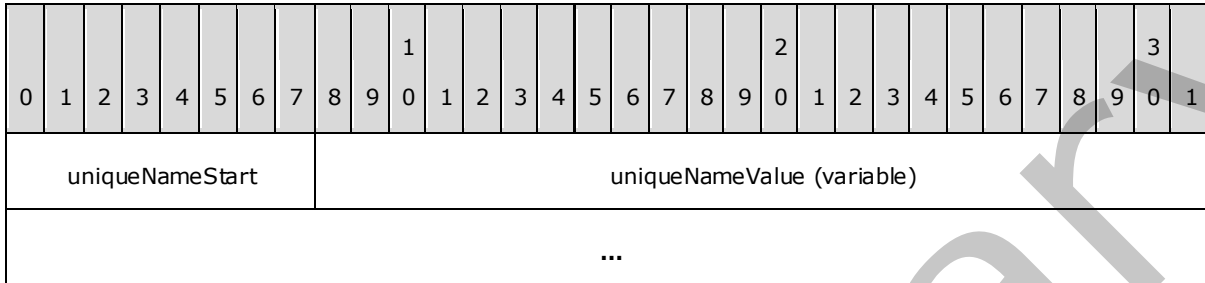
| Value | Parent |
|-------|--------------------------|
| 0x02 | SectionProperties |
| 0x16 | PageProperties |

columnSpacingValue (4 bytes): A floating point value that specifies the spacing between each column in a section. The default value is 0x00000000.

2.3.4 UniqueName

The **UniqueName** property specifies a value that uniquely identifies a record. If this property is specified, it MUST be part of an [ElementProperties.NonSharedElementProperties](#), [TablixRowHeader](#), or [TablixColumnHeader](#) record.

```
UniqueName = uniqueNameStart uniqueNameValue
```



uniqueNameStart (1 byte): A byte that specifies the start of the **UniqueName** property. The value of the **uniqueNameStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x00 | ElementProperties.NonSharedElementProperties for ReportItem ElementProperties.NonSharedElementProperties for PageHeaderElement ElementProperties.NonSharedElementProperties for PageFooterElement ElementProperties.NonSharedElementProperties for BodyElement PageProperties |
| 0x04 | ElementProperties.NonSharedElementProperties for Paragraph ElementProperties.NonSharedElementProperties for TextRun |
| 0x0B | TablixColumnHeader TablixRowHeader |

uniqueValue (variable): A **String** variable that specifies the unique identifier of a structure specified in the preceding table.

2.3.5 Name

The **Name** property specifies the name of a [ReportItem](#) record. If the **Name** property is specified, it MUST be part of a [ReportItem.ElementProperties.InlineSharedElementProperties](#) record.

```
Name = nameStart nameValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|---|---|---|---|---|---|---|---|---|----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| nameStart | | | | | | | | | | nameValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

nameStart (1 byte): A byte that specifies the start of the **Name** property. The value of the **nameStart** field MUST be the following value.

| Value | Parent |
|-------|---|
| 0x02 | ElementProperties.InlineSharedElementProperties for ReportItem |

nameValue (variable): A **String** variable that specifies the name of a **ReportItem**.

2.3.6 Label

The **Label** property specifies a label for a [ReportItem](#), [TextRun](#), or [Action](#) record. If the **Label** property is specified, it MUST be part of a [ReportItem.ElementProperties](#), **TextRun.ElementProperties**, or **Action** record.

Label = labelStart labelValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------|---|---|---|---|---|---|---|---|---|-----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| labelStart | | | | | | | | | | labelValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

labelStart (1 byte): A byte that specifies the start of the **Label** property. The value of the **labelStart** field MUST be one of the following values.

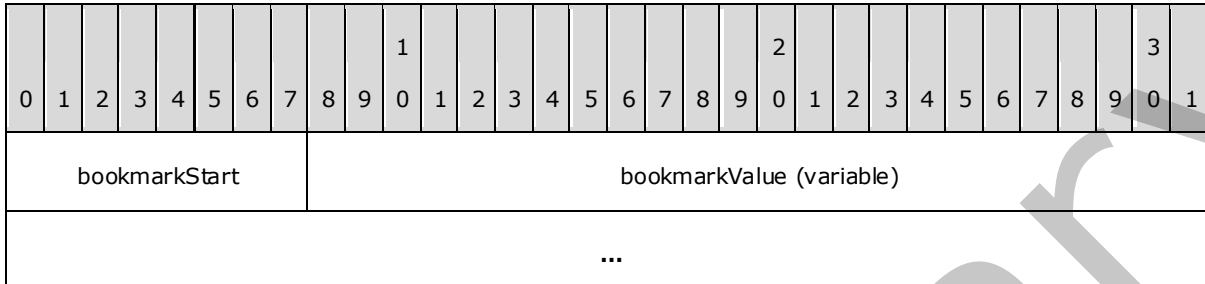
| Value | Parent |
|-------|---|
| 0x03 | ElementProperties.InlineSharedElementProperties for ReportItem ElementProperties.NonSharedElementProperties for ReportItem |
| 0x04 | Action |
| 0x08 | ElementProperties.InlineSharedElementProperties for TextRun ElementProperties.NonSharedElementProperties for TextRun |

labelValue (variable): A **String** variable that specifies a user-friendly label for the record.

2.3.7 Bookmark

The **Bookmark** property specifies a **bookmark** that can be linked to via a bookmark action in a report. If this property is specified, it MUST be part of a [ReportItem.ElementProperties](#) record.

```
Bookmark = bookmarkStart bookmarkValue
```



bookmarkStart (1 byte): A byte that specifies the start of the **Bookmark** property. The value of the **bookmarkStart** field MUST be one of the following values.

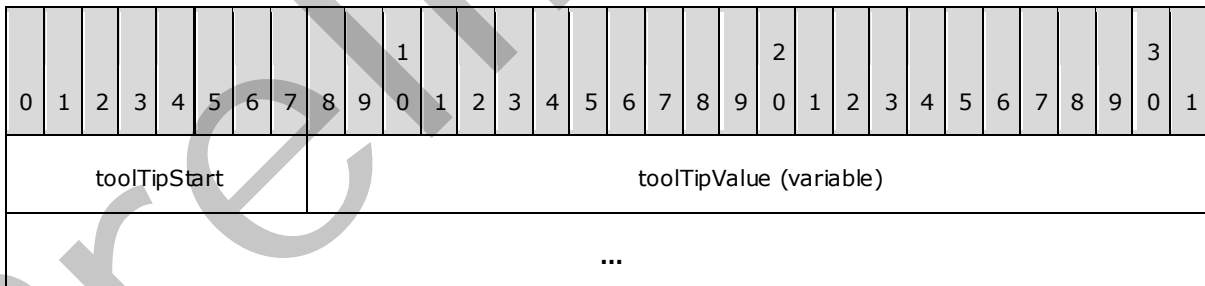
| Value | Parent |
|-------|--|
| 0x04 | ElementProperties.InlineSharedElementProperties for ReportItem ElementProperties.NonSharedElementProperties for ReportItem |

bookmarkValue (variable): A **String** variable that specifies a bookmark that can be linked to via a bookmark action.

2.3.8 ToolTip

The **ToolTip** property specifies a textual label that provides a description for a record. If the **ToolTip** property is specified, it MUST be part of a [ReportItem.ElementProperties](#), **TextRun.ElementProperties**, or [ImageMapArea](#) record.

```
ToolTip = toolTipStart toolTipValue
```



toolTipStart (1 byte): A byte that specifies the start of the **ToolTip** property. The value of the **toolTipStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x05 | ElementProperties.InlineSharedElementProperties for ReportItem ElementProperties.NonSharedElementProperties for ReportItem ImageMapArea |
| 0x09 | ElementProperties.InlineSharedElementProperties for TextRun ElementProperties.NonSharedElementProperties for TextRun |

tooltipValue (variable): A **String** variable that specifies a textual label for a report item or **text run**.

2.3.9 ToggleItem

The **ToggleItem** property specifies a text box that users click to show or hide the current item. The **ToggleItem** property MUST be part of a [ReportItem.ElementProperties.InlineSharedElementProperties](#) record.

If the **ToggleItem** property is not specified, the current item does not toggle. **ToggleItem** MUST NOT be used in [PageHeaderElement](#) or [PageFooterElement](#) records, nor can it refer to items within them. <13>

ToggleItem = toggleItemStart toggleItemValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|---|---|---|---|---|---|---|---|---|----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| toggleItemStart | | | | | | | | | | toggleItemValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

toggleItemStart (1 byte): A byte that specifies the start of the **ToggleItem** property. The value of **toggleItemStart** field MUST be the following value.

| Value | Parent |
|-------|---|
| 0x08 | ElementProperties.InlineSharedElementProperties for ReportItem |

toggleItemValue (variable): A **String** variable that specifies the corresponding text box with the matching [Name](#) property that a user clicks to set the visibility of the current item.

2.3.10 Description

The **Description** property specifies the description of a [Report](#) record. If this property is specified, it MUST be part of a [ReportProperties](#) record.

Description = descriptionStart descriptionValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| descriptionStart | | | | | | | | | | descriptionValue | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

descriptionStart (1 byte): A byte that specifies the start of the **Description** property. The value of the **descriptionStart** field MUST be the following value.

| Value | Parent |
|-------|-------------------------|
| 0x09 | ReportProperties |

descriptionValue (variable): A variable **String** that specifies the description of the report.

2.3.11 Location

The **Location** property specifies the location of a [Report](#) record as an **absolute URL**. If the **Location** property is specified, it MUST be part of a [ReportProperties](#) record.

Location = locationStart locationValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------|---|---|---|---|---|---|---|---|---|--------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| locationStart | | | | | | | | | | locationValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

locationStart (1 byte): A byte that specifies the start of the **Location** property. The value of the **locationStart** field MUST be the following value.

| Value | Parent |
|-------|-------------------------|
| 0x0A | ReportProperties |

locationValue (variable): A variable **String** that specifies the location of the report.

2.3.12 Language

The **Language** property specifies the primary **language** of the text in a record. If this property is specified, it MUST be part of a [ReportProperties](#), [SubReport.ElementProperties.NonSharedElementProperties](#), or [Style](#) record. If the **Language** property of the [SubReport](#) record is not specified, the **Language** property MUST default to the **Language** property of a [Report](#) record that contains the current item. If the **Language** property of

the **Style** record is not specified, the **Language** property MUST default to the **Language** property of the **Report** or **SubReport** record that contains the current item.

Language = languageStart languageValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------|---|---|---|---|---|---|---|---|---|--------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| languageStart | | | | | | | | | | languageValue (variable) | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

languageStart (1 byte): A byte that specifies the start of the **Language** property. The value of the **languageStart** field MUST be one of the following values.

| Value | Parent |
|-------|--|
| 0x0B | ReportProperties |
| 0x0B | ElementProperties.NonSharedElementProperties for SubReport |
| 0x20 | Style.SharedStyleProperties |
| 0x20 | Style.NonSharedStyleProperties |

languageValue (variable): A **String** variable that MUST be a valid language code, which is a combination of an ISO 639 two-letter lowercase culture code associated with a language, and an ISO 3166 (see [\[ISO-3166\]](#)) two-letter uppercase subculture code associated with a country or region.

2.3.13 ExecutionTime

The **ExecutionTime** property specifies the date and time that a report was generated for transmission. If this property is specified, it MUST be part of a [ReportProperties](#) record.

ExecutionTime = executionTimeStart executionTimeValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------|---|---|---|---|---|---|---|---|---|--------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| executionTimeStart | | | | | | | | | | executionTimeValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

executionTimeStart (1 byte): A byte that specifies the start of the **ExecutionTime** property. The value of **executionTimeStart** field MUST be the following value.

| Value | Parent |
|-------|-------------------------|
| 0x0C | ReportProperties |

executionTimeValue (8 bytes): A signed integer that specifies the date and time as the number of 100 non-second intervals since 12:00 AM January 1, year 1 A.D. in the proleptic Gregorian Calendar, and indicates the date and time that the [Report](#) record was executed.

2.3.14 Author

The **Author** property specifies the name of the author of a report. If this property is specified, it MUST be part of a [ReportProperties](#) record.

Author = authorStart authorValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|---|---|---|---|---|---|---|---|---|------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| authorStart | | | | | | | | | | authorValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

authorStart (1 byte): A byte that specifies the start of the **Author** property. The value of the **authorStart** field MUST be the following value.

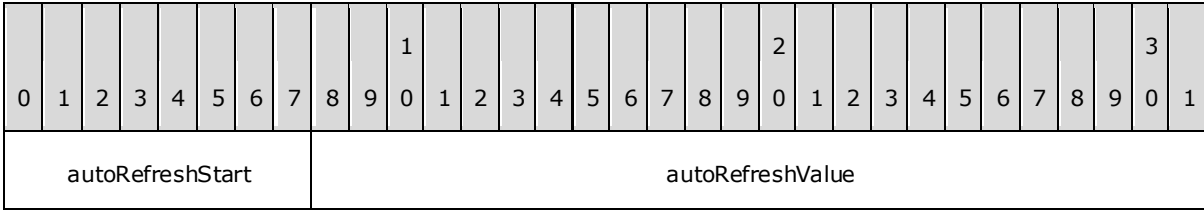
| Value | Parent |
|-------|-------------------------|
| 0x0D | ReportProperties |

authorValue (variable): A variable **String** containing the name of the author of the [Report](#) record.

2.3.15 AutoRefresh

The **AutoRefresh** property specifies, in seconds, how often to refresh a report. If this property is specified, it MUST be part of [ReportProperties](#) record. If the **AutoRefresh** property is not specified, the default value is used.

AutoRefresh = autoRefreshStart autoRefreshValue



autoRefreshStart (1 byte): A byte that specifies the start of the **AutoRefresh** property. The value of **autoRefreshStart** field MUST be the following value.

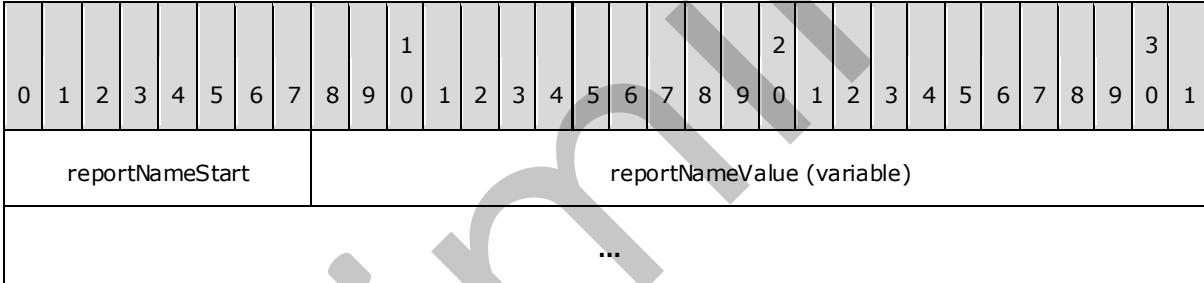
| Value | Parent |
|-------|-------------------------|
| 0x0E | ReportProperties |

autoRefreshValue (4 bytes): A signed integer that specifies the number of seconds to wait before the [Report](#) record refreshes. The default value is 0x00000000.

2.3.16 ReportName

The **ReportName** property specifies the name of a report. If this property is specified, it MUST be part of a [ReportProperties](#) or [SubReport.ElementProperties.InlineSharedElementProperties](#) record.

ReportName = reportNameStart reportNameValue



reportNameStart (1 byte): A byte that specifies the start of the **ReportName** property. The value of the **reportNameStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x0F | ReportProperties ElementProperties.InlineSharedElementProperties for SubReport |

reportNameValue (variable): A variable **String** containing the name of a [Report](#) record.

2.3.17 PageHeight

The **PageHeight** property specifies the default height for rendering a report in a **physical-page oriented renderer**. If this property is specified, it MUST be part of a [PageLayout](#) or [PageProperties](#) record. If the **PageHeight** property is not specified, the default value is used.

PageHeight = pageHeightStart pageHeightValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|---|---|---|---|---|---|---|---|---|-----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| pageHeightStart | | | | | | | | | | pageHeightValue | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

pageHeightStart (1 byte): A byte that specifies the start of the **PageHeight** property. The value of the **pageHeightStart** field MUST be one of the following values.

| Value | Parent |
|-------|--|
| 0x10 | PageLayout PageProperties |

pageHeightValue (4 bytes): A floating point value that specifies the default height, in millimeters, for rendering the [Report](#) record in a physical-page oriented renderer. The default value is 0x00000000 millimeters.

2.3.18 PageWidth

The **PageWidth** property specifies the default width for rendering a report in a physical-page oriented renderer. If the **PageWidth** property is specified, it MUST be part of a [PageLayout](#) or [PageProperties](#) record. If the **PageWidth** property is not specified, the default value is used.

PageWidth = pageWidthStart pageWidthValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| pageWidthStart | | | | | | | | | | pageWidthValue | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

pageWidthStart (1 byte): A byte that specifies the start of the **PageWidth** property. The value of the **pageWidthStart** field MUST be one of the following values.

| Value | Parent |
|-------|-------------------|
| 0x11 | PageLayout |

pageWidthValue (4 bytes): A floating point value that specifies the default width, in millimeters, for rendering a [Report](#) record in a physical-page oriented renderer. The default value is 0x00000000 millimeters.

2.3.19 MarginTop

The **MarginTop** property specifies the size of the top margin of a page. If this property is specified, it MUST be part of a [PageLayout](#) or [PageProperties](#) record. If the **MarginTop** property is not specified, the default value is used.

MarginTop = marginTopStart marginTopValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| marginTopStart | | | | | | | | | | marginTopValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

marginTopStart (1 byte): A byte that specifies the start of the **MarginTop** property. The value of the **marginTopStart** field MUST be one of the following values.

| Value | Parent |
|-------|--|
| 0x12 | PageLayout PageProperties |

marginTopValue (4 bytes): A floating point value that specifies the size, in millimeters, of the top margin of the page. The default value is 0x00000000 millimeters.

2.3.20 MarginLeft

The **MarginLeft** property specifies the size of the left margin of a page. If this property is specified, it MUST be part of a [PageLayout](#) or [PageProperties](#) record. If the **MarginLeft** property is not specified, the default value is used.

MarginLeft = marginLeftStart marginLeftValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|---|---|---|---|---|---|---|---|---|-----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| marginLeftStart | | | | | | | | | | marginLeftValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

marginLeftStart (1 byte): A byte that specifies the start of the **MarginLeft** property. The value of the **marginLeftStart** field MUST be one of the following values.

| Value | Parent |
|-------|--|
| 0x13 | PageLayout PageProperties |

marginLeftValue (4 bytes): A floating point value that specifies the size, in millimeters, of the left margin of the page. The default value is 0x00000000 millimeters.

2.3.21 MarginRight

The **MarginRight** property specifies the size of the right margin of a page. If this property is specified, it MUST be part of a [PageLayout](#) or [PageProperties](#) record. If the **MarginRight** property is not specified, the default value is used.

MarginRight = marginLeftStart marginLeftValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|---|---|---|---|---|---|---|---|---|------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| marginLeftStart | | | | | | | | | | marginRightValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

marginRightStart (1 byte): A byte that specifies the start of the **MarginRight** property. The value of **marginRightStart** field MUST be one of the following values.

| Value | Parent |
|-------|--|
| 0x15 | PageLayout PageProperties |

marginRightValue (4 bytes): A floating point value that specifies the size, in millimeters, of the right margin of the page. The default value is 0x00000000 millimeters.

2.3.22 MarginBottom

The **MarginBottom** property specifies the size of the bottom margin of a page. If this property is specified, it MUST be part of a [PageLayout](#) or [PageProperties](#) record. If the **MarginBottom** property is not specified, the default value is used.

MarginBottom = marginBottomStart marginBottomValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------|---|---|---|---|---|---|---|---|---|-------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| marginBottomStart | | | | | | | | | | marginBottomValue | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

marginBottomStart (1 byte): A byte that specifies the start of the **MarginBottom** property. The value of the **marginBottomStart** field MUST be one of the following values.

| Value | Parent |
|-------|--|
| 0x14 | PageLayout PageProperties |

marginBottomValue (4 bytes): A floating point value that specifies the size, in millimeters, of the bottom margin of the page. The default value is 0x00000000 millimeters.

2.3.23 Columns

The **Columns** property specifies the number of columns a page contains. If this property is specified, it MUST be part of a [PageProperties](#) record. If the **Columns** property is not specified, the default value is used.

Columns = columnsStart columnsValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------|---|---|---|---|---|---|---|---|---|--------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| columnsStart | | | | | | | | | | columnsValue | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

columnsStart (1 byte): A byte that specifies the start of the **Columns** property. The value of the **columnStart** field MUST be the following value.

| Value | Parent |
|-------|-----------------------|
| 0x17 | PageProperties |

columnsValue (4 bytes): A signed integer that specifies the number of columns a page contains. The default value is 0x00000000.

2.3.24 PageName (RPL 10.6)

Applies to RPL version 10.6 and later

The **PageName** property specifies the name of a page. If this property is specified, it **MUST** be part of a **PageLayout** record.

PageName = pageNameStart pageNameValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------|---|---|---|---|---|---|---|---|---|--------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|
| | | | | | | | | | | 1 | | | | | | | | | | | | | | | | | | | | 2 | | | | | | | | | | | | | | | | | | | | 3 | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| pageNameStart | | | | | | | | | | pageNameValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

pageNameStart (1 byte): A byte that specifies the start of the **PageName** property. The value of the **pageNameStart** field **MUST** be the following value.

| Value | Parent |
|-------|------------|
| 0x30 | PageLayout |

pageNameValue (variable): A String variable that specifies the name of a page.

2.3.25 Slant

The **Slant** property specifies the direction in which a [Line](#) record is drawn. If the **Slant** property is specified, it **MUST** be part of a **Line.ElementProperties.InlineSharedElementProperties** record. If the **Slant** property is not specified, the default value is used.

Slant = slantStart slantValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------|---|---|---|---|---|---|---|---|---|------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|--|--|--|---|--|--|--|--|--|--|--|--|--|--|
| | | | | | | | | | | 1 | | | | | | | | | | | | | | | | | | | | 2 | | | | | | | | | | | | | | | | | | | | 3 | | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | | | | | | | | | | | | | | | | |
| slantStart | | | | | | | | | | slantValue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

slantStart (1 byte): A byte that specifies the start of the **Slant** property. The value of the **slantStart** field **MUST** be the following value.

| Value | Parent |
|-------|---|
| 0x18 | ElementProperties.InlineSharedElementProperties for Line |

slantValue (1 byte): A byte that specifies how the line should be drawn. The default value of this field is 0x00. The value of the **slantValue** field MUST be one of the following values.

| Value | Meaning |
|-------|--|
| 0x00 | Draw from (left, bottom) to (right, top) |
| 0x01 | Draw from (left, top) to (right, bottom) |

2.3.26 CanGrow

The **CanGrow** property specifies if the height of a [RichTextBox](#) record can increase to accommodate its contents. If the **CanGrow** property is specified, it MUST be part of a [RichTextBox.ElementProperties.InlineSharedElementProperties](#) record. If the **CanGrow** property is not specified, the default value is used.

`CanGrow = canGrowStart canGrowValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------|---|---|---|---|---|---|---|---|---|--------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| canGrowStart | | | | | | | | | | canGrowValue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

canGrowStart (1 byte): A byte that specifies the start of the **CanGrow** property. The value of the **canGrowStart** field MUST be the following value.

| Value | Parent |
|-------|---|
| 0x19 | ElementProperties.InlineSharedElementProperties for RichTextBox |

canGrowValue (1 byte): A byte that specifies if the **RichTextBox** record height can increase. The default value of this field is 0x00. The value of the **canGrowValue** field MUST be one of the following values.

| Value | Meaning |
|-------|--------------------------------|
| 0x00 | False. The height cannot grow. |
| 0x01 | True. The height can grow. |

2.3.27 CanShrink

The **CanShrink** property specifies if the height of a [RichTextBox](#) record can decrease to accommodate its contents. If the **CanShrink** property is specified, it MUST be part of a [RichTextBox.ElementProperties.InlineSharedElementProperties](#) record. If the **CanShrink** property is not specified, the default value is used.

`CanShrink = canShrinkStart canShrinkValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| canShrinkStart | | | | | | | | | | canShrinkValue | | | | | | | | | | | | | | | | | | | | | |

canShrinkStart (1 byte): A byte that specifies the start of the **CanShrink** property. The value of the **canShrinkStart** field MUST be the following value.

| Value | Parent |
|-------|---|
| 0x1A | ElementProperties.InlineSharedElementProperties for RichTextBox |

canShrinkValue (1 byte): A byte that specifies if the **RichTextBox** record height can decrease. The default value of the **canShrinkValue** field is 0x00. The value of the **canShrinkValue** field MUST be one of the following values.

| Value | Meaning |
|-------|----------------------------------|
| 0x00 | False. The height cannot shrink. |
| 0x01 | True. The height can shrink. |

2.3.28 Value

The **Value** property specifies the value of a [RichTextBox](#) or [TextRun](#) record that is returned as a formatted **String**. If this property is specified, it MUST be part of a [RichTextBox.ElementProperties](#) or [TextRun.ElementProperties](#) record.

Value = valueStart valueValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------|---|---|---|---|---|---|---|---|---|-----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| valueStart | | | | | | | | | | valueValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

valueStart (1 byte): A byte that specifies the start of the **Value** property. The value of the **valueStart** field MUST be one of the following values. [<14>](#)

| Value | Parent |
|-------|---|
| 0x1B | ElementProperties.InlineSharedElementProperties for RichTextBox ElementProperties.NonSharedElementProperties for RichTextBox |
| 0x0A | ElementProperties.InlineSharedElementProperties for TextRun |

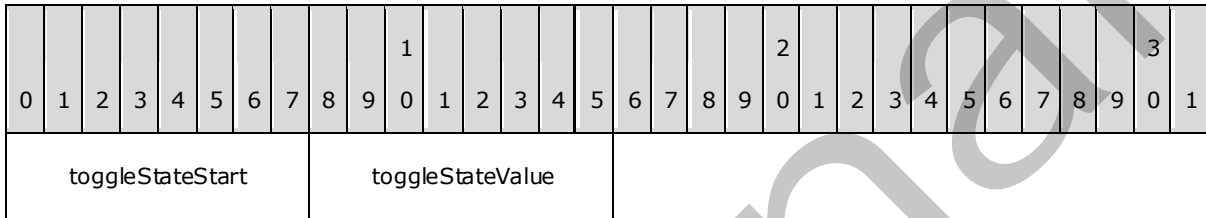
| Value | Parent |
|-------|--|
| | ElementProperties.NonSharedElementProperties for TextRun |

valueValue (variable): A **String** variable that specifies the formatted value of the **RichTextBox** or **TextRun**.

2.3.29 ToggleState

The **ToggleState** property specifies the current state of the toggle of a [RichTextBox](#) record. If the **ToggleState** is specified, it MUST be part of a [RichTextBox.ElementProperties.NonSharedElementProperties](#) record. If the **ToggleState** property is not specified, the default value is used.

`ToggleState = toggleStateStart toggleStateValue`



toggleStateStart (1 byte): A byte that specifies the start of the **ToggleState** property. The value of the **toggleStateStart** field MUST be the following value.

| Value | Parent |
|-------|--|
| 0x1C | ElementProperties.NonSharedElementProperties for RichTextBox |

toggleStateValue (1 byte): A byte that specifies the **toggle state** of the **RichTextBox** record. The default value is 0x00. The value of the **toggleStateValue** field MUST be one of the following values.

| Value | Meaning |
|-------|--------------------------|
| 0x00 | False. Toggle state off. |
| 0x01 | True. Toggle state on. |

2.3.30 CanSort

The **CanSort** property specifies if a [RichTextBox](#) record supports sorting. If the **CanSort** property is specified, it MUST be part of a [RichTextBox.ElementProperties.InlineSharedElementProperties](#) record. If the **CanSort** property is not specified, the default value is used.

`CanSort = canSortStart canSortValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------|---|---|---|---|---|---|---|---|---|--------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| canSortStart | | | | | | | | | | canSortValue | | | | | | | | | | | | | | | | | | | | | |

canSortStart (1 byte): A byte that specifies the start of **CanSort** property. The value of the **canSortStart** field MUST be the following value.

| Value | Parent |
|-------|---|
| 0x1D | ElementProperties.InlineSharedElementProperties for RichTextBox |

canSortValue (1 byte): A byte that specifies if the **RichTextBox** supports sorting. The default value is 0x00. The value of the **canSortValue** field MUST be one of the following values.

| Value | Meaning |
|-------|---------------------|
| 0x00 | False. Cannot sort. |
| 0x01 | True. Can sort. |

2.3.31 SortState

The **SortState** property specifies the **sort state** of a [RichTextBox](#) record. If the **SortState** property is specified, it MUST be part of a [RichTextBox.ElementProperties.NonSharedElementProperties](#) record. If the **SortState** property is not specified, the default value is used.

SortState = sortStateStart sortStateValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| sortStateStart | | | | | | | | | | sortStateValue | | | | | | | | | | | | | | | | | | | | | |

sortStateStart (1 byte): A byte that specifies the start of **SortState** property. The value of **sortStateStart** field MUST be the following value.

| Value | Parent |
|-------|--|
| 0x1E | ElementProperties.NonSharedElementProperties for RichTextBox |

sortStateValue (1 byte): A byte that specifies sort state of the **RichTextBox**. The default value of the **sortStateValue** field is 0x00. The list of acceptable values MUST be defined by the [SortOptions](#) enumeration.

2.3.32 Formula

The **Formula** property returns the formula used to generate a value for a [RichTextBox](#) or [TextRun](#) record. If the **Formula** property is specified, it MUST be part of a [RichTextBox.ElementProperties.InlineSharedElementProperties](#) or [TextRun.ElementProperties.InlineSharedElementProperties](#) record.

Formula = formulaStart formulaValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------|---|---|---|---|---|---|---|---|---|-------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|
| | | | | | | | | | | 1 | | | | | | | | | | | 2 | | | | | | | | | | | 3 | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| formulaStart | | | | | | | | | | formulaValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

formulaStart (1 byte): A byte that specifies the start of the **Formula** property. The value of **formulaStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x1F | ElementProperties.InlineSharedElementProperties for RichTextBox |
| 0x0C | ElementProperties.InlineSharedElementProperties for TextRun |

formulaValue (variable): A **String** value containing the formula that is used to generate the value for the **RichTextBox** or **TextRun**.

2.3.33 IsToggleParent

The **IsToggleParent** property specifies if a [RichTextBox](#) report item is used to toggle other items on a report. If this property is specified, it MUST be part of the [RichTextBox.ElementProperties](#) record. If the **IsToggleParent** property is not specified, the default value is used.

IsToggleParent = isToggleParentStart isToggleParentValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|---------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| | | | | | | | | | | 1 | | | | | | | | | | | | 2 | | | | | | | | | 3 | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| isToggleParentStart | | | | | | | | | | isToggleParentValue | | | | | | | | | | | | | | | | | | | | | | | |

isToggleParentStart (1 byte): A byte that specifies the start of the **IsToggleParent** property. The value of the **isToggleParentStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x20 | ElementProperties.InlineSharedElementProperties for RichTextBox ElementProperties.NonSharedElementProperties for RichTextBox |

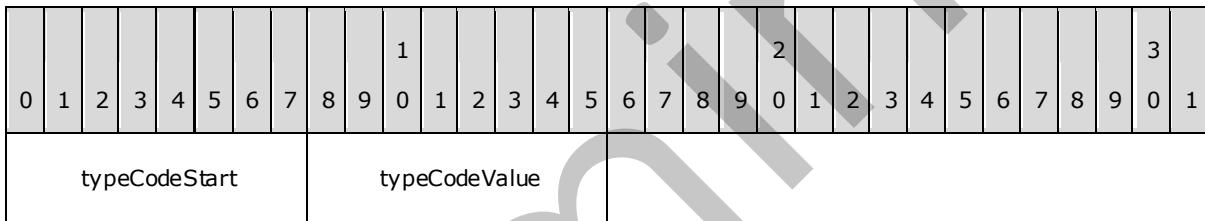
isToggleParentValue (1 byte): A byte that specifies if the **RichTextBox** is used as a toggle to control the visibility of another item on the report. The default value is 0x00. The value MUST be one of the following values.

| Value | Meaning |
|-------|--|
| 0x00 | False. The Textbox does not control the visibility of another item. |
| 0x01 | True. The Textbox controls the visibility of another item. |

2.3.34 TypeCode

The **TypeCode** property specifies the type of value for a [RichTextBox](#). If this property is specified, it MUST be part of the [RichTextBox.ElementProperties](#) record. If the **TypeCode** property is not specified, the default value is used.

TypeCode = typeCodeStart typeCodeValue



typeCodeStart (1 byte): A byte that specifies the start of the **TypeCode** property. [<16>](#) The value of the **typeCodeStart** field MUST be one of the following values.

| Value | Parent |
|-------|--|
| 0x21 | ElementProperties.InlineSharedElementProperties for RichTextBox . ElementProperties.InlineSharedElementProperties for PageFooterElement ElementProperties.NonSharedElementProperties for RichTextBox . |

typeCodeValue (1 byte): A byte that specifies the type of value for the **RichTextBox**. The default value is 0x11. The list of valid values MUST be obtained from the [TypeCode](#) enumeration. If a value that is not included in the enumeration is used, the value of the **TypeCode** enumeration is a **String**.

2.3.35 OriginalValue

The **OriginalValue** property specifies the unformatted value of a [RichTextBox](#) record. If the **OriginalValue** property is specified, it MUST be part of the [RichTextBox.ElementProperties.NonSharedElementProperties](#) record.

| Value | Meaning |
|-------|--|
| 0x00 | False. Multiple Paragraph records or one Paragraph with multiple TextRun records. |
| 0x01 | True. One Paragraph with only one TextRun record. |

2.3.37 ContentOffset

The **ContentOffset** property specifies the offset for the first paragraph in a [RichTextBox](#) record. The offset is relative to the top-left corner of the **RichTextBox** when the paragraphs are rendered top to bottom. The offset is relative to the top-right corner of the **RichTextBox** when paragraphs are rendered right to left.

If the **ContentOffset** property is specified, it MUST be part of the [RichTextBox.ElementProperties.NonSharedElementProperties](#) record. If the **ContentOffset** property is not specified, the default value is used.

`ContentOffset = contentOffsetStart contentOffsetValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------|---|---|---|---|---|---|---|---|---|--------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| contentOffsetStart | | | | | | | | | | contentOffsetValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

contentOffsetStart (1 byte): A byte that specifies the start of the **ContentOffset** property. The value of **contentOffsetStart** field MUST be the following value.

| Value | Parent |
|-------|--|
| 0x25 | ElementProperties.NonSharedElementProperties for RichTextBox |

contentOffsetValue (4 bytes): A floating point value that specifies the offset for the first paragraph. The default value is 0x00000000.

2.3.38 StreamName

The **StreamName** property specifies the name of a stream for a [Chart](#), [Map](#), or [GaugePanel](#) record. If this property is specified, it MUST be part of a [Chart.ElementProperties.NonSharedElementProperties](#) or **Map.ElementProperties.NonSharedElementProperties** or **GaugePanel.ElementProperties.NonSharedElementProperties** record.

`StreamName = streamNameStart streamNameValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|---|---|---|---|---|---|---|---|---|----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| streamNameStart | | | | | | | | | | streamNameValue (variable) | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

streamNameStart (1 byte): A byte that specifies the start of the **StreamName** property. The value of **streamNameStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x28 | ElementProperties.NonSharedElementProperties for Chart . ElementProperties.InlineSharedElementProperties for PageFooterElement ElementProperties.NonSharedElementProperties for Map . ElementProperties.NonSharedElementProperties for GaugePanel . |

streamNameValue (variable): A **String** variable that specifies the **stream name** of the **Chart**, **Map**, and **GaugePanel** report items. <18>

2.3.39 Sizing

The **Sizing** property specifies the behavior of an image if the actual image does not fit within the specified size of an [Image](#) record. If this property is specified, it MUST be part of the [Image.ElementProperties.InlineSharedElementProperties](#) record. If this property is not specified, the default value is used.

Sizing = sizingStart sizingValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|---|---|---|---|---|---|---|---|---|-------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| sizingStart | | | | | | | | | | sizingValue | | | | | | | | | | | | | | | | | | | | | |

sizingStart (1 byte): A byte that specifies the start of the **Sizing** property. The value of **sizingStart** field MUST be the following value.

| Value | Parent |
|-------|---|
| 0x29 | ElementProperties.InlineSharedElementProperties for Image |

sizingValue (1 byte): A byte that specifies the behavior if the actual image does not fit in the specified size. The list of valid values MUST be obtained from the [Sizings](#) enumeration. The default value is 0x00.

2.3.40 LinkToChild

The **LinkToChild** property specifies the **ID** of a report item contained directly within a rectangle. If the **LinkToChild** property is specified, it MUST be part of the [Rectangle.ElementProperties.InlineSharedElementProperties](#) record.

LinkToChild = linkToChildStart linkToChildValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|-----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|--|--|---|--|--|
| | | | | | | | | | | 1 | | | | | | | | | | | | 2 | | | | | | | | | | | | | | | | | 3 | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | | | | | | | |
| linkToChildStart | | | | | | | | | | linkToChildValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

linkToChildStart (1 byte): A byte that specifies the start of the **LinkToChild** property. The value of **linkToChildStart** field MUST be the following value.

| Value | Parent |
|-------|---|
| 0x2B | ElementProperties.InlineSharedElementProperties of Rectangle |

linkToChildValue (variable): A **String** variable that specifies the **ID** of a child report item contained directly within a **Rectangle** that is the target location for the **document map** label (if any).

2.3.41 PrintOnFirstPage

The **PrintOnFirstPage** property specifies if the content of a header or footer should be printed on the first page of a report. If this property is specified, it MUST be part of the [PageHeader.ElementProperties.InlineSharedElementProperties](#) or [PageFooter.ElementProperties.InlineSharedElementProperties](#) record. If the **PrintOnFirstPage** property is not specified, the default value is used.

PrintOnFirstPage = printOnFirstPageStart printOnFirstPageValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------------|---|---|---|---|---|---|---|---|---|-----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|--|--|---|--|--|--|
| | | | | | | | | | | 1 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 3 | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | | | | | | | | |
| printOnFirstPageStart | | | | | | | | | | printOnFirstPageValue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

printOnFirstPageStart (1 byte): A byte that specifies the start of **PrintOnFirstPage** property. The value of the **printOnFirstPageStart** field MUST be one of the following values.

| Value | Parent |
|-------|--|
| 0x2C | ElementProperties.InlineSharedElementProperties for PageHeaderElement |

printOnFirstPageValue (1 byte): A byte that specifies whether the content should be printed on the first page of the [Report](#) record. The default value is 0x00. The value of the **printOnFirstPageValue** field **MUST** be one of the following values.

| Value | Meaning |
|-------|------------------------------------|
| 0x00 | False. Do not print on first page. |
| 0x01 | True. Print on first page. |

2.3.42 PrintBetweenSections (RPL 10.4)

Applies to RPL 10.4 and later

The **PrintBetweenSections** property specifies if the content of a header or footer should be printed between sections of a report. If the **PrintBetweenSections** property is specified, it **MUST** be part of a [PageHeaderElement.ElementProperties.InlineSharedElementProperties](#) or [PageFooterElement.ElementProperties.InlineSharedElementProperties](#) record. If the **PrintBetweenSections** property is not specified, the default value is used.

```
PrintBetweenSections = printBetweenSectionsStart printBetweenSectionsValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|---|--|--|
| | | | | | | | | | | 1 | | | | | | | | | | | | 2 | | | | | | | | | | | | | | | 3 | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | | | | | |
| printBetweenSectionsStart | | | | | | | | | | printBetweenSectionsValue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

printBetweenSectionsStart (1 byte): A byte that specifies the start of the **PrintBetweenSections** property. The value of the **printBetweenSectionsStart** field **MUST** be one of the following values.

| Value | Parent |
|-------|--|
| 0x2F | ElementProperties.InlineSharedElementProperties for PageHeaderElement ElementProperties.InlineSharedElementProperties for PageFooterElement |

printBetweenSectionsValue (1 byte): A byte that specifies whether the content should be printed between sections of the [Report](#) record. The default value is 0x00. The value of the **printBetweenSectionsValue** field **MUST** be one of the following values.

| Value | Meaning |
|-------|---------------------------------------|
| 0x00 | False. Do not print between sections. |
| 0x01 | True. Print between sections. |

2.3.43 FormattedValueExpressionBased

The **FormattedValueExpressionBased** property specifies that the [Value](#) property of a [RichTextBox](#) record was generated from expression-based properties (that is, either the value or one of the formatting style properties was an expression). The formatting style properties are: [Format](#), [Language](#), [Calendar](#), [NumeralLanguage](#), and [NumeralVariant](#).

If the **FormattedValueExpressionBased** property is specified, it MUST be part of the [RichTextBox.ElementProperties.InlineSharedElementProperties](#) record. If the **FormattedValueExpressionBased** property is not specified, the default value is used.

```
FormattedValueExpressionBased = formattedValueExpressionBasedStart
formattedValueExpressionBasedValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------------------------|---|---|---|---|---|---|---|---|---|------------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| formattedValueExpressionBasedStart | | | | | | | | | | formattedValueExpressionBasedValue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

formattedValueExpressionBasedStart (1 byte): A byte that specifies the start of the **FormattedValueExpressionBased** property. The value of the **formattedValueExpressionBasedStart** field MUST be the following value.

| Value | Parent |
|-------|---|
| 0x2D | ElementProperties.InlineSharedElementProperties for RichTextBox . |

formattedValueExpressionBasedValue (1 byte): A byte that specifies that the **Value** property of the **RichTextBox** record was generated from expression-based properties. The default value is 0x00. The value of the **formattedValueExpressionBasedValue** field MUST be one of the following values.

| Value | Meaning |
|-------|---------|
| 0x00 | False. |
| 0x01 | True. |

2.3.44 ProcessedWithError

The **ProcessedWithError** property specifies if an error was encountered during generation of the [Value](#) property for a text box or text run. If the **ProcessedWithError** property is specified, it MUST be part of a [RichTextBox.ElementProperties.NonSharedElementProperties](#) or **TextRun.ElementProperties.NonSharedElementProperties** record. If the **ProcessedWithError** property is not specified, the default value is used.

```
ProcessedWithError = processedWithErrorStart processedWithErrorValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------------|---|---|---|---|---|---|---|---|---|-------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| processedWithErrorStart | | | | | | | | | | processedWithErrorValue | | | | | | | | | | | | | | | | | | | | | |

processedWithErrorStart (1 byte): A byte that specifies the start of **ProcessedWithError** property. The value of the **processedWithErrorStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x2E | ElementProperties.NonSharedElementProperties for RichTextBox . |
| 0x0D | ElementProperties.NonSharedElementProperties for TextRun . |

processedWithErrorValue (1 byte): A byte that specifies if an error was encountered during generation of the **Value** property. The default value is 0x00. The value of the **processedWithErrorValue** field MUST be one of the following values.

| Value | Meaning |
|-------|--|
| 0x00 | False. Value property was processed without error. |
| 0x01 | True. Value property was processed with error and its valueValue field should be String "#Error". |

2.3.45 ImageMIMETYPE

The **ImageMIMETYPE** property specifies the **MIME type** for an image. If the **ImageMIMETYPE** property is specified, it MUST be part of [ImageDataProperties.InlineSharedImageDataProperties](#) or [ImageDataProperties.NonSharedImageDataProperties](#) record.

ImageMIMETYPE = imageMIMETYPEStart imageMIMETYPEValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------|---|---|---|---|---|---|---|---|---|-------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| imageMIMETYPEStart | | | | | | | | | | imageMIMETYPEValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

imageMIMETYPEStart (1 byte): A byte that specifies the start of the **ImageMIMETYPE** property. [<19>](#) The value of the **imageMIMETYPEStart** field MUST be one of the following values.

| Value | Parent |
|-------|--|
| 0x00 | ImageDataProperties.InlineSharedImageDataProperties |

| | |
|-----|--|
| ... | |
|-----|--|

widthStart (1 byte): A byte that specifies the start of **Width** property. <22> The value of the **widthStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x03 | ImageDataProperties.InlineSharedImageDataProperties ImageDataProperties.NonSharedImageDataProperties |

widthValue (4 bytes): A signed integer that specifies the width of an image in pixels. If this property is present, the [Height](#), [HorizontalResolution](#), [VerticalResolution](#), and [RawFormat](#) properties MUST be present in the same parent record. <23>

2.3.48 Height

The **Height** property specifies the height of an image in pixels. If this property is specified, it MUST be part of an [ImageDataProperties.InlineSharedImageDataProperties](#) or [ImageDataProperties.NonSharedImageDataProperties](#) record.

Height = heightStart heightValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|---|---|---|---|---|---|---|---|---|-------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| heightStart | | | | | | | | | | heightValue | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

heightStart (1 byte): A byte that specifies the start of the **Height** property. <24> The value of the **heightStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x04 | ImageDataProperties.InlineSharedImageDataProperties ImageDataProperties.NonSharedImageDataProperties |

heightValue (4 bytes): A signed integer that specifies the height of an image in pixels. If this property is present, the [Width](#), [HorizontalResolution](#), [VerticalResolution](#), and [RawFormat](#) properties MUST be present in the same parent record. <25>

2.3.49 HorizontalResolution

The **HorizontalResolution** property specifies the **horizontal resolution**, in pixels per inch, of an image. If this property is specified, it MUST be part of the [ImageDataProperties.InlineSharedImageDataProperties](#) or [ImageDataProperties.NonSharedImageDataProperties](#) record.

HorizontalResolution = horizontalResolutionStart horizontalResolutionValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| horizontalResolutionStart | | | | | | | | | | horizontalResolutionValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

horizontalResolutionStart (1 byte): A byte that specifies the start of the **HorizontalResolution** property. <26> The value of the **horizontalResolutionStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x05 | ImageDataProperties.InlineSharedImageDataProperties ImageDataProperties.NonSharedImageDataProperties |

horizontalResolutionValue (4 bytes): A floating point value that specifies the horizontal resolution of an image. If this property is present, the [Width](#), [Height](#), [VerticalResolution](#), and [RawFormat](#) properties MUST be present in the same parent record. <27>

2.3.50 VerticalResolution

The **VerticalResolution** property specifies the **vertical resolution**, in pixels per inch, of an image. If this property is specified, it MUST be part of an [ImageDataProperties.InlineSharedImageDataProperties](#) or [ImageDataProperties.NonSharedImageDataProperties](#) record.

VerticalResolution = verticalResolutionStart verticalResolutionValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------------|---|---|---|---|---|---|---|---|---|-------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| verticalResolutionStart | | | | | | | | | | verticalResolutionValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

verticalResolutionStart (1 byte): A byte that specifies the start of the **VerticalResolution** property. <28> The value of the **verticalResolutionStart** field MUST be one of the following values.

| Value | Parent |
|-------|--|
| 0x06 | ImageDataProperties.InlineSharedImageDataProperties |

verticalResolutionValue (4 bytes): A floating point value that specifies the vertical resolution of an image. If this property is present, the [Width](#), [Height](#), [HorizontalResolution](#), and [RawFormat](#) properties MUST be present in the same parent record. [<29>](#)

2.3.51 RawFormat

The **RawFormat** property specifies the stream format of an image. If this property is specified, it MUST be part of the [ImageDataProperties.InlineSharedImageDataProperties](#) or [ImageDataProperties.NonSharedImageDataProperties](#) record.

RawFormat = rawFormatStart rawFormatValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| rawFormatStart | | | | | | | | | | rawFormatValue | | | | | | | | | | | | | | | | | | | | | |

rawFormatStart (1 byte): A byte that specifies the start of **RawFormat** property. [<30>](#) The value of the **rawFormatStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x07 | ImageDataProperties.InlineSharedImageDataProperties ImageDataProperties.NonSharedImageDataProperties |

rawFormatValue (1 byte): A byte that specifies the stream format of the image. If this property is present, the [Width](#), [Height](#), [HorizontalResolution](#), and [VerticalResolution](#) properties MUST be present in the same parent record. [<31>](#) The value of the **rawFormatValue** field MUST be a value from the [ImageRawFormat](#) enumeration.

2.3.52 Hyperlink

The **Hyperlink** property specifies the URL of a hyperlink. If this property is specified, it MUST be part of an [Action](#) record.

Hyperlink = hyperlinkStart hyperlinkValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| hyperlinkStart | | | | | | | | | | hyperlinkValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

hyperlinkStart (1 byte): A byte that specifies the start of the **Hyperlink** property. The value of the **hyperlinkStart** field **MUST** be the following value.

| Value | Parent |
|-------|---------------|
| 0x06 | Action |

hyperlinkValue (variable): A **String** variable that specifies the URL of a hyperlink.

2.3.53 BookmarkLink

The **BookmarkLink** property specifies the **ID** property of a bookmark located within a report that is the target of this action. If the **BookmarkLink** property is specified, it **MUST** be part of an [Action](#) record.

```
BookmarkLink = bookmarkLinkStart bookmarkLinkValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------|---|---|---|---|---|---|---|---|---|------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| bookmarkLinkStart | | | | | | | | | | bookmarkLinkValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

bookmarkLinkStart (1 byte): A byte that specifies the start of the **BookmarkLink** property. The value of **bookmarkLinkStart** field **MUST** be the following value.

| Value | Parent |
|-------|---------------|
| 0x07 | Action |

bookmarkLinkValue (variable): A **String** value containing the **ID** of a bookmark located within a report that is the target of this action.

2.3.54 DrillthroughId

The **DrillthroughId** property specifies the unique identifier of the **drillthrough report** to be executed and viewed as the target of an action specified by an [Action](#) record. If the **DrillthroughId** property is specified, it **MUST** be part of an **Action** record.

```
DrillthroughId = drillthroughIdStart drillthroughIdValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|--------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| drillthroughIdStart | | | | | | | | | | drillthroughIdValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

drillthroughIdStart (1 byte): A byte that specifies the start of the **DrillthroughId** property. The value of the **drillthroughStart** field MUST be the following value.

| Value | Parent |
|-------|--------|
| 0x08 | Action |

drillthroughIdValue (variable): A **String** variable that specifies the unique identifier of the drillthrough report to be executed and viewed as the target of this action.

2.3.55 DrillthroughUrl

The **DrillthroughUrl** property specifies the URL of the drillthrough report to be executed and viewed as the target of an action specified by the [Action](#) record. If the **DrillthroughUrl** property is specified, it MUST be part of an **Action** record.

`DrillthroughUrl = drillthroughUrlStart drillthroughUrlValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| drillthroughUrlStart | | | | | | | | | | drillthroughUrlValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

drillthroughUrlStart (1 byte): A byte that specifies the start of **DrillthroughUrl** property. The value of **drillthroughStart** field MUST be the following value.

| Value | Parent |
|-------|--------|
| 0x09 | Action |

drillthroughUrlValue (variable): A **String** variable that specifies the URL of the drillthrough report to be executed and viewed as the target of this action.

2.3.56 BorderColor

The **BorderColor** property specifies the color of a border. If this property is specified, it MUST be part of the [Style](#) record. The value of the **BorderColor** property is overridden by the value of the

[BorderColorLeft](#), [BorderColorRight](#), [BorderColorTop](#), or [BorderColorBottom](#) property for individual edges. If the **BorderColor** property is not specified, the default value is used.

`BorderColor = borderColorStart borderColorValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|-----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| borderColorStart | | | | | | | | | | borderColorValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

borderColorStart (1 byte): A byte that specifies the start of the **BorderColor** property. The value of the **borderColorStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x00 | SharedStyleProperties NonSharedStyleProperties |

borderColorValue (variable): A **String** variable that specifies the border color. A color value can be a color name (user-defined system colors are not supported), a Red Green Blue (RGB) color **String** of the form #HHHHHH, or an Alpha Red Green Blue (ARGB) color **String** of the form #HHHHHHHH. The default value is "Black".

2.3.57 BorderColorLeft

The **BorderColorLeft** property specifies the color of the left border. If this property is specified, it MUST be part of a [Style](#) record. The value of the **BorderColorLeft** property overrides the [BorderColor](#) property for the left border. If the **BorderColorLeft** property is not defined, the value of **BorderColor** is used.

`BorderColorLeft = borderColorLeftStart borderColorLeftValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| borderColorLeftStart | | | | | | | | | | borderColorLeftValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

borderColorLeftStart (1 byte): A byte that specifies the start of the **BorderColorLeft** property. The value of the **borderColorLeftStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x01 | SharedStyleProperties NonSharedStyleProperties |

borderColorLeftValue (variable): A **String** variable that specifies the left border color. A color value can be a color name (user-defined system colors are not supported), an RGB color **String** of the form #HHHHHH, or an ARGB color **String** of the form #HHHHHHHH.

2.3.58 borderColorRight

The **BorderColorRight** property specifies the color of the right border. If this property is specified, it MUST be part of a [Style](#) record. The value of the **BorderColorRight** property overrides the [BorderColor](#) property for the right border. If the **BorderColorRight** property is not defined, the value of **BorderColor** is used.

```
BorderColorRight = borderColorRightStart borderColorRightValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------------|---|---|---|---|---|---|---|---|---|---|----------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|--|--|--|--|--|--|--|--|---|--|--|--|--|
| | | | | | | | | | | | 1 | | | | | | | | | | | | | | | | | | 2 | | | | | | | | | | | | | | | | 3 | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | | | | | | | | | | | | | | | |
| borderColorRightStart | | | | | | | | | | | borderColorRightValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

borderColorRightStart (1 byte): A byte that specifies the start of the **BorderColorRight** property. The value of **borderColorRightStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x02 | SharedStyleProperties NonSharedStyleProperties |

borderColorRightValue (variable): A **String** variable that specifies the color of the right border. A color value can be a color name (user defined system colors are not supported), an RGB color **String** of the form #HHHHHH, or an ARGB color **String** of the form #HHHHHHHH.

2.3.59 borderColorTop

The **BorderColorTop** property specifies the color of the top border. If this property is specified, it MUST be part of a [Style](#) record. The value of the **BorderColorTop** property overrides the [BorderColor](#) property for the top border. If the **BorderColorTop** property is not defined, the value of **BorderColor** is used.

```
BorderColorTop = borderColorTopStart borderColorTopValue
```


| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|--------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| borderColorTopStart | | | | | | | | | | borderColorTopValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

borderColorTopStart (1 byte): A byte that specifies the start of the **BorderColorTop** property. The value of the **borderColorTopStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x03 | SharedStyleProperties NonSharedStyleProperties |

borderColorTopValue (variable): A **String** variable that specifies the color of the top border. A color value can be a color name (user-defined system colors are not supported), an RGB color **String** of the form #HHHHHH, or an ARGB color **String** of the form #HHHHHHHH.

2.3.60 BorderColorBottom

The **BorderColorBottom** property specifies the color of the bottom border. If this property is specified, it MUST be part of a [Style](#) record. The value of the **BorderColorBottom** property overrides the [BorderColor](#) property for the bottom border. If the **BorderColorBottom** property is not defined, the value of **BorderColor** is used.

```
BorderColorBottom = borderColorBottomStart borderColorBottomValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------------|---|---|---|---|---|---|---|---|---|-----------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| | | | | | | | | | | 1 | | | | | | | | | | | 2 | | | | | | | | | 3 | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| borderColorBottomStart | | | | | | | | | | borderColorBottomValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

borderColorBottomStart (1 byte): A byte that specifies the start of the **BorderColorBottom** property. The value of the **borderColorBottomStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x04 | SharedStyleProperties NonSharedStyleProperties |

borderColorBottomValue (variable): A **String** variable that specifies the color of the bottom border. A color value can be a color name (user-defined system colors are not supported), an RGB color **String** of the form #HHHHHH, or an ARGB color **String** of the form #HHHHHHHH.

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|---|--|
| | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | | | | | 3 | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | |
| borderStyleTopStart | | | | | | | | | borderStyleTopValue | | | | | | | | | | | | | | | | | | | | | | | | | | |

borderStyleTopStart (1 byte): A byte that specifies the start of the **BorderStyleTop** property. The value of **borderStyleTopStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x08 | SharedStyleProperties NonSharedStyleProperties |

borderStyleTopValue (1 byte): A byte that specifies the style of the top border. The value of the **borderStyleTopValue** field is of type [BorderStyle](#) and it MUST be contained in the **BorderStyles** enumeration. <35>

2.3.65 BorderStyleBottom

The **BorderStyleBottom** property specifies the style of the bottom border. If this property is specified, it MUST be part of a [Style](#) record. The value of this property overrides the [BorderStyle](#) property for the bottom border. If the **BorderStyleBottom** property is not defined, the value of **BorderStyle** is used.

`BorderStyleBottom = borderStyleBottomStart borderStyleBottomValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------------|---|---|---|---|---|---|---|---|------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|---|--|
| | | | | | | | | | | 1 | | | | | | | | | | | 2 | | | | | | | | | | | | 3 | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | |
| borderStyleBottomStart | | | | | | | | | borderStyleBottomValue | | | | | | | | | | | | | | | | | | | | | | | | | |

borderStyleBottomStart (1 byte): A byte that specifies the start of the **BorderStyleBottom** property. The value of the **borderStyleBottomStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x09 | SharedStyleProperties NonSharedStyleProperties |

borderStyleBottomValue (1 byte): A byte that specifies the style of the bottom border. The value of this field is of type [BorderStyle](#) and it MUST be contained in the enumeration. <36>

2.3.66 BorderWidth

The **BorderWidth** property specifies the width of a border. If this property is specified, it MUST be part of a [Style](#) record. The value of the **BorderWidth** property is overridden by the value of the

[BorderWidthLeft](#) for the left border, [BorderWidthRight](#) for the right border, [BorderWidthTop](#) for the top border, or [BorderWidthBottom](#) for the bottom border. If the **BorderWidth** property is not specified, the default value is used.

`BorderWidth = borderWidthStart borderWidthValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|-----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| borderWidthStart | | | | | | | | | | borderWidthValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

borderWidthStart (1 byte): A byte that specifies the start of the **BorderWidth** property. The value of the **borderWidthStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x0A | SharedStyleProperties NonSharedStyleProperties |

borderWidthValue (variable): An [RPLSize](#) variable that specifies the border width. The default value is "0pt".

2.3.67 BorderWidthLeft

The **BorderWidthLeft** property specifies the width of the left border. If this property is specified, it MUST be part of a [Style](#) record. The value of the **BorderWidthLeft** property overrides the [BorderWidth](#) property for the left border. If the **BorderWidthLeft** property is not defined, the value of **BorderWidth** is used.

`BorderWidthLeft = borderWidthLeftStart borderWidthLeftValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| borderWidthLeftStart | | | | | | | | | | borderWidthLeftValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

borderWidthLeftStart (1 byte): A byte that specifies the start of the **BorderWidthLeft** property. The value of the **borderWidthLeftStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x0B | SharedStyleProperties NonSharedStyleProperties |

borderWidthLeftValue (variable): An [RPLSize](#) variable that specifies the width of the left border.

2.3.68 BorderWidthRight

The **BorderWidthRight** property specifies the width of the right border. If this property is specified, it MUST be part of a [Style](#) record. The value of the **BorderWidthRight** property overrides the [BorderWidth](#) property for the right border. If the **BorderWidthRight** property is not defined, the value of **BorderWidth** is used.

```
BorderWidthRight = borderWidthRightStart borderWidthRightValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------------|---|---|---|---|---|---|---|---|---|----------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| borderWidthRightStart | | | | | | | | | | borderWidthRightValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

borderWidthRightStart (1 byte): A byte that specifies the start of the **BorderWidthRight** property. The value of the **borderWidthRightStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x0C | SharedStyleProperties NonSharedStyleProperties |

borderWidthRightValue (variable): An [RPLSize](#) variable that specifies the width of the right border.

2.3.69 BorderWidthTop

The **BorderWidthTop** property specifies the width of the top border. If this property is specified, it MUST be part of a [Style](#) record. The value of the **BorderWidthTop** property overrides the [BorderWidth](#) property for the top border. If the **BorderWidthTop** property is not defined, the value of **BorderWidth** is used.

```
BorderWidthTop = borderWidthTopStart borderWidthTopValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|--------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| borderWidthTopStart | | | | | | | | | | borderWidthTopValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

borderWidthTopStart (1 byte): A byte that specifies the start of the **BorderWidthTop** property. The value of the **borderWidthTopStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x0D | SharedStyleProperties NonSharedStyleProperties |

borderWidthTopValue (variable): An [RPLSize](#) variable that specifies the width of the top border.

2.3.70 BorderWidthBottom

The **BorderWidthBottom** property specifies the width of the bottom border. If this property is specified, it MUST be part of a [Style](#) record. The value of the **BorderWidthBottom** property overrides the [BorderWidth](#) property for the bottom border. If the **BorderWidthBottom** property is not defined, the value of **BorderWidth** is used.

BorderWidthBottom = borderWidthBottomStart borderWidthBottomValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------------|---|---|---|---|---|---|---|---|---|-----------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| borderWidthBottomStart | | | | | | | | | | borderWidthBottomValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

borderWidthBottomStart (1 byte): A byte that specifies the start of the **BorderWidthBottom** property. The value of the **borderWidthBottomStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x0E | SharedStyleProperties NonSharedStyleProperties |

borderWidthBottomValue (variable): An [RPLSize](#) variable that specifies the width of the bottom border.

2.3.71 PaddingLeft

The **PaddingLeft** property specifies the padding between the left edge of the **ancestor** record the **Style** record that contains this property applies to and its contents. If the **PaddingLeft** property is specified, it **MUST** be part of a **Style** record. If the **PaddingLeft** property is not specified, the default value is used.

```
PaddingLeft = paddingLeftStart paddingLeftValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|-----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| paddingLeftStart | | | | | | | | | | paddingLeftValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

paddingLeftStart (1 byte): A byte that specifies the start of the **PaddingLeft** property. The value of the **paddingLeftStart** field **MUST** be one of the following values.

| Value | Parent |
|-------|---|
| 0x0F | SharedStyleProperties NonSharedStyleProperties |

paddingLeftValue (variable): An [RPLSize](#) variable that specifies the padding between the left edge and its contents. The default value is "0pt".

2.3.72 PaddingRight

The **PaddingRight** property specifies the padding between the right edge of the ancestor record the **Style** record that contains this property applies to and its contents. If this property is specified, it **MUST** be part of the **Style** record. If the **PaddingRight** property is not specified, the default value is used.

```
PaddingRight = paddingRightStart paddingRightValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------|---|---|---|---|---|---|---|---|---|------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| paddingRightStart | | | | | | | | | | paddingRightValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

paddingRightStart (1 byte): A byte that specifies the start of the **PaddingRight** property. The value of the **paddingRightStart** field **MUST** be one of the following values.

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| textAlignStart | | | | | | | | | | textAlignValue | | | | | | | | | | | | | | | | | | | | | |

textAlignStart (1 byte): A byte that specifies the start of the **TextAlign** property. The value of the **textAlignStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x19 | SharedStyleProperties NonSharedStyleProperties |

textAlignValue (1 byte): A byte that specifies the horizontal alignment of text. The value of the **textAlignValue** field is of type [TextAlignments](#) and it MUST be contained in the **TextAlignments** enumeration. [<40>](#) The default value is 0x00.

2.3.82 VerticalAlign

The **VerticalAlign** property specifies the vertical alignment for a contiguous set of characters. If this property is specified, it MUST be part of a [Style](#) record. If the **VerticalAlign** property is not specified, the default value is used.

VerticalAlign = verticalAlignStart verticalAlignValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------|---|---|---|---|---|---|---|---|---|--------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| verticalAlignStart | | | | | | | | | | verticalAlignValue | | | | | | | | | | | | | | | | | | | | | |

verticalAlignStart (1 byte): A byte that specifies the start of **VerticalAlign** structure. The value of the **verticalAlignStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x1A | SharedStyleProperties NonSharedStyleProperties |

verticalAlignValue (1 byte): A byte that specifies the vertical alignment of text. The value of this field is of type [VerticalAlignments](#) and it MUST be contained in the **VerticalAlignments** enumeration. [<41>](#) The default value is 0x00.

2.3.83 Color

The **Color** property specifies the foreground color. If this property is specified, it MUST be part of a [Style](#) record. If the **Color** property is not specified, the default value is used.

writingModeValue (1 byte): A byte that specifies whether text is written horizontally or vertically. The value of this field is of type [WritingModes](#) and it MUST be contained in the **WritingModes** enumeration. <44> The default value is 0x00.

2.3.87 UnicodeBiDi

The **UnicodeBiDi** property specifies the level of embedding with respect to the bidirectional algorithm. If this property is specified, it MUST be part of a [Style](#) record. If the **UnicodeBiDi** property is not specified, the default value is used.

UnicodeBiDi = unicodeBiDiStart unicodeBiDiValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | 1 | | | | | | | | | | | 2 | | | | | | | | | 3 | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| unicodeBiDiStart | | | | | | | | | unicodeBiDiValue | | | | | | | | | | | | | | | | | | | | | | |

unicodeBiDiStart (1 byte): A byte that specifies the start of the **UnicodeBiDi** property. The value of the **unicodeBiDiStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x1F | SharedStyleProperties NonSharedStyleProperties |

unicodeBiDiValue (1 byte): A byte that specifies the level of embedding with respect to the bidirectional algorithm. The value of this field is of type [UnicodeBiDiTypes](#) and it MUST be contained in the **UnicodeBiDiTypes** enumeration. <45> The default value is 0x00.

2.3.88 BackgroundImage

The **BackgroundImage** property specifies the image to use as a background. If this property is specified, it MUST be part of a [Style](#) record.

BackgroundImage = backgroundImageStart backgroundImageValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|
| | | | | | | | | | 1 | | | | | | | | | | | | 2 | | | | | | | | | 3 | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | |
| backgroundImageStart | | | | | | | | | backgroundImageValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

backgroundImageStart (1 byte): A byte that specifies the start of the **BackgroundImage** property. The value of the **backgroundImageStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x21 | SharedStyleProperties NonSharedStyleProperties |

backgroundImageValue (variable): An [ImageDataProperties](#) record that specifies the background image of a structure.

2.3.89 BackgroundColor

The **BackgroundColor** property specifies the color to use for the background. If this property is specified, it MUST be part of a [Style](#) record. If the **BackgroundColor** property is not specified, the background color MUST be interpreted as being transparent.

`BackgroundColor = backgroundColorStart backgroundColorValue`

| 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | |
| backgroundColorStart | | | | | | | | | | backgroundColorValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

backgroundColorStart (1 byte): A byte that specifies the start of the **BackgroundColor** property. The value of **backgroundColorStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x22 | SharedStyleProperties NonSharedStyleProperties |

backgroundColorValue (variable): A **String** variable that specifies the background color. A background color value can be a color name (user-defined system colors are not supported), an RGB color **String** of the form #HHHHHH, or an ARGB color **String** of the form #HHHHHHHH. [<46>](#)

2.3.90 BackgroundRepeat

The **BackgroundRepeat** property specifies how a background image should occupy the available space within its parent container. If this property is specified, it MUST be part of a [Style](#) record. If the **BackgroundRepeat** property is not specified, the default value is used.

`BackgroundRepeat = backgroundRepeatStart backgroundRepeatValue`

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------------|---|---|---|---|---|---|---|---|---|-----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| backgroundRepeatStart | | | | | | | | | | backgroundRepeatValue | | | | | | | | | | | | | | | | | | | | | |

backgroundRepeatStart (1 byte): A byte that specifies the start of **BackgroundRepeat** property. The value of the **backgroundRepeatStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x23 | SharedStyleProperties NonSharedStyleProperties |

backgroundRepeatValue (1 byte): A byte that specifies how the background image should occupy the available space. The value of this field is of type [BackgroundRepeatTypes](#) and it MUST be contained in the **BackgroundRepeatTypes** enumeration. <47> The default value is 0x00.

2.3.91 NumeralLanguage

The **NumeralLanguage** property specifies the digit format to use as determined by a language. If this property is specified, it MUST be part of a [Style](#) record. If the **NumeralLanguage** property is not specified, its value MUST be the same as the peer [Language](#) style property.

NumeralLanguage = numeralLanguageStart numeralLanguageValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|---------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| numeralLanguageStart | | | | | | | | | | numeralLanguageValue (variable) | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

numeralLanguageStart (1 byte): A byte that specifies the start of the **NumeralLanguage** property. The value of **numeralLanguageStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x24 | SharedStyleProperties NonSharedStyleProperties |

numeralLanguageValue (variable): A **String** variable that specifies the digit format to use as determined by a language. The value MUST be a valid language code, which is a combination of an ISO 639 two-letter lowercase culture code associated with a language, and an ISO 3166 (see [ISO-3166](#)) two-letter uppercase subculture code associated with a country or region.

2.3.92 NumeralVariant

The **NumeralVariant** property specifies the variant of the digit format to use. If this property is specified, it **MUST** be part of a [Style](#) record.

NumeralVariant = numeralVariantStart numeralVariantValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|---------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| numeralVariantStart | | | | | | | | | | numeralVariantValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

numeralVariantStart (1 byte): A byte that specifies the start of the **NumeralVariant** property. The value of the **numeralVariantStart** field value **MUST** be one of the following values.

| Value | Parent |
|-------|---|
| 0x25 | SharedStyleProperties NonSharedStyleProperties |

numeralVariantValue (4 bytes): An integer variable value that specifies the variant of the digit format to use. [<48>](#) The default value is 0x00000001. The value of the **numeralVariantValue** field **MUST** be one of the following values.

| Value | Meaning |
|------------|--|
| 0x00000001 | default, follow Unicode context rules |
| 0x00000002 | 0123456789 |
| 0x00000003 | traditional digits for the script as defined in GDI+. Currently supported for: ar bn bo fa gu hi kn kok lo mr ms or pa sa ta te th ur and variants |
| 0x00000004 | ko, ja, zh-CHS, zh-CHT only |
| 0x00000005 | ko, ja, zh-CHS, zh-CHT only |
| 0x00000006 | ko, ja, zh-CHS, zh-CHT only [Wide versions of regular digits] |
| 0x00000007 | ko only |

2.3.93 Calendar

The **Calendar** property specifies the **calendar** to use for formatting dates. If the **Calendar** property is specified, its value **MUST** be compatible with the .NET Framework Language setting. The property **MUST** be part of a [Style](#) record. If the **Calendar** property is not specified, the default value is used.

2.3.95 RowHeaderColumns

The **RowHeaderColumns** property specifies the number of visible columns in the [Tablix](#) row headers area. If the **RowHeaderColumns** property is specified, it MUST be part of the [TablixMeasurements](#) record. If the **RowHeaderColumns** property is not specified, the default value is used.

RowHeaderColumns = rowHeaderColumnsStart rowHeaderColumnsValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------------|---|---|---|---|---|---|---|---|---|-----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| rowHeaderColumnsStart | | | | | | | | | | rowHeaderColumnsValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

rowHeaderColumnsStart (1 byte): A byte that specifies the start of the **RowHeaderColumns** property. The value of the **rowHeaderColumnsStart** field MUST be the following value.

| Value | Parent |
|-------|---------------------------|
| 0x01 | TablixMeasurements |

rowHeaderColumnsValue (4 bytes): A signed **integer** that MUST be greater than or equal to zero. [51](#) The default value of this field is 0x00000000.

2.3.96 ColsBeforeRowHeader

The **ColsBeforeRowHeader** property specifies the number of [Tablix](#) body area columns, which are displayed in order specified by the [LayoutDirection](#) property. The **LayoutDirection** property MUST be rendered before the row headers area. If the **ColsBeforeRowHeader** property is specified, it MUST be part of the [TablixMeasurements](#) record. If this property is not specified, the default value is used.

ColsBeforeRowHeader = colsBeforeRowHeaderStart colsBeforeRowHeaderValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------------|---|---|---|---|---|---|---|---|---|--------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| colsBeforeRowHeaderStart | | | | | | | | | | colsBeforeRowHeaderValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

colsBeforeRowHeaderStart (1 byte): A byte that specifies the start of the **ColsBeforeRowHeader** property. The value of the **colsBeforeRowHeaderStart** field MUST be the following value.

| Value | Parent |
|-------|--------------------|
| 0x02 | TablixMeasurements |

colsBeforeRowHeaderValue (4 bytes): A signed **integer** that MUST be greater than or equal to zero. <52> The default value of this field is 0x00000000.

2.3.97 LayoutDirection

The **LayoutDirection** property specifies the direction of the layout of **Tablix** columns. If this property is specified, it MUST be part of the **TablixMeasurements** record. If this property is not specified, the default value is used.

LayoutDirection = layoutDirectionStart layoutDirectionValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | 1 | | | | | | | | 2 | | | | | | | | 3 | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| layoutDirectionStart | | | | | | | | | | | | | | | | layoutDirectionValue | | | | | | | | | | | | | | | |

layoutDirectionStart (1 byte): A byte that specifies the start of the **LayoutDirection** property. The value of the **layoutDirectionStart** field MUST be the following value.

| Value | Parent |
|-------|--------------------|
| 0x03 | TablixMeasurements |

layoutDirectionValue (1 byte): A byte that specifies the layout direction of tablix columns. The value of the **layoutDirectionValue** field MUST be defined by the **Directions** enumeration. The default value is 0x00.

2.3.98 DefinitionPath

The **DefinitionPath** property specifies a unique identifier for a **TablixMemberDef** record. If this property is specified, it MUST be part of the **TablixMemberDef** record.

DefinitionPath = definitionPathStart definitionPathValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|--------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|
| | | | | | | | | | 1 | | | | | | | | | 2 | | | | | | | | | 3 | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | |
| definitionPathStart | | | | | | | | | definitionPathValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | |

...

definitionPathStart (1 byte): A byte that specifies the start of a **DefinitionPath** property. The value of the **definitionPathStart** field MUST be the following value.

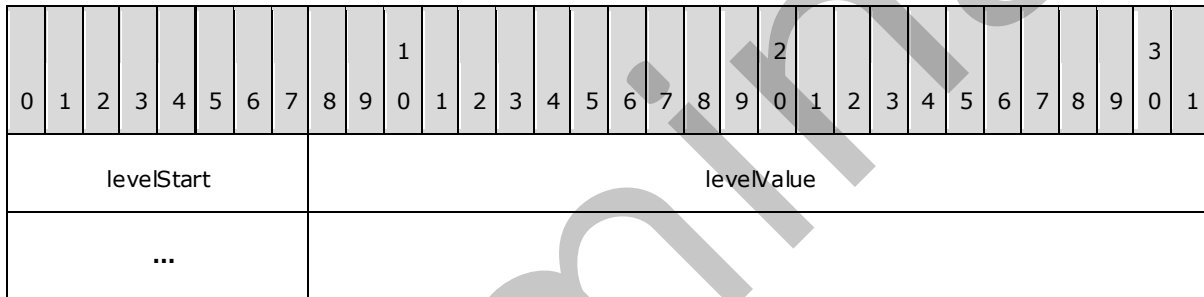
| Value | Parent |
|-------|------------------------|
| 0x00 | TablixMemberDef |

definitionPathValue (variable): A **String** variable that specifies a unique identifier for a **tablix member**.

2.3.99 Level

The **Level** property specifies the level of a [TablixMemberDef](#) record in the row or column hierarchy of a [Tablix](#) record. If the **Level** property is specified, it MUST be part of a **TablixMemberDef** record. If the **Level** property is not specified, the default value is used.

Level = levelStart levelValue



levelStart (1 byte): A byte that specifies the start of the **Level** property. The value of the **levelStart** field MUST be the following value.

| Value | Parent |
|-------|------------------------|
| 0x01 | TablixMemberDef |

levelValue (4 bytes): A signed **integer** that specifies the level of a tablix member definition. The default value is 0x00000000.

2.3.100 MemberCellIndex

The **MemberCellIndex** property specifies the index of a tablix body cell in the report definition. In the report definition, tablix body cells are organized in a double-indexed array. The first index specifies the row index. The second index specifies the column index.

If the **MemberCellIndex** property is specified, it MUST be part of a [TablixMemberDef](#) record. If the ancestor of the **MemberCellIndex** is [TablixRowHeader](#), the **MemberCellIndex** property MUST be a row index in the tablix body cells in report definition. If the ancestor of the **MemberCellIndex** property is [TablixColumnHeader](#), the **MemberCellIndex** property MUST be a column index in the tablix body cells in report definition. If this property is not specified, the default value is used.

MemberCellIndex = memberCellIndexStart memberCellIndexValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| memberCellIndexStart | | | | | | | | | | memberCellIndexValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

memberCellIndexStart (1 byte): A byte that specifies the start of a **MemberCellIndex** property. The value of **memberCellIndexStart** field MUST be the following value.

| Value | Parent |
|-------|------------------------|
| 0x02 | TablixMemberDef |

memberCellIndexValue (4 bytes): A signed **integer** value that specifies the index of a tablix body cell in the report definition. The default value is 0x00000000.

2.3.101 CellItemOffset

The **CellItemOffset** property specifies the position of a [ReportElementEnd](#) record in an RPL stream. The value of the **CellItemOffset** property MUST correspond to the top-level report item of a tablix cell.

If the **CellItemOffset** property is specified, it MUST be part of a [TablixCorner](#), [TablixRowHeader](#), [TablixColumnHeader](#), or [TablixBodyCell](#) record.

CellItemOffset = cellItemOffsetStart cellItemOffset

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------|---|---|---|---|---|---|---|---|---|---------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| cellItemOffsetStart | | | | | | | | | | cellItemOffsetValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

cellItemOffsetStart (1 byte): A byte that specifies the start of the **CellItemOffset** property. The value of the **cellItemOffsetStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x04 | TablixCorner TablixColumnHeader TablixRowHeader TablixBodyCell |

cellItemOffsetValue (8 bytes): A signed integer that specifies the position of a **ReportElementEnd** record in the RPL stream, and that MUST correspond to the top-level [ReportItem](#) record of the parent tablix cell.

2.3.102 ColSpan

The **ColSpan** property specifies the number of columns a cell spans in a [Tablix](#) record. If this property is specified, it MUST be part of a [TablixCorner](#), [TablixRowHeader](#), [TablixColumnHeader](#), or [TablixBodyCell](#) record. If the **ColSpan** property is not specified, the default value is used.

ColSpan = colSpanStart colSpanValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------|---|---|---|---|---|---|---|---|---|--------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| colSpanStart | | | | | | | | | | colSpanValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

colSpanStart (1 byte): A byte that specifies the start of the **ColSpan** property. The value of the **colSpanStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x05 | TablixCorner TablixColumnHeader TablixRowHeader TablixBodyCell |

colSpanValue (4 bytes): A signed integer that specifies the number of columns the parent cell spans. The default value is 0x00000001.

2.3.103 RowSpan

The **RowSpan** property specifies the number of rows a cell spans in a [Tablix](#) record. If the **RowSpan** property is specified, it MUST be part of a [TablixCorner](#), [TablixRowHeader](#), [TablixColumnHeader](#), or [TablixBodyCell](#) record. If the **RowSpan** property is not specified, the default value is used.

RowSpan = rowSpanStart rowSpanValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------|---|---|---|---|---|---|---|---|---|--------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| rowSpanStart | | | | | | | | | | rowSpanValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

rowSpanStart (1 byte): A byte that specifies the start of the **RowSpan** property. The value of the **rowSpanStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x06 | TablixCorner TablixColumnHeader TablixRowHeader TablixBodyCell |

rowSpanValue (4 bytes): A signed integer that specifies the number of rows the parent cell spans. The default value is 0x00000001.

2.3.104 DefIndex

The **DefIndex** property specifies the zero-based index of a [TablixMemberDef](#) record. The index is stored in the **tablixMembersDefContents** array field of a [TablixColMembersDef](#) or [TablixRowMembersDef](#) record that corresponds to a column or row header in a [Tablix](#) record. If the **DefIndex** property is specified, it MUST be part of a [TablixColumnHeader](#) or [TablixRowHeader](#) record.

DefIndex = defIndexStart defIndexValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------|---|---|---|---|---|---|---|---|---|---------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| defIndexStart | | | | | | | | | | defIndexValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

defIndexStart (1 byte): A byte that specifies the start of the **DefIndex** property. The value of **defIndexStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x07 | TablixColumnHeader TablixRowHeader |

defIndexValue (4 bytes): A signed integer that MUST specify the index of a **TablixMemberDef** record. The index is stored into the **tablixMembersDefContents** array field of a **TablixColMembersDef** or **TablixRowMembersDef** record. If the parent is a **TablixColumnHeader**, the value of the **defIndexValue** field MUST be greater than or equal to zero and MUST be less than the **count** field of the **TablixColMembersDef** record of the parent **Tablix** record. If the parent is a **TablixRowHeader**, the value of the **defIndexValue** field MUST be greater than or equal to zero and MUST be less than the **count** field of the **TablixRowMembersDef** record of the parent **Tablix** record.

2.3.105 ColumnIndex

The **ColumnIndex** property specifies the zero-based index of the first column for a tablix cell. If the **ColumnIndex** property is specified, it MUST be part of [TablixCorner](#), [TablixRowHeader](#), [TablixColumnHeader](#), or [TablixBodyCell](#) record. If the **ColumnIndex** property is not specified, the default value is used.

ColumnIndex = columnIndexStart columnIndexValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| columnIndexStart | | | | | | | | | | columnIndexValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

columnIndexStart (1 byte): A byte that specifies the start of the **ColumnIndex** property. The value of the **columnIndexStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x08 | TablixCorner TablixColumnHeader TablixRowHeader TablixBodyCell |

columnIndexValue (4 bytes): A signed integer that specifies the zero-based index of the first column for a tablix cell. The value of this field MUST greater or equal to zero. The addition of the **ColSpan** property to the **columnIndexValue** field MUST result in a value that is less than the **count** field of **ColumnsWidths** record of the [Tablix](#) record. <53> The default value is 0xFFFFFFFF.

groupLabelStart (1 byte): A byte that specifies the start of the **GroupLabel** structure. The value of the **groupLabelStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x0A | TablixColumnHeader TablixRowHeader |

groupLabelValue (variable): A **String** that specifies the label for a row header or column header of the **Tablix** record.

2.3.108 RecursiveToggleLevel

The **RecursiveToggleLevel** property specifies the **recursive toggling** state of a row header or column header of a **Tablix** record. If the **RecursiveToggleLevel** property is specified, it MUST be part of a **TablixRowHeader** or **TablixColumnHeader**. If the **RecursiveToggleLevel** property is not specified, the default value is used.

```
RecursiveToggleLevel = recursiveToggleLevelStart recursiveToggleLevelValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---------------------------|---|---|---|---|---|---|---|---|---|---------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | |
| recursiveToggleLevelStart | | | | | | | | | | recursiveToggleLevelValue | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

recursiveToggleLevelStart (1 byte): A byte that specifies the start of the **RecursiveToggleLevel** property. The value of the **recursiveToggleLevelStart** field MUST be one of the following values.

| Value | Parent |
|-------|---|
| 0x0E | TablixColumnHeader TablixRowHeader |

recursiveToggleLevelValue (4 bytes): A signed integer that specifies the recursive toggling state of a row header or column header of the **Tablix** record. The default value is 0xFFFFFFFF. The value of the **recursiveToggleLevelValue** field MUST be one of the following.

| Value | Parent |
|-------|--|
| <0 | The group of parent header is not recursive, the parent header is not toggleable, or the toggle is not recursive. |
| 0 | This parent header is a root of a toggleable recursive forest. |
| >0 | This parent header is a child in the toggleable recursive hierarchy. The value specifies the depth of the item in the recursive hierarchy. |

2.3.109 ListStyle

The **ListStyle** property specifies if a [Paragraph](#) record is part of a list and identifies the numbering type. If the **ListStyle** property is specified, it MUST be part of a [Paragraph.ElementProperties](#) record. If the **ListStyle** property is not specified, the default value is used.

ListStyle = listStyleStart listStyleValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| listStyleStart | | | | | | | | | | listStyleValue | | | | | | | | | | | | | | | | | | | | | |

listStyleStart (1 byte): A byte that specifies the start of the **ListStyle** property. The value of the **listStyleStart** field MUST be one of the following.

| Value | Parent |
|-------|---|
| 0x07 | ElementProperties.InlineSharedElementProperties for Paragraph ElementProperties.NonSharedElementProperties for Paragraph |

listStyleValue (1 byte): A byte that specifies whether the paragraph is part of a list, and that identifies the numbering type. The value of the **listStyleValue** field is of type [ListStyles](#) and it MUST be contained in the **ListStyles** enumeration. The default value is 0x00.

2.3.110 ListLevel

The **ListLevel** property specifies the numbering style and/or indentation level of a [Paragraph](#) record. If the **ListLevel** property is specified, it MUST be part of a [Paragraph.ElementProperties](#) record. If the **ListLevel** property is not specified, the default value is used.

ListLevel = listLevelStart listLevelValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| listLevelStart | | | | | | | | | | listLevelValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

listLevelStart (1 byte): A byte that specifies the start of the **ListLevel** property. The value of the **listLevelStart** field MUST be one of the following.

| Value | Parent |
|-------|---|
| 0x08 | ElementProperties.InlineSharedElementProperties for Paragraph ElementProperties.NonSharedElementProperties for Paragraph |

listLevelValue (4 bytes): A signed integer that specifies the numbering style and/or indentation level of the **Paragraph** record. The default value is 0x00000000.

2.3.111 ParagraphNumber

The **ParagraphNumber** property specifies the one-based index of a [Paragraph](#) record, which is a member of list of paragraphs that have the same value for a [ListLevel](#) property and MUST have the [ListStyle](#) property set to 0x01. If the **ParagraphNumber** property is specified, it MUST be part of a [ElementProperties.NonSharedElementProperties](#) record for a **Paragraph** record. If the **ParagraphNumber** property is not specified, the default value is used.

ParagraphNumber = paragraphNumberStart paragraphNumberValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------------|---|---|---|---|---|---|---|---|---|----------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|
| | | | | | | | | | | 1 | | | | | | | | | | | 2 | | | | | | | | | | | 3 | | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | | |
| paragraphNumberStart | | | | | | | | | | paragraphNumberValue | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

paragraphNumberStart (1 byte): A byte that specifies the start of the **ParagraphNumber** property. The value of the **paragraphNumberStart** field MUST be one of the following.

| Value | Parent |
|-------|---|
| 0x0E | ElementProperties.NonSharedElementProperties for Paragraph |

paragraphNumberValue (4 bytes): A signed integer that specifies the number of the paragraph, which is a member of a list of paragraphs that have the same value for the **ListLevel** property and MUST have the **ListStyle** property set to 0x01. [<55>](#) The default value is 0x00000000.

2.3.112 RightIndent

The **RightIndent** property specifies the indentation for all text within a paragraph from the right edge of the text box containing the [Paragraph](#). If this property is specified, it MUST be part of [Paragraph.ElementProperties](#) record.

RightIndent = rightIndentStart rightIndentValue

2.3.114 HangingIndent

The **HangingIndent** property specifies the indent of the first line of text within a paragraph relative to the **LeftIndent** property of the same paragraph. If this property is specified, it MUST be part of a [Paragraph.ElementProperties](#) record.

HangingIndent = hangingIndentStart hangingIndentValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------|---|---|---|---|---|---|---|---|---|-------------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|--|--|--|
| | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | | | | | |
| hangingIndentStart | | | | | | | | | | hangingIndentValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

hangingIndentStart (1 byte): A byte that specifies the start of the **HangingIndent** property. The value of the **hangingIndentStart** field MUST be one of the following.

| Value | Parent |
|-------|---|
| 0x0B | ElementProperties.InlineSharedElementProperties for Paragraph ElementProperties.NonSharedElementProperties for Paragraph |

hangingIndentValue (variable): An [RPLSize](#) variable that specifies the indent of the first line of text within a paragraph relative to the **LeftIndent** property of the same paragraph.

2.3.115 SpaceBefore

The **SpaceBefore** property specifies the vertical spacing before a paragraph. If the **SpaceBefore** property is specified, it MUST be part of a [Paragraph.ElementProperties](#) record.

SpaceBefore = spaceBeforeStart spaceBeforeValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|-----------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|--|--|--|--|--|--|--|--|
| | | | | | | | | | | 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | | | | | | | | | |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | | | | | | | | |
| spaceBeforeStart | | | | | | | | | | spaceBeforeValue (variable) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

spaceBeforeStart (1 byte): A byte that specifies the start of the **SpaceBefore** property. The value of the **spaceBeforeStart** field MUST be one of the following.

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------------|---|---|---|---|---|---|---|---|---|----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| firstLineStart | | | | | | | | | | firstLineValue | | | | | | | | | | | | | | | | | | | | | |

firstLineStart (1 byte): A byte that specifies the start of the **FirstLine** structure. The value of the **firstLineStart** field MUST be one of the following.

| Value | Parent |
|-------|--|
| 0x0F | ElementProperties.NonSharedElementProperties for Paragraph |

firstLineValue (1 byte): A **Boolean** value that specifies whether the first line of text of a paragraph is part of the current report page. The default value is 0x01.

2.3.118 Markup

The **Markup** property specifies the markup style for a [TextRun](#) record. If this property is specified, it MUST be part of an [ElementProperties](#) record of a **TextRun**. If the **Markup** property is not specified, the default value is used.

Markup = markupStart markupValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------|---|---|---|---|---|---|---|---|---|-------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| markupStart | | | | | | | | | | markupValue | | | | | | | | | | | | | | | | | | | | | |

markupStart (1 byte): A byte that specifies the start of the **Markup** property. The value of the **markupStart** field MUST be one of the following.

| Value | Parent |
|-------|---|
| 0x07 | ElementProperties.InlineSharedElementProperties for TextRun ElementProperties.NonSharedElementProperties for TextRun |

markupValue (1 byte): A byte that specifies the markup style for a **TextRun** record. The value of the **markupValue** field is of type [MarkupStyles](#) and it MUST be contained in the **MarkupStyles** enumeration. The default value is 0x00.

2.3.119 ContentTop

The **ContentTop** property specifies the top position of the content relative to the container top. If this property is specified, it MUST be part of a [TextRun](#), [Paragraph](#), [TablixMeasurements](#), [TablixCorner](#), [TablixColumnHeader](#), [TablixRowHeader](#), or [TablixBodyCell](#) record. If the **ContentTop** property is not specified, the default value is used.

ContentTop = contentTopStart contentTopValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------------|---|---|---|---|---|---|---|---|---|-----------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| contentTopStart | | | | | | | | | | contentTopValue | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

contentTopStart (1 byte): A byte that specifies the start of the **ContentTop** property. The value of the **contentTopStart** field MUST be one of the following.

| Value | Parent |
|-------|--|
| 0x00 | ElementProperties.NonSharedElementProperties for TextRun ElementProperties.NonSharedElementProperties for Paragraph TablixCorner TablixColumnHeader TablixRowHeader TablixBodyCell |
| 0x06 | TablixMeasurements |

contentTopValue (4 bytes): A floating point value that specifies the top position of the content relative to the container start. The default value is 0x00000000.

2.3.120 ContentLeft

The **ContentLeft** property specifies the size of the left position relative to the container top. The **ContentLeft** property MUST be part of a [TextRun](#), [Paragraph](#), [TablixMeasurements](#), [TablixCorner](#), [TablixColumnHeader](#), [TablixRowHeader](#), or [TablixBodyCell](#) record. If the **ContentLeft** property is not specified, the default value is used.

ContentLeft = contentLeftStart contentLeftValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------------|---|---|---|---|---|---|---|---|---|------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| contentLeftStart | | | | | | | | | | contentLeftValue | | | | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

contentLeftStart (1 byte): A byte that specifies the start of the **ContentLeft** property. The value of the **contentLeftStart** field MUST be one of the following.

| Value | Parent |
|-------|---|
| 0x01 | ElementProperties.NonSharedElementProperties for TextRun ElementProperties.NonSharedElementProperties for Paragraph TablixCorner TablixColumnHeader TablixRowHeader TablixBody Cell |
| 0x07 | TablixMeasurements |

contentLeftValue (4 bytes): A floating point value that specifies the left position relative to the container top. The default value is 0x00000000.

2.3.121 ContentWidth

The **ContentWidth** property specifies the width of the content of a container. If the **ContentWidth** property is specified, it MUST be part of a [TextRun](#), [Paragraph](#), [TablixCorner](#), [TablixColumnHeader](#), [TablixRowHeader](#), or [TablixBodyCell](#) record. If the **ContentWidth** property is not specified, the default value is used.

ContentWidth = contentWidthStart contentWidthValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-------------------|---|---|---|---|---|---|---|---|---|-------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| contentWidthStart | | | | | | | | | | contentWidthValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

contentWidthStart (1 byte): A byte that specifies the start of the **ContentWidth** property. The value of the **contentWidthStart** field MUST be one of the following.

| Value | Parent |
|-------|---|
| 0x02 | ElementProperties.NonSharedElementProperties for TextRun ElementProperties.NonSharedElementProperties for Paragraph TablixCorner TablixColumnHeader TablixRowHeader TablixBody Cell |

contentWidthValue (4 bytes): A floating point value that specifies the width of the content of a container. The default value is defined by the container width.

2.3.122 ContentHeight

The **ContentHeight** property specifies the height of the content of a container. If the **ContentHeight** property is specified, it MUST be part of a [TextRun](#), [Paragraph](#), [TablixCorner](#), [TablixColumnHeader](#), [TablixRowHeader](#), [TablixBodyCell](#), or [ElementProperties.NonSharedElementProperty](#) record. If the **ContentHeight** property is not specified, the default value is used.

```
ContentHeight = contentHeightStart contentHeightValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|--------------------|---|---|---|---|---|---|---|---|---|--------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| contentHeightStart | | | | | | | | | | contentHeightValue | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

contentHeightStart (1 byte): A byte that specifies the start of the **ContentHeight** property. The value of the **contentHeightStart** field MUST be one of the following.

| Value | Parent |
|-------|--|
| 0x03 | TextRun Paragraph TablixCorner TablixColumnHeader TablixRowHeader TablixBody Cell |
| 0x24 | ElementProperties.NonSharedElementProperty for RichTextBox |

contentHeightValue (4 bytes): A floating point value that specifies the height of the content of a container. The default value is defined by the container height.

2.3.123 State

The **State** property specifies the state of a row header or column header of a [Tablix](#) record. If the **State** property is specified, it MUST be part of a [TablixRowHeader](#) or [TablixColumnHeader](#) record. If the **State** property is not specified, the default value is used.

```
State = stateStart stateValue
```

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------|---|---|---|---|---|---|---|---|---|------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| stateStart | | | | | | | | | | stateValue | | | | | | | | | | | | | | | | | | | | | | | | |

stateStart (1 byte): A byte that specifies the start of the **State** property. The value of the **stateStart** field MUST be one of the following.

| Value | Parent |
|-------|---|
| 0x0C | TablixRowHeader TablixColumnHeader |

stateValue (1 byte): A byte that specifies the state of a row header or column header of a **Tablix** record. The meaning of this field value is derived from the [TablixMemberStateValues](#) structure. The default value is 0x00.

2.3.124 CellItemState

The **CellItemState** property specifies the state of the top-level report item of a tablix cell. If the **CellItemState** property is specified, it MUST be part of a [TablixBodyCell](#), [TablixCorner](#), [TablixRowHeader](#), or [TablixColumnHeader](#) record. If this property is not specified, the default value is used.

CellItemState = stateStart stateValue

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|------------|---|---|---|---|---|---|---|---|---|------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 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 |
| stateStart | | | | | | | | | | stateValue | | | | | | | | | | | | | | | | | | | | | | | | |

stateStart (1 byte): A byte that specifies the start of the **CellItemState** property. The value of the **stateStart** field MUST be one of the following.

| Value | Parent |
|-------|---|
| 0x0D | TablixCorner TablixColumnHeader TablixRowHeader TablixBodyCell |

stateValue (1 byte): A byte that specifies the state of the top-level report item of a tablix cell. The meaning of this field value is derived from the [StateValues](#) enumeration. The default value is 0x00.

| Value | Meaning |
|-------|-----------------|
| 0x00 | AutoSize |
| 0x01 | Fit |
| 0x02 | FitProportional |
| 0x03 | Clip |

2.4.3 ShapeType

The **ShapeType** enumeration specifies the type of shape that is used for an image map. This enumeration contains valid values for the [ImageMapArea](#) property. The value of the **ShapeType** enumeration MUST be one of the following.

| Value | Meaning |
|-------|-----------|
| 0x00 | Rectangle |
| 0x01 | Polygon |
| 0x02 | Circle |

2.4.4 ImageRawFormat

The **ImageRawFormat** enumeration specifies the **raw format** for an image. This enumeration contains valid values for the [RawFormat](#) property. The value of the **ImageRawFormat** enumeration MUST be one of the following.

| Value | Meaning |
|-------|---------|
| 0x00 | BMP |
| 0x01 | JPEG |
| 0x02 | GIF |
| 0x03 | PNG |

2.4.5 FontStyles

The **FontStyles** enumeration specifies the style of font used for a contiguous set of characters. This enumeration contains valid values for the [FontStyle](#) property. The value of the **FontStyles** enumeration MUST be one of the following.

| Value | Meaning |
|-------|---------|
| 0x00 | Normal |
| 0x01 | Italic |

2.4.6 FontWeights

The **FontWeights** enumeration specifies the weight of the font used for a contiguous set of characters. This enumeration contains valid values for the [FontWeight](#) property. The value of the **FontWeights** enumeration MUST be one of the following.

| Value | Meaning |
|-------|------------|
| 0x00 | Normal |
| 0x01 | Thin |
| 0x02 | ExtraLight |
| 0x03 | Light |
| 0x04 | Medium |
| 0x05 | SemiBold |
| 0x06 | Bold |
| 0x07 | ExtraBold |
| 0x08 | Heavy |

2.4.7 TextDecorations

The **TextDecorations** enumeration specifies the decoration used for a contiguous set of characters. This enumeration contains valid values for the [TextDecoration](#) property. The value of the **TextDecorations** enumeration MUST be one of the following.

| Value | Meaning |
|-------|-------------|
| 0x00 | None |
| 0x01 | Underline |
| 0x02 | Overline |
| 0x03 | LineThrough |

2.4.8 TextAlignments

The **TextAlignments** enumeration specifies the horizontal alignment of a contiguous set of characters. This enumeration contains valid values for the [TextAlign](#) property. The **General** value specifies that the text alignment is derived from the data type of the value for the span of text and the peer style [Direction](#) property. The value of the **TextAlignments** enumeration MUST be one of the following.

| Value | Meaning |
|-------|---------|
| 0x00 | General |
| 0x01 | Left |

| Value | Meaning |
|-------|---------|
| 0x02 | Center |
| 0x03 | Right |

2.4.9 VerticalAlignments

The **VerticalAlignments** enumeration specifies the vertical alignment of a contiguous set of characters. This enumeration contains valid values for the [VerticalAlign](#) property. The value of the **VerticalAlignments** enumeration MUST be one of the following.

| Value | Meaning |
|-------|---------|
| 0x00 | Top |
| 0x01 | Middle |
| 0x02 | Bottom |

2.4.10 Directions

The **Directions** enumeration specifies the direction in which a contiguous set of characters is written and the layout direction of tablix columns. This enumeration contains valid values for the style [Direction](#) and tablix [LayoutDirection](#) properties. The value of the **Directions** enumeration MUST be one of the following.

| Value | Meaning |
|-------|---------|
| 0x00 | LTR |
| 0x01 | RTL |

2.4.11 WritingModes

The **WritingModes** enumeration specifies the direction in which to write a contiguous set of characters. This enumeration contains valid values for the [WritingMode](#) property. The value of the **WritingModes** enumeration MUST be one of the following.

| Value | Meaning |
|-------|------------|
| 0x00 | Horizontal |
| 0x01 | Vertical |

2.4.12 UnicodeBiDiTypes

The **UnicodeBiDiTypes** enumeration specifies the level of embedding with respect to the bi-directional algorithm. This enumeration contains valid values for the [UnicodeBiDi](#) property. The value of the **UnicodeBiDiTypes** enumeration MUST be one of the following.

| Value | Meaning |
|-------|--------------|
| 0x00 | Normal |
| 0x01 | Embed |
| 0x02 | BiDiOverride |

2.4.13 Calendars

The **Calendars** enumeration specifies the type of calendar to use for formatting dates. This enumeration contains valid values for the [Calendar](#) property. The value of the **Calendars** enumeration MUST be one of the following.

| Value | Meaning |
|-------|--------------------------------|
| 0x00 | Gregorian |
| 0x01 | GregorianArabic |
| 0x02 | GregorianMiddleEastFrench |
| 0x03 | GregorianTransliteratedEnglish |
| 0x04 | GregorianTransliteratedFrench |
| 0x05 | GregorianUSEnglish |
| 0x06 | Hebrew |
| 0x07 | Hijri |
| 0x08 | Japanese |
| 0x09 | Korean |
| 0x10 | Julian |
| 0x11 | Taiwan |
| 0x12 | ThaiBuddist |

2.4.14 BorderStyles

The **BorderStyles** enumeration specifies the style for a given border. This enumeration contains valid values for the following border style properties: [BorderStyle](#), [BorderStyleLeft](#), [BorderStyleRight](#), [BorderStyleTop](#), and [BorderStyleBottom](#). The value of the **BorderStyles** enumeration MUST be one of the following.

| Value | Meaning |
|-------|---------|
| 0x00 | None |
| 0x01 | Dotted |
| 0x02 | Dashed |

| Value | Meaning |
|-------|---------|
| 0x03 | Solid |
| 0x04 | Double |

2.4.15 BackgroundRepeatTypes

The **BackgroundRepeatTypes** enumeration specifies how a background image should fill the space available within its parent container. This enumeration contains valid values for the [BackgroundRepeat](#) property. The value of the **BackgroundRepeatTypes** enumeration MUST be one of following.

| Value | Meaning |
|-------|---------|
| 0x00 | Repeat |
| 0x01 | Clip |
| 0x02 | RepeatX |
| 0x03 | RepeatY |

2.4.16 ListStyles

The **ListStyles** enumeration specifies the style used for a list. This enumeration contains valid values for the [ListStyle](#) property. The value of the **ListStyles** enumeration MUST be one of the following.

| Value | Meaning |
|-------|----------|
| 0x00 | None |
| 0x01 | Numbered |
| 0x02 | Bulleted |

2.4.17 MarkupStyles

The **MarkupStyles** enumeration specifies the type of markup to use for a block of text. This enumeration contains valid values for the [Markup](#) property. The value of the **MarkupStyles** enumeration MUST be one of the following.

| Value | Meaning |
|-------|---------|
| 0x00 | None |
| 0x01 | HTML |
| 0x02 | RTF |

C - StaticHeadersTree (1 bit): A bit that specifies whether the row or column header and all its children are static.

| Value | Meaning |
|-------|---|
| 0x0 | False. The row or column header tree is not static. |
| 0x1 | True. The row or column header tree is static. |

reserved (5 bits): The value MUST be zero.

2.4.22 RPLSize

The **RPLSize** structure specifies size information for the properties [BorderWidth](#), [BorderWidthLeft](#), [BorderWidthRight](#), [BorderWidthTop](#), [BorderWidthBottom](#), [PaddingLeft](#), [PaddingRight](#), [PaddingTop](#), [PaddingBottom](#), [FontSize](#), [LineHeight](#), [NumeralLanguage](#), [RightIndent](#), [LeftIndent](#), [HangingIndent](#), [SpaceBefore](#), and [SpaceAfter](#).

RPLSize = FormatValue FormatType

| 1 | | | | | | | | | | 2 | | | | | | | | | | 3 | |
|------------------------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 |
| formatValue (variable) | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | |
| formatType (variable) | | | | | | | | | | | | | | | | | | | | | |
| ... | | | | | | | | | | | | | | | | | | | | | |

formatValue (variable): The value of the **formatValue** field MUST be a numeric value serialized as a **String** in the form of an integer with no thousand separators, optionally followed by a decimal point and a positive integer. Unless otherwise specified, the value of the **formatValue** field MUST NOT be a negative integer or float serialized as a **String**.

formatType (variable): The value of the **formatType** field MUST be a **String** that specifies the unit type that is used for the distance. The value of the **formatType** field MUST be one of the following.

| Value | Meaning |
|-------|---|
| pt | Specifies a size in units of points. |
| pc | Specifies a size in units of picas. |
| in | Specifies a size in units of inches. |
| mm | Specifies a size in units of millimeters. |

| Value | Meaning |
|-------|---|
| cm | Specifies a size in units of centimeters. |

Preliminary

3 Structure Examples

This section contains examples of some of the most commonly used data structures in the RPL stream. The examples are meant to be a starting point for an implementer who is learning the stream format. They do not cover all the records in the stream format.

The following conventions are followed for the examples, unless noted otherwise:

- The order of the records, structures, and fields within an example match their corresponding order in the stream format.
- The examples begin with the first record that is relevant to the example and end with the last record that is relevant to the example. With the exception of the Report example, none of the examples can be used as a complete and stand-alone MS-RPL 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.
- Offset values for the records are included in the leftmost column because they are heavily relied upon in the RPL stream. While the value of the **offset** is an 8-byte integer, only one byte is shown in the Offset column.

3.1 Line

The following example illustrates a [Line](#) record with a [Slant](#) property in an RPL stream.

| Offset | Size | Structure | Value |
|--------|------|--------------------------------------|--------------------|
| | 0011 | Line | |
| 0x01 | 0001 | BYTE - lineToken | 0x08 |
| | 0006 | LineProperties | |
| 0x02 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0004 | SharedElementProperties | |
| 0x03 | 0001 | BYTE - shared | 0x00 |
| | 0002 | ItemProperties | |
| 0x04 | 0001 | BYTE - slantStart | 0x18 |
| 0x05 | 0001 | BYTE - slantValue | 0x00 |
| 0x06 | 0001 | BYTE - delimiter | 0xFF |
| 0x07 | 0001 | BYTE - delimiter | 0xFF |
| | 000A | ReportElementEnd | |
| 0x08 | 0001 | BYTE - elementEnd | 0xFE |
| 0x09 | 0008 | INT - offset | 0x0000000000000001 |

| Offset | Size | Structure | Value |
|--------|------|-------------------------|-------|
| 0x11 | 0001 | BYTE - delimiter | 0xFF |

Line: A Line record that specifies the **Line** report item.

lineToken: 0x08 specifies the start **token** of the **Line** record.

LineProperties: An [ElementProperties](#) record that specifies the properties for the **Line** report item.

elementPropertiesStart: 0x0F specifies the start of the **ElementProperties** record.

SharedElementProperties: A [SharedElementProperties](#) record that specifies the [InlineSharedElementProperties](#) record of the **Line** record.

shared: 0x00 specifies the start token of the InlineSharedElementProperties record.

ItemProperties: An array of properties for the **Line** record.

slantStart: 0x18 specifies the start of the Slant property.

slantValue: 0x00 specifies that the line will be drawn from the lower left to the upper right.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

delimiter: 0xFF specifies the end of the **LineProperties** record.

ReportElementEnd: A [ReportElementEnd](#) record specifies the end of the **Line** report item.

elementEnd: 0xFE specifies the end of the **Line** record.

offset: 0x0000000000000001 specifies the position of the **lineToken** field in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

3.2 Image

The following example illustrates an [Image](#) record with a [Sizing](#) property in an RPL stream.

| Offset | Size | Structure | Value |
|--------|------|--------------------------------------|-------|
| | 0011 | Image | |
| 0x01 | 0001 | BYTE - imageStart | 0x09 |
| | 0006 | ImageProperties | |
| 0x02 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0004 | SharedElementProperties | |
| 0x03 | 0001 | BYTE - shared | 0x00 |
| | 0002 | ItemProperties | |
| 0x04 | 0001 | BYTE - sizingStart | 0x29 |
| 0x05 | 0001 | BYTE - sizingValue | 0x00 |

| Offset | Size | Structure | Value |
|--------|------|--------------------------|--------------------|
| 0x06 | 0001 | BYTE - delimiter | 0xFF |
| 0x07 | 0001 | BYTE - delimiter | 0xFF |
| | 000A | ReportElementEnd | |
| 0x08 | 0001 | BYTE - elementEnd | 0xFE |
| 0x09 | 0008 | INT - offset | 0x0000000000000001 |
| 0x11 | 0001 | BYTE - delimiter | 0xFF |

Image: An Image record that specifies the **Image** report item.

imageStart: 0x09 specifies the start token of the **Image** record.

ImageProperties: An ImageProperties record specifies the properties of the **Image** report item.

elementPropertiesStart: 0x0F specifies the start of the **ImageProperties** record.

SharedElementProperties: A **SharedElementProperties** structure that specifies the [InlineSharedElementProperties](#) record of the **Image** record.

shared: 0x00 specifies the start token of the **SharedElementProperties** record.

ItemProperties: An array of properties for the **Image** record.

sizingStart: 0x29 specifies the start token of the Sizing property.

sizingValue: 0x00 specifies the default **Sizing** value. The **Image** report item will be set to **AutoSize**.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

delimiter: 0xFF specifies the end of the **ImageProperties** record.

ReportElementEnd: A [ReportElementEnd](#) record specifies the end of image.

elementEnd: 0xFE specifies the end of the **Image** record.

offset: 0x0000000000000001 specifies the position of the **imageToken** field in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

3.3 Chart

The following example illustrates a [Chart](#) record with [Label](#) and [StreamName](#) properties in an RPL stream.

| Offset | Size | Structure | Value |
|--------|------|--------------------------|-------|
| | 002F | Chart | |
| 0x01 | 0001 | BYTE - chartToken | 0x0B |
| | 0024 | ChartProperties | |

| Offset | Size | Structure | Value |
|--------|------|--------------------------------------|--------------------|
| 0x02 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0010 | SharedElementProperties | |
| 0x03 | 0001 | BYTE - shared | 0x00 |
| | 000E | ItemProperties | |
| 0x04 | 0001 | BYTE - labelStart | 0x03 |
| 0x05 | 000D | STRING - labelValue | Chart1 |
| 0x12 | 0001 | BYTE - delimiter | 0xFF |
| | 0012 | NonSharedElementProperties | |
| 0x13 | 0001 | BYTE - nonShared | 0x01 |
| | 0010 | ItemProperties | |
| 0x14 | 0001 | BYTE - streamNameStart | 0x28 |
| 0x15 | 000F | STRING - streamNameValue | Stream1 |
| 0x24 | 0001 | BYTE - delimiter | 0xFF |
| 0x25 | 0001 | BYTE - delimiter | 0xFF |
| | 000A | ReportElementEnd | |
| 0x26 | 0001 | BYTE - elementEnd | 0xFE |
| 0x27 | 0008 | INT - offset | 0x0000000000000001 |
| 0x2F | 0001 | BYTE - delimiter | 0xFF |

Chart: A [Chart](#) record that specifies the **Chart** report item.

chartToken: 0x0B specifies the start token of the **Chart** record.

ChartProperties: An [ElementProperties](#) record that specifies the properties of the **Chart** report item.

elementPropertiesStart: 0x0F specifies the start of the **ElementProperties** record.

SharedElementProperties: A [SharedElementProperties](#) record that specifies the [InlineSharedElementProperties](#) record of the **Chart** record.

shared: 0x00 specifies the start token of the **InlineSharedElementProperties** record.

ItemProperties: An array of properties for the **Chart** record.

labelStart: 0x03 specifies the start of Label property.

labelValue: *Chart1* specifies the label of the **Chart** report item.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

NonSharedElementProperties: A [NonSharedElementProperties](#) record that specifies the properties of the **Chart** record.

nonShared: 0x01 specifies the token of the NonSharedElementProperties record.

ItemProperties: An array of properties for the **Chart** record.

streamNameStart: 0x28 specifies the start token of the StreamName property.

streamNameValue: *Stream1* specifies the stream name of the **Chart** report item.

delimiter: 0xFF specifies the end of the **NonSharedElementProperties** record.

delimiter: 0xFF specifies the end of the **ChartProperties** record.

ReportElementEnd: A [ReportElementEnd](#) structure specifies the end of the **Chart** record.

elementEnd: 0xFE specifies the end of the **Chart** element.

offset: 0x0000000000000001 specifies the position of the **chartToken** field in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

3.4 GaugePanel

The following example illustrates a [GaugePanel](#) record with [Label](#) and [Tooltip](#) properties in an RPL stream.

| Offset | Size | Structure | Value |
|--------|------|--------------------------------------|--------|
| | 002D | GaugePanel | |
| 0x01 | 0001 | BYTE - gaugePanelToken | 0x0E |
| | 0024 | GaugePanelProperties | |
| 0x02 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0010 | SharedElementProperties | |
| 0x03 | 0001 | BYTE - shared | 0x00 |
| | 000E | ItemProperties | |
| 0x04 | 0001 | BYTE - labelStart | 0x03 |
| 0x05 | 000D | STRING - labelValue | Panel1 |
| 0x12 | 0001 | BYTE - delimiter | 0xFF |
| | 0010 | NonSharedElementProperties | |
| 0x13 | 0001 | BYTE - nonShared | 0x01 |
| | 000E | ItemProperties | |
| 0x14 | 0001 | BYTE - toolTipStart | 0x05 |
| 0x15 | 000D | STRING - toolTipValue | Gauge1 |

| Offset | Size | Structure | Value |
|--------|------|--------------------------|--------------------|
| 0x22 | 0001 | BYTE - delimiter | 0xFF |
| 0x23 | 0001 | BYTE - delimiter | 0xFF |
| | 000A | ReportElementEnd | |
| 0x24 | 0001 | BYTE - elementEnd | 0xFE |
| 0x25 | 0008 | INT - offset | 0x0000000000000001 |
| 0x2D | 0001 | BYTE - delimiter | 0xFF |

GaugePanel: A GaugePanel record that specifies the **GaugePanel** report item.

gaugePanelToken: 0x0E specifies the start token of the **GaugePanel** record.

GaugePanelProperties: An [ElementProperties](#) record that specifies the properties of the **GaugePanel** report item.

elementPropertiesStart: 0x0F specifies the start token of the **GaugePanelProperties** record.

SharedElementProperties: A [SharedElementProperties](#) record that specifies the [InlineSharedElementProperties](#) record of the **GaugePanel** record.

shared: 0x00 specifies the start token of the **InlineSharedElementProperties** record.

ItemProperties: An array of properties for the **GaugePanel** record.

labelStart: 0x03 specifies the start of the Label property.

labelValue: *Panel1* specifies the label of the **GaugePanel** report item.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

NonSharedElementProperties: A [NonSharedElementProperties](#) record specifies the properties of the **GaugePanel** record.

nonShared: 0x01 specifies the start token of the **NonSharedElementProperties** record.

ItemProperties: An array of properties for the **GaugePanel** record.

toolTipStart: 0x05 specifies the start token of the Tooltip property.

toolTipValue: *Gauge1* specifies the text that is shown in the tool tip for the **GaugePanel** report item.

delimiter: 0xFF specifies the end of the **NonSharedElementProperties** record.

delimiter: 0xFF specifies the end of the **GaugePanelProperties** record.

ReportElementEnd: A [ReportElementEnd](#) record specifies the end of the **GaugePanel** record.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** element.

offset: 0x0000000000000001 specifies the position of the **gaugePanelToken** field in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

3.5 SubReport

The following example illustrates a [SubReport](#) record with a [ReportName](#) property in an RPL stream.

| Offset | Size | Structure | Value |
|--------|-------|--------------------------------------|--------------------|
| | 005A | SubReport | |
| 0x01 | 0001 | BYTE - subReportToken | 0x0C |
| | 0014 | SubReportProperties | |
| 0x02 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0012 | SharedElementProperties | |
| 0x03 | 0001 | BYTE - shared | 0x00 |
| | 0010 | ItemProperties | |
| 0x04 | 0001 | BYTE - reportNameStart | 0x0F |
| 0x05 | 000F | STRING - reportNameValue | report1 |
| 0x14 | 0001 | BYTE - delimiter | 0xFF |
| 0x15 | 0001 | BYTE - delimiter | 0xFF |
| | 0011 | ReportItems | |
| 0x16 | 0011 | Line | |
| | 002A | Measurements | |
| 0x27 | 0001 | BYTE - measurementStart | 0x10 |
| 0x28 | 0008 | INT - offset | 0x0000000000000001 |
| 0x30 | 0004 | INT - count | 0x00000001 |
| | 001D | MeasurementsContent | |
| 0x34 | 0004 | FLOAT - left | 0x00000000 |
| 0x38 | 0004 | FLOAT - top | 0x00000000 |
| 0x3C | 0004 | FLOAT - width | 0X00000098 |
| 0x40 | 0004 | FLOAT - height | 0x00000019 |
| 0x44 | 0004 | INT - zIndex | 0x00000000 |
| 0x48 | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |

| Offset | Size | Structure | Value |
|--------|-------|-------------------------------------|--------------------|
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x49 | 0008 | INT - reportElementEndOffset | 0X000000000000001D |
| | 000A | ReportElementEnd | |
| 0x51 | 0001 | BYTE - elementEnd | 0xFE |
| 0x52 | 0008 | INT - offset | 0x0000000000000027 |
| 0x5A | 0001 | BYTE - delimiter | 0xFF |

SubReport: A SubReport record that specifies the **SubReport** report item.

subReportToken: 0x0C specifies the start token of the **SubReport** record.

subReportProperties: An [ElementProperties](#) record that specifies the properties for the **SubReport** report item.

elementPropertiesStart: 0x0F specifies the start of the **ElementProperties** record.

SharedElementProperties: A [SharedElementProperties](#) record specifies the [InlineSharedElementProperties](#) record of the **SubReport** record.

shared: 0x00 specifies the start token of the **SharedElementProperties** record.

ItemProperties: An array of properties for the **SubReport** record.

reportNameStart: 0x0F specifies the start of ReportName property.

reportNameValue: *report1* specifies the name of the **SubReport** report item.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

delimiter: 0xFF specifies the end of the **SubReportProperties** record.

ReportItems: A [ReportItem](#) record specifies an array of report items.

Line: A [Line](#) record specifies a **Line** report item in the **SubReport** report item. See the Line example (section [3.1](#)) for more details.

Measurements: A [Measurements](#) record specifies size, location, and state properties for the **ReportItems** record.

measurementsStart: 0x10 specifies the start token of the **Measurements** record.

offset: 0x0000000000000001 specifies the start of the **SubReport** record in the RPL stream.

count: 0x00000001 specifies one **Measurement** record.

measurementsContent (variable): A **MeasurementsContent** structure that specifies an array of [Measurement](#) structures.

left: 0x00000000 specifies that the left position of the line is 0 millimeters relative to its parent record.

top: 0x00000000 specifies that the top position of the line is 0 millimeters relative to its parent record.

width: 0x00000098 specifies that the line is 152 millimeters (6 inches) wide.

height: 0x00000019 specifies that the line is 25.5 millimeters (1 inch) tall.

zIndex: 0x00000001 specifies that the line is drawn first.

state: 0x0C specifies the start token of the [State](#) property.

reportElementEndOffset: 0x000000000000001D specifies the **offset** of the **ReportElementEnd** record for the **Line** record in the RPL stream.

ReportElementEnd: A [ReportElementEnd](#) structure specifies the end of the **SubReport** record.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x0000000000000027 specifies the position of the **Measurements** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

3.6 RichTextBox

The following example illustrates a [RichTextBox](#) record with a single paragraph and two text runs in an RPL stream.

| Offset | Size | Structure | Value |
|--------|------|--------------------------------------|-------|
| | 00C2 | RichTextBox | |
| 0x01 | 0001 | BYTE - richTextBoxToken | 0x07 |
| | 0006 | RichTextBoxProperties | |
| 0x02 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0004 | SharedElementProperties | |
| 0x03 | 0001 | BYTE - shared | 0x00 |
| | 0002 | ItemProperties | |
| 0x04 | 0001 | BYTE - canGrowStart | 0x19 |
| 0x05 | 0001 | BYTE - canGrowValue | 0x00 |
| 0x06 | 0001 | BYTE - delimiter | 0xFF |
| 0x07 | 0001 | BYTE - delimiter | 0xFF |

| Offset | Size | Structure | Value |
|--------|------|--------------------------------------|---------------|
| | 009B | Paragraphs | |
| | 0034 | TextRun | |
| 0x08 | 0001 | BYTE - textRunStart | 0x14 |
| | 0032 | TextRunProperties | |
| 0x09 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0012 | SharedElementProperties | |
| 0x0A | 0001 | BYTE - shared | 0x00 |
| | 0010 | ItemProperties | |
| 0x0B | 0001 | BYTE - labelStart | 0x03 |
| 0x0C | 000F | STRING - labelValue | txtRun1 |
| 0x1B | 0001 | BYTE - delimiter | 0xFF |
| | 001E | NonSharedElementProperties | |
| 0x1C | 0001 | BYTE - nonShared | 0x01 |
| | 001C | ItemProperties | |
| 0x1D | 0001 | BYTE - valueStart | 0x19 |
| 0x1E | 001B | STRING - valueValue | first textrun |
| 0x39 | 0001 | BYTE - delimiter | 0xFF |
| 0x3A | 0001 | BYTE - delimiter | 0xFF |
| 0x3B | 0001 | BYTE - delimiter | 0xFF |
| | 0036 | TextRun | |
| 0x3C | 0001 | BYTE - textRunStart | 0x14 |
| | 0034 | TextRunProperties | |
| 0x3D | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0012 | SharedElementProperties | |
| 0x3E | 0001 | BYTE - shared | 0x00 |
| | 0010 | ItemProperties | |
| 0x3F | 0001 | BYTE - labelStart | 0x03 |
| 0x40 | 000F | STRING - labelValue | txtRun2 |
| 0x4F | 0001 | BYTE - delimiter | 0xFF |
| | 0020 | NonSharedElementProperties | |

| Offset | Size | Structure | Value |
|--------|------|--------------------------------------|--------------------|
| 0x50 | 0001 | BYTE - nonShared | 0x01 |
| | 001E | ItemProperties | |
| 0x51 | 0001 | BYTE - valueStart | 0x19 |
| 0x52 | 001D | STRING - valueValue | second textrun |
| 0x6F | 0001 | BYTE - delimiter | 0xFF |
| 0x70 | 0001 | BYTE - delimiter | 0xFF |
| 0x71 | 0001 | BYTE - delimiter | 0xFF |
| | 0031 | Paragraph | |
| 0x72 | 0001 | BYTE - paragraphStart | 0x13 |
| | 001B | ParagraphProperties | |
| 0x73 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0012 | SharedElementProperties | |
| 0x74 | 0001 | BYTE - shared | 0x00 |
| | 0010 | ItemProperties | |
| 0x75 | 0001 | BYTE - labelStart | 0x03 |
| 0x76 | 000F | STRING - labelValue | paragr1 |
| 0x85 | 0001 | BYTE - delimiter | 0xFF |
| | 0007 | NonSharedElementProperties | |
| 0x86 | 0001 | BYTE - nonShared | 0x01 |
| | 0005 | ItemProperties | |
| 0x87 | 0001 | BYTE - paragraphNumberStart | 0x0E |
| 0x88 | 0004 | INT - paragraphNumberValue | 0x00000001 |
| 0x8C | 0001 | BYTE - delimiter | 0xFF |
| 0x8D | 0001 | BYTE - delimiter | 0xFF |
| 0x8E | 0004 | INT - count | 0x00000002 |
| | 0010 | OffsetsTextRun | |
| 0x92 | 0008 | INT - offset1 | 0x0000000000000008 |
| 0x9A | 0008 | INT - offset2 | 0x000000000000003C |
| 0xA2 | 0001 | BYTE - delimiter | 0xFF |
| | 0016 | RichTextBoxStructure | |

| Offset | Size | Structure | Value |
|--------|------|---|--------------------|
| 0xA3 | 0001 | BYTE - richTextBoxStructureStart | 0x12 |
| 0xA4 | 0008 | INT - offset | 0x0000000000000001 |
| 0xAC | 0004 | INT - count | 0x00000001 |
| | 0008 | OffsetsParagraphContent | |
| 0xB0 | 0008 | INT - offset1 | 0x0000000000000072 |
| 0xB8 | 0001 | BYTE - delimiter | 0xFF |
| | 000A | ReportElementEnd | |
| 0xB9 | 0001 | BYTE - elementEnd | 0xFE |
| 0xBA | 0008 | INT - offset | 0x00000000000000A3 |
| 0xC2 | 0001 | BYTE - delimiter | 0xFF |

RichTextBox: A **RichTextBox** record that specifies the text box report item.

richTextBoxToken: 0x07 specifies the start token of the **RichTextBox** record.

RichTextBoxProperties: An [ElementProperties](#) record that specifies the properties for the **RichTextBox** report item.

elementPropertiesStart: 0x0F specifies the start token of the **RichTextBoxProperties** record.

SharedElementProperties: A [SharedElementProperties](#) record specifies the [InlineSharedElementProperties](#) record of the **RichTextBox** record.

shared: 0x00 specifies the start token for the **SharedElementProperties** record.

ItemProperties: An array of properties for the **RichTextBox** record.

canGrowStart: 0x19 specifies the start token of the [CanGrow](#) property.

canGrowValue: 0x00 specifies a false value, which indicates that the height of the **RichTextBox** cannot increase.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

delimiter: 0xFF specifies the end of the **RichTextBoxProperties** record.

delimiter: 0xFF specifies the end of the [TextRun](#) record.

Paragraphs: An array of [ParagraphContent](#) records specifies the contents of the **RichTextBox** record.

TextRun: A **TextRun** structure specifies the contents and properties of a span of text.

textRunStart: 0x14 specifies the start token of the **TextRun** record.

TextRunProperties: An **ElementProperties** record specifies the properties of the **TextRun** record.

elementPropertiesStart: 0x0F specifies the start token of the **TextRunProperties** record.

SharedElementProperties: A **SharedElementProperties** record that specifies the **InlineSharedElementProperties** record of the **TextRun** record.

shared: 0x00 specifies the start token of the **InlineSharedElementProperties** record.

ItemProperties: An array of properties for the **TextRun** record.

labelStart: 0x03 specifies the start of the **Label** property.

labelValue: *txtRun1* specifies the label of the **TextRun** record.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

NonSharedElementProperties: A [NonSharedElementProperties](#) record specifies the properties of the **TextRun** record.

nonShared: 0x01 specifies the start token of the **NonSharedElementProperties** record.

ItemProperties: An array of properties for the **TextRun** record.

valueStart: 0x0A specifies the start token of the [Value](#) property.

valueValue: *first textrun* specifies the text to show in the **RichTextBox** record.

delimiter: 0xFF specifies the end of the **NonSharedElementProperties** record.

delimiter: 0xFF specifies the end of the **TextRunProperties** record.

delimiter: 0xFF specifies the end of the **TextRun** record.

TextRun: A **TextRun** record specifies the contents and properties of a span of text.

textRunStart: 0x14 specifies the start token of the **TextRun** record.

TextRunProperties: An **ElementProperties** record specifies the properties of the **TextRun** record.

elementPropertiesStart: 0x0F specifies the start token of the **TextRunProperties** record.

SharedElementProperties: A **SharedElementProperties** record that specifies the **InlineSharedElementProperties** record of the **TextRun** record.

shared: 0x00 specifies the start token of the **InlineSharedElementProperties** record.

ItemProperties: An array of properties for the **TextRun** record.

labelStart: 0x03 specifies the start of the **Label** property.

labelValue: *txtRun2* specifies the label of the **TextRun** record.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

NonSharedElementProperties: A **NonSharedElementProperties** structure specifies the properties of the **TextRun** record.

nonShared: 0x01 specifies the start token of the **NonSharedElementProperties** record.

ItemProperties: An array of properties for the **TextRun** record.

valueStart: 0x0A specifies the start token of the **Value** property.

valueValue: *second textrun* specifies the text to be shown in the **RichTextBox** record.

delimiter: 0xFF specifies the end of the **NonSharedElementProperties** record.

delimiter: 0xFF specifies the end of the **TextRunProperties** record.

delimiter: 0xFF specifies the end of the **TextRun** record.

Paragraph: A [Paragraph](#) record specifies the properties and layout of a **RichTextBox** record.

paragraphStart: 0x13 specifies the start token of the **Paragraph** record.

ParagraphProperties: An **ElementProperties** record that specifies the properties for the **Paragraph** record.

elementPropertiesStart: 0x0F specifies the start token of the **ParagraphProperties** record.

SharedElementProperties: A **SharedElementProperties** record that specifies the **InlineSharedElementProperties** record of the **Paragraph** record.

shared: 0x00 specifies the start token of the **InlineSharedElementProperties** record.

ItemProperties: An array of properties for the **Paragraph** record.

labelStart: 0x03 specifies the start of the **Label** property.

labelValue: *paragr1* specifies the label of the **Paragraph** record.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

NonSharedElementProperties: A **NonSharedElementProperties** record specifies the properties of the **Paragraph** record.

NonShared: 0x01 specifies the start token of the **NonSharedElementProperties** record.

ItemProperties: An array of properties for the **Paragraph** record.

paragraphNumberStart: 0x0E specifies the start token of the [ParagraphNumber](#) property.

paragraphNumberValue: 0x00000001 specifies that 1 is the number of this paragraph.

delimiter: 0xFF specifies the end of the **NonSharedElementProperties** record.

delimiter: 0xFF specifies the end of the **ParagraphProperties** record.

count: 0x00000002 specifies that there are two **TextRun** records present in the **ParagraphContent** record.

OffsetsTextRun: An array of offsets pointing to the beginning of each **TextRun** record in the RPL stream.

offset1: 0x0000000000000008 specifies the position of the beginning of the first **TextRun** record.

offset2: 0x000000000000003C specifies the position of the beginning of the second **TextRun** record.

delimiter: 0xFF specifies the end of the **Paragraph** record.

RichTextBoxStructure: A [RichTextBoxStructure](#) record specifies the structure of the text box.

richTextBoxStructureStart: 0x12 specifies the start token of the **RichTextBoxStructure** record.

offset: 0x0000000000000001 specifies the position of the **RichTextBox** record in the RPL stream.

count: 0x00000001 specifies that one **ParagraphContent** record is present in the **RichTextBox** record.

offsetsParagraphContent: An array of offsets pointing to the beginning of each **Paragraph** record in the RPL stream.

offset1: 0x0000000000000072 specifies the position of the first **Paragraph** record in the RPL stream.

delimiter: 0xFF specifies the end of the **RichTextBoxStructure** record.

ReportElementEnd: A [ReportElementEnd](#) record specifies the end of the **RichTextBox** record.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x00000000000000A3 specifies the position of the **RichTextBoxStructureStart** token in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

3.7 Tablix

The following example illustrates a [Tablix](#) record with two rows and three columns. The first row consists of a [TablixComer](#) record and two [TablixColumnHeader](#) records, and the second row consists of a [TablixRowHeader](#) record and two [TablixBodyCell](#) records, each containing a [Line](#) record and an [Image](#) record.

| Offset | Size | Structure | Value |
|--------|------|--------------------------------------|---------|
| | 0093 | Tablix | |
| 0x01 | 0001 | BYTE - tablixToken | 0x0D |
| | 0014 | tablixProperties | |
| 0x02 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0012 | SharedElementProperties | |
| 0x03 | 0001 | BYTE - shared | 0x00 |
| | 0010 | ItemProperties | |
| 0x04 | 0001 | BYTE - nameStart | 0x02 |
| 0x05 | 000F | BYTE - nameStart | 0x02 |
| 0x14 | 0001 | STRING - nameValue | tablix1 |
| 0x15 | 0001 | BYTE - delimiter | 0xFF |

| Offset | Size | Structure | Value |
|--------|-------|-------------------------------------|---------------------|
| | 00A1 | BYTE - delimiter | |
| | 0003E | TablixContent | |
| | 0022 | ReportItems | |
| 0x16 | 0011 | Line | |
| 0x27 | 0011 | Image | |
| | 001C | BYTE - tablixBodyRowStart | 0x12 |
| 00x39 | 0004 | INT - rowIndex | 0x00000001 |
| | 0016 | TablixBodyCells | |
| 0x3D | 0001 | BYTE - tablixBodyCellStart | 0x0D |
| | 0009 | TablixCellProperties | |
| 0x3E | 0001 | BYTE - cellItemOffsetStart | 0x04 |
| 0x3F | 0008 | INT - cellItemOffsetValue | 0x0000000000000001D |
| 0x47 | 0001 | BYTE - delimiter | 0xFF |
| 0x48 | 0001 | BYTE - tablixBodyCellStart | 0x0D |
| | 0009 | TablixCellProperties | |
| 0x49 | 0001 | BYTE - cellItemOffsetStart | 0x04 |
| 0x4A | 0008 | INT - cellItemOffsetValue | 0x0000000000000002E |
| 0x52 | 0001 | BYTE - delimiter | 0xFF |
| 0x53 | 0001 | BYTE - delimiter | 0xFF |
| | 0036 | TablixStructure | |
| 00x54 | 0001 | BYTE - tablixStructureStart | 0x11 |
| 0x55 | 0008 | INT - offset | 0x00000000000000001 |
| | 002C | TablixMeasurements | |
| | 0025 | tablixMeasurementsProperties | |
| 0x5D | 0001 | BYTE - layoutDirectionStart | 0x03 |
| 0x5F | 0001 | BYTE - tablixRowStart | 0x08 |
| | 0029 | TablixRows | |
| 0x5F | 0001 | BYTE - tablixRowStart | 0x08 |
| | 0015 | TablixRowCell | |
| | 0007 | TablixCorner | |

| Offset | Size | Structure | Value |
|--------|------|---------------------------------------|------------|
| 0x60 | 0001 | BYTE - tablixCornerStart | 0x0A |
| | 0005 | tablixCellProperties | |
| 0x61 | 0001 | BYTE - colSpanStart | 0x05 |
| 0x62 | 0004 | INT - colSpanValue | 0x00000001 |
| 0x66 | 0001 | BYTE - delimiter | 0xFF |
| | 0007 | TablixColumnHeader | |
| 0x67 | 0001 | BYTE - tablixColumnHeaderStart | 0x0B |
| 0x68 | 0001 | BYTE - colSpanStart | 0x05 |
| 0x69 | 0004 | INT - colSpanValue | 0x00000001 |
| | 0005 | tablixCellProperties | |
| 0x6D | 0001 | BYTE - delimiter | 0xFF |
| | 0001 | BYTE - delimiter | 0xFF |
| | 0007 | TablixColumnHeader | |
| 0x6E | 0001 | BYTE - tablixColumnHeaderStart | 0x0B |
| 0x6E | 0001 | BYTE - tablixColumnHeaderStart | 0x0B |
| | 0005 | tablixCellProperties | |
| 0x6F | 0001 | BYTE - colSpanStart | 0x05 |
| 0x70 | 0004 | INT - colSpanValue | 0x00000001 |
| 00x74 | 0001 | BYTE - delimiter | 0xFF |
| 00x75 | 0001 | BYTE - delimiter | 0xFF |
| 0x76 | 0001 | BYTE - tablixRowStart | 0x08 |
| | 0019 | TablixRowCell | |
| | 0007 | TablixRowHeader | |
| 00x77 | 0001 | BYTE - tablixRowHeaderStart | 0x0 |
| | 0005 | tablixCellProperties | |
| 0x78 | 0001 | BYTE - defIndexStart | 0x08 |
| 0x79 | 0004 | INT - defIndexValue | 0x00000000 |
| 0x7D | 0001 | BYTE - delimiter | 0xFF |
| | 0009 | TablixBodyRowCells | |
| 0x7E | 0001 | BYTE - tablixBodyRowCellsStart | 0x09 |

| Offset | Size | Structure | Value |
|--------------|------|-------------------------|--------------------|
| 0x7F | 0008 | INT -offset | 0x0000000000000038 |
| 0x87 | 0001 | BYTE -delimiter | 0xFF |
| 0x88 | 0001 | BYTE -delimiter | 0xFF |
| 0x89 | 0001 | BYTE -delimiter | 0xFF |
| | 000A | ReportElementEnd | |
| 0x8A | 0001 | BYTE -elementEnd | 0xFE |
| 0x8B 0x9B | 0008 | INT -offset | 0x000000000000005E |
| 0x93 | 0001 | BYTE -delimiter | 0xFF |

Tablix: A **Tablix** record specifies the **Tablix** report item.

tablixToken: 0x0D specifies the start token of the **Tablix** record.

TablixProperties: An [ElementProperties](#) record specifies the properties for the **Tablix** report item.

elementPropertiesStart: 0x0F specifies the start of the **TablixProperties** record.

SharedElementProperties: A [SharedElementProperties](#) structure that specifies the [InlineSharedElementProperties](#) record of the **Tablix** record.

shared: 0x00 specifies the start token of the **SharedElementProperties** record.

ItemProperties: An array of properties for the **Tablix** record.

nameStart: 0x02 specifies the start of the [Name](#) property.

nameValue: *tablix1* specifies the name of the **Tablix** report item.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

delimiter: 0xFF specifies the end of the **TablixProperties** record.

TablixContent: A [TablixContent](#) record specifies the **ReportItems** and [TablixBodyRow](#) records of the **tablix**.

ReportItems: A **ReportItems** record specifies an array of report items containing one line and one image.

Line: A **Line** structure specifies a **Line** report item in the **Tablix** report item. See the **Line** example (section [3.1](#)) for additional details.

Image: An **Image** record specifies a **Line** report item in the **Tablix** report item. See the **Image** example (section [3.2](#)) for additional details.

TablixBodyRow: The **TablixBodyRow** record specifies the content of a row in the **Tablix** record.

tablixBodyRowStart: 0x12 specifies the start token of **TablixBodyRow**.

rowIndex: 0x00000001 specifies that 1 is the index of this row.

TablixBodyCells: The **TablixBodyCells** record specifies an array of **TablixBodyCell** records.

tablixBodyCellStart: 0x0D specifies the start token of **TablixBodyCell**.

tablixCellProperties: An array of properties for the **Tablix** cell.

cellItemOffsetStart: 0x04 specifies the start token of the [CellItemOffset](#) property.

cellItemOffsetValue: 0x000000000000001D specifies the offset pointing to the **ReportElementEnd** record of the **Line** report item.

columnIndexStart: 0x08 specifies the start token of the [ColumnIndex](#) property.

columnIndexValue: 0x00000001 specifies that this cell will appear in the second column.

delimiter: 0xFF specifies the end of the **TablixBodyCell** record.

tablixBodyCellStart: 0x0D specifies the start token of **TablixBodyCell**.

tablixCellProperties: An array of properties for the **Tablix** cell.

cellItemOffsetStart: 0x04 specifies the start token of the **CellItemOffset** property.

cellItemOffsetValue: 0x000000000000002E specifies the offset pointing to the **ReportElementEnd** of the image report item.

columnIndexStart: 0x08 specifies the start token of the **ColumnIndex** property.

columnIndexValue: 0x00000002 specifies that this cell will appear in the third column.

delimiter: 0xFF specifies the end of the **TablixBodyCell** record.

delimiter: 0xFF specifies the end of the **TablixBodyRow** record.

TablixStructure: A [TablixStructure](#) record specifies the measurements of the tablix.

tablixStructureStart: 0x11 specifies the start token of the **TablixStructure** record.

offset: 0x0000000000000001 specifies the position of the **Tablix** record in the RPL stream.

TablixMeasurements: A [TablixMeasurements](#) record specifies the measurements of the tablix.

TablixMeasurementsProperties: A collection of **ItemProperty** records specifies the properties for the tablix measurements.

layoutDirectionStart: 0x03 specifies the start token of the [LayoutDirection](#) property.

layoutDirectionValue: 0x00 specifies that the layout direction of the tablix is left to right.

ColumnsWidths: A **ColumnWidths** record specifies the width of each column.

columnsWidthsStart: 0x04 specifies the start token of the **ColumnsWidths** record.

count: 0x03 specifies there are three columns in this tablix.

Columns: A [Columns](#) record specifies an array of properties for each column.

width: 0x0000CC41 specifies that the first column is 25.5 millimeters wide.

fixed: 0x00 specifies a false value, which indicates that the columns will disappear when part of the tablix is scrolled off the screen.

width: 0x0000CC41 specifies that the second column is 25.5 millimeters wide.

fixed: 0x00 specifies a false value, which indicates that the columns will disappear when part of the tablix is scrolled off the screen.

width: 0x0000CC41 specifies that the third column is 25.5 millimeters wide.

fixed: 0x00 specifies a false value, which indicates that the columns will disappear when part of the tablix is scrolled off the screen.

RowHeights: A [RowHeights](#) record specifies the heights of each row.

RowHeightsStart: 0x05 specifies the start token of the **RowHeights** property.

count: 0x02 specifies that there are two rows in this tablix.

height: 0x0000CC41 specifies that the first row is 25.5 millimeters tall.

fixedRows: 0x0 specifies a false value, which indicates the row will disappear when part of the tablix is scrolled off the screen.

sharedLayoutRows: 0x0 specifies a false value, which indicates that the layout cannot be shared by another row.

useSharedLayoutRows: 0x0 specifies a false value, which indicates that this row is not using the same layout as a previous row.

reserved: 0x0 specifies the field is 0 and must be ignored.

height: 0x0000CC41 specifies that the second row is 25.5 millimeters tall.

fixedRows: 0x0 specifies a false value, which indicates that the row will disappear when part of the tablix is scrolled off the screen.

sharedLayoutRows: 0x0 specifies a false value, which indicates that the layout cannot be shared by another row.

useSharedLayoutRows: 0x0 specifies a false value, which indicates that this row does not use the same layout as a previous row.

reserved: 0x0 specifies the field is 0 and must be ignored.

TablixRows: A **TablixRows** structure specifies an array of the [TablixRow](#) record.

tablixRowStart: 0x08 specifies the start token of the tablix row.

TablixRowCells: A **TablixRowCells** record specifies an array of the [TablixRowCell](#) record.

TablixCorner: A **TablixCorner** record specifies the contents of the tablix corner.

tablixCornerStart: 0x0A specifies the start token of **TablixCorner**.

tablixCellProperties: An array of properties for the tablix corner cell.

columnIndexStart: 0x08 specifies the start token of the **columnIndex** property.

columnIndexValue: 0x00000000 specifies this cell will appear in the first column.

delimiter: 0xFF specifies the end of **TablixCorner**.

TablixColumnHeader: A **TablixColumnHeader** record specifies the contents of the tablix header.

tablixColumnHeaderStart: 0x0B specifies the start token of **TablixColumnHeader**.

tablixCellProperties: An array of properties for the tablix column header cell.

columnIndexStart: 0x08 specifies the start token of the **columnIndex** property.

columnIndexValue: 0x00000001 specifies that this cell will appear in the second column.

delimiter: 0xFF specifies the end of **TablixColumnHeader**.

TablixColumnHeader: A **TablixColumnHeader** record specifies the contents of the tablix header.

tablixColumnHeaderStart: 0x0B specifies the start token of **TablixColumnHeader**.

tablixCellProperties: A collection of **ItemProperty** records specifies the properties for the tablix column header cell.

columnIndexStart: 0x08 specifies the start token of the **columnIndex** property.

columnIndexValue: 0x00000002 specifies that this cell will appear in the third column.

delimiter: 0xFF specifies the end of the **TablixColumnHeader** record.

delimiter: 0xFF specifies the end of the **TablixRow** record.

tablixRowStart: 0x08 specifies the start token of **TablixRow**.

TablixRowCells: A **TablixRowCells** record specifies an array of **TablixRowCell**.

TablixRowHeader: A **TablixRowHeader** record specifies the contents of the tablix header.

tablixRowHeaderStart: 0x0C specifies the start token of **TablixRowHeader**.

tablixCellProperties: An array of properties for the tablix row header cell.

columnIndexStart: 0x08 specifies the start token of the **columnIndex** property.

columnIndexValue: 0x00000000 specifies that this cell will appear in the first column.

delimiter: 0xFF specifies the end of the **TablixRowHeader** record.

TablixBodyRowCells: A **TablixBodyRowCell** record specifies the contents of the tablix body row cells.

tablixBodyRowCellsStart: 0x09 specifies the start token of the **TablixBodyRowCells** record.

offset: 0x0000000000000038 specifies the position of the **TablixBodyRow** record in the RPL stream.

delimiter: 0xFF specifies the end of the **TablixRows** record.

delimiter: 0xFF specifies the end of the **TablixMeasurements** record.

delimiter: 0xFF specifies the end of the **TablixStructure** record.

ReportElementEnd: A [ReportElementEnd](#) record specifies the end of the tablix record.

elementEnd: 0xFE specifies the end of the **Tablix** record.

offset: 0x000000000000005E specifies the position of the **TablixStructureStart** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

3.8 Report

This example specifies the records that are required to produce a [Report](#) by using RPL.

| Offset | Size | Structure | Value |
|--------|------|------------------------------------|------------|
| | 0240 | Report | |
| 0x1 | 000B | STRING- RPLStamp | RPLIF |
| | 0006 | Version | |
| 0xC | 0001 | BYTE- majorVersion | 0x0A |
| 0xD | 0001 | BYTE- minorVersion | 0x04 |
| 0xE | 0004 | INT- build | 0x00000000 |
| 0x12 | 0001 | BYTE- reportStart | 0x00 |
| | 0017 | ReportProperties | |
| 0x13 | 0001 | BYTE- reportPropertiesStart | 0x02 |
| | 0015 | ItemProperties | |
| 0x14 | 0001 | BYTE - reportNameStart | 0x0F |
| 0x15 | 000F | STRING - reportNameValue | Report1 |
| 0x24 | 0001 | BYTE - autoRefreshStart | 0x0E |
| 0x25 | 0004 | INT - autoRefreshValue | 0x0000001E |
| 0x29 | 0001 | BYTE - delimiter | 0xFF |
| | 01F2 | Pages | |
| 0x2A | 0001 | BYTE- pageContentStart | 0x13 |
| | 000C | PageLayout | |
| 0x2B | 0001 | BYTE - pagePropertiesStart | 0x03 |
| | 000A | ItemProperties | |
| 0x2C | 0001 | BYTE - pageHeightStart | 0x10 |
| 0x2D | 0004 | FLOAT - pageHeightValue | 0X00808B43 |
| 0x31 | 0001 | BYTE - pageWidthStart | 0x11 |

| Offset | Size | Structure | Value |
|--------|------|--------------------------------------|--------------------|
| 0x32 | 0004 | FLOAT - pageWidthValue | 0X00005843 |
| 0x36 | 0001 | BYTE - delimiter | 0xFF |
| | 01B1 | Sections | |
| 0x37 | 0001 | BYTE - sectionStart | 0x15 |
| | 0019 | SectionProperties | |
| 0x38 | 0001 | BYTE - sectionPropertiesStart | 0x16 |
| | 0017 | ItemProperties | |
| 0x39 | 0001 | BYTE - idStart | 0x00 |
| 0x3A | 0011 | STRING - idValue | Section1 |
| 0x4B | 0001 | BYTE - columnCountStart | 0x01 |
| 0x4C | 0004 | INT - columnCountValue | 0x00000001 |
| 0x50 | 0001 | BYTE - delimiter | 0xFF |
| | 0091 | BodyArea Element | |
| 0x51 | 0001 | BYTE - columns | 0x14 |
| | 005C | BodyArea | |
| 0x52 | 0001 | BYTE - body | 0x06 |
| | 0016 | ElementProperties | |
| 0x53 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0014 | SharedElementProperties | |
| 0x54 | 0001 | BYTE - shared | 0x00 |
| | 0012 | ItemProperties | |
| 0x55 | 0001 | BYTE - idStart | 0x00 |
| 0x56 | 0011 | STRING - idValue | BodyArea |
| 0x67 | 0001 | BYTE - delimiter | 0xFF |
| 0x68 | 0001 | BYTE - delimiter | 0xFF |
| | 0011 | ReportItems | |
| 0x69 | 0011 | Image | |
| | 002A | Measurements | |
| 0x7A | 0001 | BYTE - measurementsStart | 0x10 |
| 0x7B | 0008 | INT- offset | 0x0000000000000052 |

| Offset | Size | Structure | Value |
|--------|-------|-------------------------------------|--------------------|
| 0x83 | 0004 | INT- count | 0x00000001 |
| | 001D | measurementsContent | |
| 0x87 | 0004 | FLOAT - left | 0x00000000 |
| 0x8B | 0004 | FLOAT - top | 0x00000000 |
| 0x8F | 0004 | FLOAT - width | 0x00001843 |
| 0x93 | 0004 | FLOAT - height | 0x00003243 |
| 0x97 | 0004 | INT - zIndex | 0x00000000 |
| 0x9B | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x9C | 0008 | INT - reportElementEndOffset | 0x0000000000000070 |
| | 000A | ReportElementEnd | |
| 0xA4 | 0001 | BYTE - elementEnd | 0xFE |
| 0xA5 | 0008 | INT- offset | 0x000000000000007A |
| 0xAD | 0001 | BYTE - delimiter | 0xFF |
| | 002A | Measurements | |
| 0xAE | 0001 | BYTE - measurementsStart | 0x10 |
| 0xAF | 0008 | INT- offset | 0x0000000000000051 |
| 0xB7 | 0004 | INT- count | 0x00000001 |
| | 001D | measurementsContent | |
| 0xBB | 0004 | FLOAT - left | 0x00000000 |
| 0xBF | 0004 | FLOAT - top | 0x0000CC41 |
| 0xC3 | 0004 | FLOAT - width | 0x00001843 |
| 0xC7 | 0004 | FLOAT - height | 0x00003243 |

| Offset | Size | Structure | Value |
|--------|-------|--------------------------------------|--------------------|
| 0xCB | 0004 | INT - zIndex | 0x00000000 |
| 0xCF | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0xD0 | 0008 | INT - reportElementEndOffset | 0x00000000000000A4 |
| | 000A | ReportElementEnd | |
| 0xD8 | 0001 | BYTE - elementEnd | 0xFE |
| 0xD9 | 0008 | INT- offset | 0x00000000000000AE |
| 0xE1 | 0001 | BYTE - delimiter | 0xFF |
| | 004C | PageFooterElement | |
| 0xE2 | 0001 | BYTE - pageFooter | 0x05 |
| | 0006 | ElementProperties | |
| 0xE3 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0004 | SharedElementProperties | |
| 0xE4 | 0001 | BYTE - shared | 0x00 |
| | 0002 | ItemProperties | |
| 0xE5 | 0001 | BYTE - printOnFirstPageStart | 0x2C |
| 0xE6 | 0001 | BYTE - printOnFirstPageValue | 0x00 |
| 0xE7 | 0001 | BYTE - delimiter | 0xFF |
| 0xE8 | 0001 | BYTE - delimiter | 0xFF |
| | 0011 | ReportItems | |
| 0xE9 | 0011 | Line | |
| | 002A | Measurements | |
| 0xFA | 0001 | BYTE - measurementsStart | 0x10 |

| Offset | Size | Structure | Value |
|--------|-------|--------------------------------------|--------------------|
| 0xFB | 0008 | INT- offset | 0x00000000000000E2 |
| 0x103 | 0004 | INT- count | 0x00000001 |
| | 001D | measurementsContent | |
| 0x107 | 0004 | FLOAT - left | 0x00000000 |
| 0x10B | 0004 | FLOAT - top | 0x00000000 |
| 0x10F | 0004 | FLOAT - width | 0X00001843 |
| 0x113 | 0004 | FLOAT - height | 0x0000CC41 |
| 0x117 | 0004 | INT - zIndex | 0x00000000 |
| 0x11B | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x11C | 0008 | INT - reportElementEndOffset | 0x00000000000000F0 |
| | 000A | ReportElementEnd | |
| 0x124 | 0001 | BYTE - elementEnd | 0xFE |
| 0x125 | 0008 | INT- offset | 0x00000000000000FA |
| 0x12D | 0001 | BYTE - delimiter | 0xFF |
| | 004C | PageHeaderElement | |
| 0x12E | 0001 | BYTE - pageHeader | 0x04 |
| | 0006 | ElementProperties | |
| 0x12F | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0004 | SharedElementProperties | |
| 0x130 | 0001 | BYTE - shared | 0x00 |
| | 0002 | ItemProperties | |
| 0x131 | 0001 | BYTE - printOnFirstPageStart | 0x2C |

| Offset | Size | Structure | Value |
|--------|-------|-------------------------------------|--------------------|
| 0x132 | 0001 | BYTE - printOnFirstPageValue | 0x01 |
| 0x133 | 0001 | BYTE - delimiter | 0xFF |
| 0x134 | 0001 | BYTE - delimiter | 0xFF |
| | 0011 | ReportItems | |
| 0x135 | 0011 | Line | |
| | 002A | Measurements | |
| 0x146 | 0001 | BYTE - measurementsStart | 0x10 |
| 0x147 | 0008 | INT- offset | 0x000000000000012E |
| 0x14F | 0004 | INT- count | 0x00000001 |
| | 001D | measurementsContent | |
| 0x153 | 0004 | FLOAT - left | 0x00000000 |
| 0x157 | 0004 | FLOAT - top | 0x00000000 |
| 0x15B | 0004 | FLOAT - width | 0x00001843 |
| 0x15F | 0004 | FLOAT - height | 0x0000CC41 |
| 0x163 | 0004 | INT - zIndex | 0x00000000 |
| 0x167 | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x168 | 0008 | INT - reportElementEndOffset | 0x000000000000013C |
| | 000A | ReportElementEnd | |
| 0x170 | 0001 | BYTE - elementEnd | 0xFE |
| 0x171 | 0008 | INT- offset | 0x0000000000000146 |
| 0x179 | 0001 | BYTE - delimiter | 0xFF |
| | 0064 | Measurements | |

| Offset | Size | Structure | Value |
|--------|-------|-------------------------------------|--------------------|
| 0x17A | 0001 | BYTE - measurementsStart | 0x10 |
| 0x17B | 0008 | INT- offset | 0x0000000000000037 |
| 0x183 | 0004 | INT- count | 0x00000003 |
| | 0057 | measurementsContent | |
| 0x187 | 0004 | FLOAT - left | 0x00000000 |
| 0x18B | 0004 | FLOAT - top | 0X0000CC41 |
| 0x18F | 0004 | FLOAT - width | 0x00001843 |
| 0x193 | 0004 | FLOAT - height | 0x00003243 |
| 0x197 | 0004 | INT - zIndex | 0x00000000 |
| 0x19B | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x19C | 0008 | INT - reportElementEndOffset | 0x00000000000000D8 |
| 0x1A4 | 0004 | FLOAT - left | 0x00000000 |
| 0x1A8 | 0004 | FLOAT - top | 0X00804B43 |
| 0x1AC | 0004 | FLOAT - width | 0x00001843 |
| 0x1B0 | 0004 | FLOAT - height | 0x0000CC41 |
| 0x1B4 | 0004 | INT - zIndex | 0x00000001 |
| 0x1B8 | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |

| Offset | Size | Structure | Value |
|--------|-------|-------------------------------------|--------------------|
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x1B9 | 0008 | INT - reportElementEndOffset | 0x0000000000000124 |
| 0x1C1 | 0004 | FLOAT - left | 0x00000000 |
| 0x1C5 | 0004 | FLOAT - top | 0X00000000 |
| 0x1C9 | 0004 | FLOAT - width | 0x00001843 |
| 0x1CD | 0004 | FLOAT - height | 0x0000CC41 |
| 0x1D1 | 0004 | INT - zIndex | 0x00000002 |
| 0x1D5 | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x1D6 | 0008 | INT - reportElementEndOffset | 0x0000000000000170 |
| | 000A | ReportElementEnd | |
| 0x1DE | 0001 | BYTE - elementEnd | 0xFE |
| 0x1DF | 0008 | INT- offset | 0x000000000000017A |
| 0x1E7 | 0001 | BYTE - delimiter | 0xFF |
| | 002A | Measurements | |
| 0x1E8 | 0001 | BYTE - measurementsStart | 0x10 |
| 0x1E9 | 0008 | INT- offset | 0x000000000000002A |
| 0x1F1 | 0004 | INT- count | 0x00000001 |
| | 001D | measurementsContent | |
| 0x1F5 | 0004 | FLOAT - left | 0x00000000 |
| 0x1F9 | 0004 | FLOAT - top | 0x00000000 |

| Offset | Size | Structure | Value |
|--------|-------|---------------------------------------|---------------------|
| 0x1FD | 0004 | FLOAT - width | 0x00001843 |
| 0x201 | 0004 | FLOAT - height | 0x00006543 |
| 0x205 | 0004 | INT - zIndex | 0x00000000 |
| 0x209 | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x20A | 0008 | INT - reportElementEndOffset | 0x000000000000001DE |
| | 000A | ReportElementEnd | |
| 0x212 | 0001 | BYTE - elementEnd | 0xFE |
| 0x213 | 0008 | INT- offset | 0x000000000000001E8 |
| 0x21B | 0001 | BYTE - delimiter | 0xFF |
| | 0015 | OffsetsArrayElement | |
| 0x21C | 0001 | BYTE- offsetsArrayElementStart | 0x12 |
| 0x21D | 0008 | INT- offset | 0x00000000000000012 |
| 0x225 | 0004 | INT- count | 0x00000001 |
| | 0008 | OffsetsPageContent | |
| 0x229 | 0008 | INT- offset | 0x00000000000000212 |
| | 000A | ReportElementEnd | |
| 0x231 | 0001 | BYTE - elementEnd | 0xFE |
| 0x232 | 0008 | INT- offset | 0x0000000000000021C |
| 0x23A | 0001 | BYTE - delimiter | 0xFF |
| | 0006 | Version | |
| 0x23B | 0001 | BYTE- majorVersion | 0x0A |
| 0x23C | 0001 | BYTE- minorVersion | 0x04 |

| Offset | Size | Structure | Value |
|--------|------|-------------------|------------|
| 0x23D | 0004 | INT- build | 0x00000000 |

Report: A Report record specifies the contents of the report.

RPLStamp: *RPLIF* specifies the value for the stamp.

Version: A [Version](#) record specifies the RPL version.

majorVersion: 0x0A specifies that the major version is 10.

minorVersion: 0x04 specifies that the minor version is 4.

build: 0x00000000 specifies that the build is 0.

reportStart: 0x00 specifies the start token of the **Report** record.

ReportProperties: A [ReportProperties](#) record specifies properties of the report.

reportPropertiesStart: 0x02 specifies the start token of the **ReportProperties** record.

ItemProperties: An array of item properties for the **Report** record.

reportNameStart: 0x0F specifies the start token of the [ReportName](#) property.

reportNameValue: *Report1* specifies the name of the report.

autoRefreshStart: 0x0E specifies the start token of the [AutoRefresh](#) property.

autoRefreshValue: 0x0000001E specifies that the automatic refresh interval is 30 seconds.

delimiter: 0xFF specifies the end of the **ReportProperties** record.

Pages: A [PageContent](#) record specifies the pages of a report.

pageContentStart: 0x13 specifies the start token of the **PageContent** record.

PageLayout: A [PageLayout](#) record specifies the layout of a page.

pagePropertiesStart: 0x03 specifies the start token of the **PageLayout** record.

ItemProperties: An array of properties for the **Page** record.

pageHeightStart: 0x10 specifies the start token of the **PageHeight** property.

pageHeightValue: 0x00808B43 specifies that the height of the page is 279 millimeters.

pageWidthStart: 0x11 specifies the start token of the **PageWidth** property.

pageWidthValue: 0x00005843 specifies that the width of the page is 216 millimeters.

delimiter: 0xFF specifies the end of the **PageLayout** record.

Sections: A [Section](#) record specifies the sections of a page.

sectionStart: 0x15 specifies the start token of the **Sections** record.

SectionProperties: A [SectionProperties](#) record specifies the properties of a section.

sectionPropertiesStart: 0x16 specifies the start token of the **SectionProperties** structure.

ItemProperties: An array of properties for the **Sections** record.

idStart: 0x00 specifies the start token of the [ID](#) property.

idValue: *Section1* specifies the identifier for the section.

columnCountStart: 0x01 specifies the start token of the [ColumnCount](#) property.

columnCountValue: 0x00000001 specifies that there is one column in this section.

delimiter: 0xFF specifies the end of the **SectionProperties** record.

BodyAreaElement: A [BodyAreaElement](#) record specifies an area of the body of a section.

Columns: 0x14 specifies the start token of the **BodyAreaElement** record.

BodyArea: A [BodyElement](#) record specifies the content of the main area of a **BodyAreaElement**.

body: 0x06 specifies the start token of the **BodyArea** record.

ElementProperties: An [ElementProperties](#) record that specifies the properties for the **BodyArea** record.

elementPropertiesStart: 0x0F specifies the start token of the **ElementProperties** record.

SharedElementProperties: A [SharedElementProperties](#) record that specifies the [InlineSharedElementProperties](#) record of the **BodyArea** record.

shared: 0x00 specifies the start token of the **SharedElementProperties** record.

ItemProperties: An array of properties for the **BodyAreaElement** record.

idStart: 0x00 specifies the start token of the ID property.

idValue: *BodyArea* specifies the **ID** property for the **BodyArea** record.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

delimiter: 0xFF specifies the end of the **ElementProperties** record.

ReportItems: A [ReportItem](#) record specifies items in the report.

Image: An [Image](#) record specifies an **Image** report item in the **BodyArea** record. See the **Image** example (section [3.2](#)) for additional details.

Measurements: A [Measurements](#) record specifies the measurements of an image report item.

measurementsStart: 0x10 specifies the start token of the **Measurements** record.

offset: 0x0000000000000052 specifies the position of the body token in the RPL stream.

count: 0x00000001 specifies that there is one [Measurement](#) record.

MeasurementsContent: A **MeasurementsContent** record specifies an array of **Measurement** records.

left: 0x00000000 specifies that the image is placed 0 millimeters to the left of the parent.

top: 0x00000000 specifies that the image is placed 0 millimeters below the parent.

width: 0x00001843 specifies that the image is 152 millimeters wide.

height: 0x00003243 specifies that the image is 178 millimeters tall.

zIndex: 0x00000000 specifies that the image is drawn first.

State: A **State** property specifies the state information for the **Measurement** record of the report item.

A: 0x0 specifies that the top border is not omitted.

B: 0x0 specifies that the bottom border is not omitted.

C: 0x0 specifies that the left border is not omitted.

D: 0x0 specifies that the right border is not omitted.

E: 0x0 specifies that the report item does not toggle visibility.

F: 0x0 specifies that the report item is not set to collapse.

Reserved: 0x0 specifies that the bit MUST be set to 0.

Reserved: 0x0 specifies that the bit MUST be set to 0.

reportElementEndOffset: 0x0000000000000070 specifies the position of the [ReportElementEnd](#) record for the **Image** report item structure in the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of the **BodyArea** record.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x000000000000007A specifies the position of the **measurementsStart** token for the **BodyArea** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

Measurements: A **Measurements** record specifies the measurements of the body areas.

measurementsStart: 0x10 specifies the start token of the **Measurements** record.

offset: 0x0000000000000051 specifies the position of the columns token in the RPL stream.

count: 0x00000001 specifies that there is one **Measurement** record.

MeasurementsContent: A **MeasurementsContent** record specifies an array of **Measurement** records.

left: 0x00000000 specifies that the body area is placed 0 millimeters to the left of the parent.

top: 0x0000CC41 specifies that the body area is placed 25.5 millimeters below the parent.

width: 0x00001843 specifies that the body area is 152 millimeters wide.

height: 0x00003243 specifies that the body area is 178 millimeters tall.

zIndex: 0x00000000 specifies that the body area is drawn first.

State: A **State** property specifies the state information for the **Measurement** record of the report item.

reportElementEndOffset: 0x00000000000000A4 specifies the position of the **ReportElementEnd** record for the **BodyElement** recording the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of **BodyAreaElement** record.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x00000000000000AE specifies the position of the **measurementsStart** token for **BodyAreaElement** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

PageFooterElement: A [PageFooterElement](#) record specifies the footer of a page.

pageFooter: 0x05 specifies the start token of the **PageFooterElement** record.

ElementProperties: An **ElementProperties** record specifies the properties for **PageFooterElement**.

elementPropertiesStart: 0x0F specifies the start token of the **ElementProperties** record.

SharedElementProperties: A **SharedElementProperties** record that specifies the shared properties of a **PageFooter** record.

shared: 0x00 specifies the start token of the **SharedElementProperties** record.

ItemProperties: An array of properties for the **PageFooterElement** record.

printOnFirstPageStart: 0x2C specifies the start token of the [PrintOnFirstPage](#) property.

printOnFirstPageValue: 0x00 specifies a false value, which indicates that the page footer will not print on the first page.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

delimiter: 0xFF specifies the end of the **ElementProperties** record.

ReportItems: A **ReportItem** record specifies items in the report.

Line: A [Line](#) record specifies a **Line** report item in the **PageFooter** record. See the **Line** example (section [3.1](#)) for additional details.

Measurements: A **Measurements** record specifies the measurements of the report items.

measurementsStart: 0x10 specifies the start token of the **Measurements** record.

offset: 0x00000000000000E2 specifies the position of the **pageFooter** token in the RPL stream.

count: 0x00000001 specifies that there is one **Measurement** record.

MeasurementsContent: A **MeasurementsContent** record specifies an array of **Measurement** records.

left: 0x00000000 specifies that the line is placed 0 millimeters to the left of the parent.

top: 0x00000000 specifies that the line is placed 0 millimeters below the parent.

width: 0x00001843 specifies that the line is 152 millimeters wide.

height: 0x0000C41 specifies that the line is 25.5 millimeters tall.

zIndex: 0x00000000 specifies that the line is drawn first.

State: A **State** property specifies the state information for the **Measurement** record of the report item.

reportElementEndOffset: 0x00000000000000F0 specifies the position of the **ReportElementEnd** record for the **Line** report item for the **PageFooterElement** record in the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of **PageFooterElement**.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x00000000000000FA specifies the position of the **measurementsStart** token for the **PageFooterElement** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

PageHeaderElement: A [PageHeaderElement](#) record specifies the header of a page.

pageHeader: 0x04 specifies the start token of the **PageHeaderElement** record.

ElementProperties: An **ElementProperties** record specifies the properties for **PageHeaderElement**.

elementPropertiesStart: 0x0F specifies the start token of the **ElementProperties** record.

SharedElementProperties: A **SharedElementProperties** record that specifies the shared properties of a **PageHeader** record.

shared: 0x00 specifies the start token of the **SharedElementProperties** record.

ItemProperties: An array of properties for the **PageHeaderElement** record.

printOnFirstPageStart: 0x2C specifies the start token of the **PrintOnFirstPage** property.

printOnFirstPageValue: 0x01 specifies a true value, which indicates that the page header will print on the first page.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

delimiter: 0xFF specifies the end of the **ElementProperties** record.

ReportItems: A **ReportItem** record that specifies the items in the report.

Line: A **Line** record specifies a **Line** report item in the **PageHeader** record. See the **Line** example (section [3.1](#)) for additional details.

Measurements: A **Measurements** record specifies the measurements of the report items.

measurementsStart: 0x10 specifies the start token of the **Measurements** record.

offset: 0x0000000000000012E specifies the position of the **pageHeader** token in the RPL stream.

count: 0x00000001 specifies that there is one **Measurement** record.

MeasurementsContent: A **MeasurementsContent** record specifies an array of **Measurement** records.

left: 0x00000000 specifies that the line is placed 0 millimeters to the left of the parent.

top: 0x00000000 specifies that the line is placed 0 millimeters below the parent.

width: 0x00001843 specifies that the line is 152 millimeters wide.

height: 0x0000CC41 specifies that the line is 25.5 millimeters tall.

zIndex: 0x00000000 specifies that the line is drawn first.

State: A **State** property specifies the state information for the **Measurement** record of the report item.

reportElementEndOffset: 0x000000000000013C specifies the position of the **ReportElementEnd** record for the **Line** report item for the **PageHeaderElement** record in the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of the **PageHeaderElement**.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x0000000000000146 specifies the position of the **measurementsStart** token for the **PageHeaderElement** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

Measurements: A **Measurements** record specifies the measurements of the body area element, page header footer, and page header element.

measurementsStart: 0x10 specifies the start token of the **Measurements** record.

offset: 0x0000000000000037 specifies the position of the **sectionStart** token in the RPL stream.

count: 0x00000003 specifies that there are three **Measurement** records.

MeasurementsContent: A **MeasurementsContent** record specifies an array of **Measurement** records.

left: 0x00000000 specifies that the body area is placed 0 millimeters to the left of the parent.

top: 0x0000CC41 specifies that the body area is placed 25.5 millimeters below the parent.

width: 0x00001843 specifies that the body area is 152 millimeters wide.

height: 0x00003243 specifies that the body area is 178 millimeters tall.

zIndex: 0x00000000 specifies that the body area is drawn first.

State: A **State** property specifies the state information for the **Measurement** record of the report item.

reportElementEndOffset: 0x00000000000000D8 specifies the position of the **ReportElementEnd** record for the **BodyAreaElement** record in the RPL stream.

left: 0x00000000 specifies that the page footer is placed 0 millimeters to the left of the parent.

top: 0x00804B43 specifies that the page footer is placed 203.5 millimeters below the parent.

width: 0x00001843 specifies that the page footer is 152 millimeters wide.

height: 0x0000CC41 specifies that the page footer is 25.5 millimeters tall.

zIndex: 0x00000001 specifies that the page footer is drawn second.

State: A **State** property specifies the state information for the **Measurement** record of the report item.

reportElementEndOffset: 0x0000000000000124 specifies the position of the **ReportElementEnd** record for the **PageFooterElement** record in the RPL stream.

left: 0x00000000 specifies that the page header is placed 0 millimeters to the left of the parent.

top: 0x00000000 specifies that the page header is placed 0 millimeters below the parent.

width: 0x00001843 specifies that the page header is 152 millimeters wide.

height: 0x0000CC41 specifies that the page header is 25.5 millimeters tall.

zIndex: 0x00000002 specifies that the page header is drawn third.

State: A **State** property specifies the state information for the **Measurement** record of the report item.

reportElementEndOffset: 0x0000000000000170 specifies the position of the **ReportElementEnd** record for the **PageHeaderElement** record in the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of the **SimpleSection** record.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x000000000000017A specifies the position of the **measurementsStart** token for **SimpleSection** in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

Measurements: A **Measurements** record specifies the measurements of the sections.

measurementsStart: 0x10 specifies the start token of the **Measurements** structure.

offset: 0x000000000000002A specifies the position of the **pageContentStart** token in the RPL stream.

count: 0x00000001 specifies that there is one Measurement record.

MeasurementsContent: A **MeasurementsContent** structure specifies an array of **Measurement** records.

left: 0x00000000 specifies that the section is placed 0 millimeters to the left of the parent.

top: 0x00000000 specifies that the section is placed 0 millimeters below the parent.

width: 0x00001843 specifies that the section is 152 millimeters wide.

height: 0x00006543 specifies that the section is 229 millimeters tall.

zIndex: 0x00000000 specifies that the section is drawn first.

State: A **State** property specifies the state information for the **Measurement** record of the report item.

reportElementEndOffset: 0x00000000000001DE specifies the position of the **ReportElementEnd** record for the **PageContent** record in the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of the **PageContent** record.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x00000000000001E8 specifies the position of the **measurementsStart** token for the **PageContent** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

OffsetsArrayElement: An [OffsetsArrayElement](#) record specifies the offset for each page.

offsetsArrayElementStart: 0x12 specifies the start token of the **OffsetsArrayElement** record.

offset: 0x0000000000000012 specifies the position of the **reportStart** token in the RPL stream.

count: 0x00000001 specifies that there is one **PageContent** record.

OffsetsPageContent: An **OffsetsPageContent** record specifies an array with offsets to each **reportElementEnd** field of a **PageContent** record in the RPL stream.

offset: 0x0000000000000212 specifies the position of the **ReportElementEnd** record for the **PageContent** record in the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of the Report.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** structure.

offset: 0x000000000000021C specifies the position of the **offsetsArrayElementStart** token in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

Version: A **Version** record specifies the RPL version.

majorVersion: 0x0A specifies that the major version is 10.

minorVersion: 0x04 specifies that the minor version is 4.

build: 0x00000000 specifies that the build is 0.

3.9 Report 2008

This example specifies the records that are required to produce a [Report](#) by using RPL 10.3.

| Offset | Size | Structure | Value |
|--------|------|---------------|-------|
| | 0201 | Report | |

| Offset | Size | Structure | Value |
|--------|------|--------------------------------------|------------|
| 0x1 | 000B | STRING- RPLStamp | RPLIF |
| | 0006 | Version | |
| 0xC | 0001 | BYTE- majorVersion | 0x0A |
| 0xD | 0001 | BYTE- minorVersion | 0x03 |
| 0xE | 0004 | INT- build | 0x00000000 |
| 0x12 | 0001 | BYTE- reportStart | 0x00 |
| | 0017 | ReportProperties | |
| 0x13 | 0001 | BYTE- reportPropertiesStart | 0x02 |
| | 0015 | ItemProperties | |
| 0x14 | 0001 | BYTE - reportNameStart | 0x0F |
| 0x15 | 000F | STRING - reportNameValue | Report1 |
| 0x24 | 0001 | BYTE - autoRefreshStart | 0x0E |
| 0x25 | 0004 | INT - autoRefreshValue | 0x0000001E |
| 0x29 | 0001 | BYTE - delimiter | 0xFF |
| | 01B3 | PageContent | |
| 0x2A | 0001 | BYTE- pageContentStart | 0x13 |
| | 00CB | BodyAreaElement | |
| 0x2B | 0001 | BYTE - columns | 0x14 |
| | 005C | BodyArea | |
| 0x2C | 0001 | BYTE - body | 0x06 |
| | 0016 | ElementProperties | |
| 0x2D | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0014 | SharedElementProperties | |
| 0x2E | 0001 | BYTE - shared | 0x00 |
| | 0012 | ItemProperties | |
| 0x2F | 0001 | BYTE - idStart | 0x00 |
| 0x30 | 0011 | STRING - idValue | BodyArea |
| 0x41 | 0001 | BYTE - delimiter | 0xFF |
| 0x42 | 0001 | BYTE - delimiter | 0xFF |
| | 0011 | ReportItems | |

| Offset | Size | Structure | Value |
|--------|-------|-------------------------------------|--------------------|
| 0x43 | 0011 | Image | |
| | 002A | Measurements | |
| 0x54 | 0001 | BYTE - measurementsStart | 0x10 |
| 0x55 | 0008 | INT- offset | 0x000000000000002C |
| 0x5D | 0004 | INT- count | 0x00000001 |
| | 001D | measurementsContent | |
| 0x61 | 0004 | FLOAT - left | 0x00000000 |
| 0x65 | 0004 | FLOAT - top | 0x00000000 |
| 0x69 | 0004 | FLOAT - width | 0x00001843 |
| 0x6D | 0004 | FLOAT - height | 0x00003243 |
| 0x71 | 0004 | INT - zIndex | 0x00000000 |
| 0x75 | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x76 | 0008 | INT - reportElementEndOffset | 0x000000000000004A |
| | 000A | ReportElementEnd | |
| 0x7E | 0001 | BYTE - elementEnd | 0xFE |
| 0x7F | 0008 | INT- offset | 0x0000000000000054 |
| 0x87 | 0001 | BYTE - delimiter | 0xFF |
| | 002A | Measurements | |
| 0x88 | 0001 | BYTE - measurementsStart | 0x10 |
| 0x89 | 0008 | INT- offset | 0x000000000000002B |
| 0x91 | 0004 | INT- count | 0x00000001 |
| | 001D | measurementsContent | |

| Offset | Size | Structure | Value |
|--------|-------|-------------------------------------|--------------------|
| 0x95 | 0004 | FLOAT - left | 0x00000000 |
| 0x99 | 0004 | FLOAT - top | 0x0000CC41 |
| 0x9D | 0004 | FLOAT - width | 0x00001843 |
| 0xA1 | 0004 | FLOAT - height | 0x00003243 |
| 0xA5 | 0004 | INT - zIndex | 0x00000000 |
| 0xA9 | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0xAA | 0008 | INT - reportElementEndOffset | 0x000000000000007E |
| | 000A | ReportElementEnd | |
| 0xB2 | 0001 | BYTE - elementEnd | 0xFE |
| 0xB3 | 0008 | INT- offset | 0x0000000000000088 |
| 0xBB | 0001 | BYTE - delimiter | 0xFF |
| | 00B3 | Page | |
| 0xBC | 0001 | BYTE - pageStart | 0x01 |
| | 0019 | PageProperties | |
| 0xBD | 0001 | BYTE - pagePropertiesStart | 0x03 |
| | 0017 | ItemProperties | |
| 0xBE | 0001 | BYTE - idStart | 0x00 |
| 0xBF | 0011 | STRING - idValue | PageID01 |
| 0xD0 | 0001 | BYTE - columnsStart | 0x17 |
| 0xD1 | 0004 | INT - columnsValue | 0x00000001 |
| 0xD5 | 0001 | BYTE - delimiter | 0xFF |
| | 004C | PageHeaderElement | |

| Offset | Size | Structure | Value |
|--------|-------|--------------------------------------|--------------------|
| 0xD6 | 0001 | BYTE - pageHeader | 0x04 |
| | 0006 | ElementProperties | |
| 0xD7 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0004 | SharedElementProperties | |
| 0xD8 | 0001 | BYTE - shared | 0x00 |
| | 0002 | ItemProperties | |
| 0xD9 | 0001 | BYTE - printOnFirstPageStart | 0x2C |
| 0xDA | 0001 | BYTE - printOnFirstPageValue | 0x01 |
| 0xDB | 0001 | BYTE - delimiter | 0xFF |
| 0xDC | 0001 | BYTE - delimiter | 0xFF |
| | 0011 | ReportItems | |
| 0xDD | 0011 | Line | |
| | 002A | Measurements | |
| 0xEE | 0001 | BYTE - measurementsStart | 0x10 |
| 0xEF | 0008 | INT- offset | 0x00000000000000D6 |
| 0xF7 | 0004 | INT- count | 0x00000001 |
| | 001D | measurementsContent | |
| 0xFB | 0004 | FLOAT - left | 0x00000000 |
| 0xFF | 0004 | FLOAT - top | 0x00000000 |
| 0x103 | 0004 | FLOAT - width | 0x00001843 |
| 0x107 | 0004 | FLOAT - height | 0x0000CC42 |
| 0x10B | 0004 | INT - zIndex | 0x00000000 |
| 0x10F | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |

| Offset | Size | Structure | Value |
|--------|-------|--------------------------------------|--------------------|
| | 1 bit | BIT - reserved | 0x0 |
| 0x110 | 0008 | INT - reportElementEndOffset | 0x00000000000000E4 |
| | 000A | ReportElementEnd | |
| 0x118 | 0001 | BYTE - elementEnd | 0xFE |
| 0x119 | 0008 | INT- offset | 0x00000000000000EE |
| 0x121 | 0001 | BYTE - delimiter | 0xFF |
| | 004C | PageFooterElement | |
| 0x122 | 0001 | BYTE - pageFooter | 0x05 |
| | 0006 | ElementProperties | |
| 0x123 | 0001 | BYTE - elementPropertiesStart | 0x0F |
| | 0004 | SharedElementProperties | |
| 0x124 | 0001 | BYTE - shared | 0x00 |
| | 0002 | ItemProperties | |
| 0x125 | 0001 | BYTE - printOnFirstPageStart | 0x2C |
| 0x126 | 0001 | BYTE - printOnFirstPageValue | 0x00 |
| 0x127 | 0001 | BYTE - delimiter | 0xFF |
| 0x128 | 0001 | BYTE - delimiter | 0xFF |
| | 0011 | ReportItems | |
| 0x129 | 0011 | Line | |
| | 002A | Measurements | |
| 0x13A | 0001 | BYTE - measurementsStart | 0x10 |
| 0x13B | 0008 | INT- offset | 0x0000000000000122 |
| 0x143 | 0004 | INT- count | 0x00000001 |
| | 001D | measurementsContent | |
| 0x147 | 0004 | FLOAT - left | 0x00000000 |
| 0x14B | 0004 | FLOAT - top | 0x00000000 |
| 0x14F | 0004 | FLOAT - width | 0X00001843 |
| 0x153 | 0004 | FLOAT - height | 0x0000CC41 |
| 0x157 | 0004 | INT - zIndex | 0x00000000 |
| 0x15B | 0001 | BYTE - state | |

| Offset | Size | Structure | Value |
|--------|-------|-------------------------------------|---------------------|
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x15C | 0008 | INT - reportElementEndOffset | 0x00000000000000130 |
| | 000A | ReportElementEnd | |
| 0x164 | 0001 | BYTE - elementEnd | 0xFE |
| 0x165 | 0008 | INT- offset | 0x0000000000000013A |
| 0x16D | 0001 | BYTE - delimiter | 0xFF |
| 0x16E | 0001 | BYTE - delimiter | 0xFF |
| | 0064 | Measurements | |
| 0x16F | 0001 | BYTE - measurementsStart | 0x10 |
| 0x170 | 0008 | INT- offset | 0x0000000000000002A |
| 0x178 | 0004 | INT- count | 0x00000003 |
| | 0057 | measurementsContent | |
| 0x17C | 0004 | FLOAT - left | 0x00000000 |
| 0x180 | 0004 | FLOAT - top | 0x0000CC41 |
| 0x184 | 0004 | FLOAT - width | 0x00001843 |
| 0x188 | 0004 | FLOAT - height | 0x00003243 |
| 0x18C | 0004 | INT - zIndex | 0x00000000 |
| 0x190 | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |

| Offset | Size | Structure | Value |
|--------|-------|-------------------------------------|--------------------|
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x191 | 0008 | INT - reportElementEndOffset | 0x00000000000000B2 |
| 0x199 | 0004 | FLOAT - left | 0x00000000 |
| 0x19D | 0004 | FLOAT - top | 0x00000000 |
| 0x1A1 | 0004 | FLOAT - width | 0x00001843 |
| 0x1A5 | 0004 | FLOAT - height | 0x0000CC41 |
| 0x1A9 | 0004 | INT - zIndex | 0x00000000 |
| 0x1AD | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x1AE | 0008 | INT - reportElementEndOffset | 0x0000000000000118 |
| 0x1B6 | 0004 | FLOAT - left | 0x00000000 |
| 0x1BA | 0004 | FLOAT - top | 0x00804B43 |
| 0x1BE | 0004 | FLOAT - width | 0x00001843 |
| 0x1C2 | 0004 | FLOAT - height | 0x0000CC41 |
| 0x1C6 | 0004 | INT - zIndex | 0x00000000 |
| 0x1CA | 0001 | BYTE - state | |
| | 1 bit | BIT - omitTop | 0x0 |
| | 1 bit | BIT - omitBottom | 0x0 |
| | 1 bit | BIT - omitLeft | 0x0 |
| | 1 bit | BIT - omitRight | 0x0 |
| | 1 bit | BIT - hasToggle | 0x0 |

| Offset | Size | Structure | Value |
|--------|-------|---------------------------------------|---------------------|
| | 1 bit | BIT - toggleCollapse | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| | 1 bit | BIT - reserved | 0x0 |
| 0x1CB | 0008 | INT - reportElementEndOffset | 0x00000000000000164 |
| | 000A | ReportElementEnd | |
| 0x1D3 | 0001 | BYTE - elementEnd | 0xFE |
| 0x1D4 | 0008 | INT- offset | 0x0000000000000016F |
| 0x1DC | 0001 | BYTE - delimiter | 0xFF |
| | 0015 | OffSetsArrayElement | |
| 0x1DD | 0001 | BYTE- offsetsArrayElementStart | 0x12 |
| 0x1DE | 0008 | INT- offset | 0x00000000000000012 |
| 0x1E6 | 0004 | INT- count | 0x00000001 |
| | 0008 | OffsetsPageContent | |
| 0x1EA | 0008 | INT- offset | 0x000000000000001D3 |
| | 000A | ReportElementEnd | |
| 0x1F2 | 0001 | BYTE - elementEnd | 0xFE |
| 0x1F3 | 0008 | INT- offset | 0x000000000000001DD |
| 0x1FB | 0001 | BYTE - delimiter | 0xFF |
| | 0006 | Version | |
| 0x1FC | 0001 | BYTE- majorVersion | 0x0A |
| 0x1FD | 0001 | BYTE- minorVersion | 0x03 |
| 0x1FE | 0004 | INT- build | 0x00000000 |

Report: A Report record specifies the contents of the report.

RPLStamp: The "RPLIF" value specifies the value for the stamp.

Version: A [Version](#) structure specifies the RPL version.

majorVersion: 0x0A specifies that the major version is 10.

minorVersion: 0x03 specifies that the minor version is 3.

build: 0x00000000 specifies that the build is 0.

reportStart: 0x00 specifies the start token of the **Report** record.

ReportProperties: A [ReportProperties](#) record specifies properties of the report.

reportPropertiesStart: 0x02 specifies the start token of the **ReportProperties** record.

ItemProperties: An array of item properties for the **Report** record.

reportNameStart: 0x0F specifies the start token of the [ReportName](#) property.

reportNameValue: *Report1* specifies the name of the report.

autoRefreshStart: 0x0E specifies the start token of the [AutoRefresh](#) property.

autoRefreshValue: 0x0000001E specifies that the automatic refresh interval is 30 seconds.

delimiter: 0xFF specifies the end of the **ReportProperties** record.

PageContent: A [PageContent](#) (RPL 10.3) record specifies the pages of a report.

pageContentStart: 0x13 specifies the start token of the **PageContent** structure.

BodyAreaElement: A [BodyAreaElement](#) (RPL 10.3) record specifies an area of the body of **PageContent**.

Columns: 0x14 specifies the start token of the **BodyAreaElement** record.

BodyArea: A [BodyElement](#) record specifies the content of the main area of **BodyAreaElement**.

body: 0x06 specifies the start token of the **BodyArea** record.

ElementProperties: An [ElementProperties](#) record that specifies the properties for the **BodyArea** record.

elementPropertiesStart: 0x0F specifies the start token of the **ElementProperties** record.

SharedElementProperties: A [SharedElementProperties](#) record that specifies the [InlineSharedElementProperties](#) record of the **BodyArea** record.

shared: 0x00 specifies the start token of the **SharedElementProperties** record.

ItemProperties: An array of properties for the **BodyAreaElement** record.

idStart: 0x00 specifies the start token of the [ID](#) property.

idValue: *BodyArea* specifies **ID** for the **BodyArea** record.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

delimiter: 0xFF specifies the end of the **ElementProperties** record.

ReportItems: A [ReportItem](#) record specifies items in the report.

Image: An [Image](#) record specifies an **Image** report item in the **BodyArea** record. See the **Image** example (section [3.2](#)) for additional details.

Measurements: A [Measurements](#) record specifies the measurements of an **Image** report item.

measurementsStart: 0x10 specifies the start token of the **Measurements** record.

offset: 0x000000000000002C specifies the position of the body token in the RPL stream.

count: 0x00000001 specifies that there is one measurement record.

MeasurementsContent: A **MeasurementsContent** record specifies an array of measurements.

left: 0x00000000 specifies that the image is placed 0 millimeters to the left of the parent.

top: 0x00000000 specifies that the image is placed 0 millimeters below the parent.

width: 0x00001843 specifies that the image is 152 millimeters wide.

height: 0x00003243 specifies that the image is 178 millimeters tall.

zIndex: 0x00000000 specifies that the image is drawn first.

State: A [State](#) property specifies the state information for the **Measurement** record of the report item.

A: 0x0 specifies that the top border is not omitted.

B: 0x0 specifies that the bottom border is not omitted.

C: 0x0 specifies that the left border is not omitted.

D: 0x0 specifies that the right border is not omitted.

E: 0x0 specifies that the report item does not toggle visibility.

F: 0x0 specifies that the report item is not set to collapse.

Reserved: 0x0 specifies that the bit MUST be set to 0.

Reserved: 0x0 specifies that the bit MUST be set to 0.

reportElementEndOffset: 0x000000000000004A specifies the position of the **ReportElementEnd** record for the **Image** report item structure in the RPL stream.

ReportElementEnd: A [ReportElementEnd](#) record specifies the end of the **BodyArea** record.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x0000000000000054 specifies the position of the **measurementsStart** token for the **BodyArea** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

Measurements: A **Measurements** record specifies the measurements of the body areas.

measurementsStart: 0x10 specifies the start token of the **Measurements** record.

offset: 0x000000000000002B specifies the position of the columns token in the RPL stream.

count: 0x00000001 specifies that there is one measurement record.

MeasurementsContent: A **MeasurementsContent** structure specifies an array of measurements.

left: 0x00000000 specifies that the body area is placed 0 millimeters to the left of the parent.

top: 0x0000CC41 specifies that the body area is placed 25.5 millimeters below the parent.

width: 0x00001843 specifies that the body area is 152 millimeters wide.

height: 0x00003243 specifies that the body area is 178 millimeters tall.

zIndex: 0x00000000 specifies that the body area is drawn first.

State: A **State** property specifies the state information for the [Measurement](#) record of the report item.

reportElementEndOffset: 0x000000000000007E specifies the position of the **ReportElementEnd** record for the **BodyElement** structure in the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of the **BodyAreaElement** record.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x0000000000000088 specifies the position of the **measurementsStart** token for the **BodyAreaElement** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

Page: A [Page](#) record specifies a **Page** record in [PageContent](#).

pageStart: 0x01 specifies the start token of the **Page** record.

PageProperties: A [PageProperties](#) record specifies the properties of a page.

pagePropertiesStart: 0x03 specifies the start token of the **PageProperties** record.

ItemProperties: An array of properties for the **Page** record.

idStart: 0x00 specifies the start token of the **ID** property.

idValue: *PageID01* specifies the identifier for the page.

columnsStart: 0x17 specifies the start token of the **ColumnCount** property.

columnsValue: 0x00000001 specifies that there is one column in this page.

delimiter: 0xFF specifies the end of the **PageProperties** record.

PageHeaderElement: A [PageHeaderElement](#) record specifies the header of a page.

pageHeader: 0x04 specifies the start token of the **PageHeaderElement** record.

ElementProperties: An **ElementProperties** record specifies the properties for the **PageHeaderElement** record.

elementPropertiesStart: 0x0F specifies the start token of the **ElementProperties** record.

SharedElementProperties: A **SharedElementProperties** record that specifies the shared properties of a **pageHeader** record.

shared: 0x00 specifies the start token of the **SharedElementProperties** record.

ItemProperties: An array of properties for the **PageHeaderElement** record.

printOnFirstPageStart: 0x2C specifies the start token of the **PrintOnFirstPage** property.

printOnFirstPageValue: 0x01 specifies a true value, which indicates that the page header will print on the first page.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

delimiter: 0xFF specifies the end of the **ElementProperties** record.

ReportItems: A ReportItem record that specifies the items in the report.

Line: A [Line](#) record specifies a **Line** report item in the **PageHeader** record. See the **Line** example (section [3.1](#)) for additional details.

Measurements: A **Measurements** record specifies the measurements of the report items.

measurementsStart: 0x10 specifies the start token of the **Measurements** record.

offset: 0x00000000000000D6 specifies the position of the **pageHeader** token in the RPL stream.

count: 0x00000001 specifies that there is one **Measurement** record.

MeasurementsContent: A **MeasurementsContent** record specifies an array of **Measurement** records.

left: 0x00000000 specifies that the line is placed 0 millimeters to the left of the parent.

top: 0x00000000 specifies that the line is placed 0 millimeters below the parent.

width: 0x00001843 specifies that the line is 152 millimeters wide.

height: 0x0000CC41 specifies that the line is 25.5 millimeters tall.

zIndex: 0x00000000 specifies that the line is drawn first.

State: A **State** property specifies the state information for the Measurement record of the report item.

reportElementEndOffset: 0x00000000000000E4 specifies the position of the **ReportElementEnd** record of the **Line** report item in the **PageHeaderElement** record in the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of the **PageHeaderElement** record.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x00000000000000EE specifies the position of the **measurementsStart** token for the **PageHeaderElement** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

PageFooterElement: A [PageFooterElement](#) record specifies the footer of a page.

pageFooter: 0x05 specifies the start token of the **PageFooterElement** record.

ElementProperties: An ElementProperties record specifies the properties for the **PageFooterElement** record.

elementPropertiesStart: 0x0F specifies the start token of the **ElementProperties** record.

SharedElementProperties: A **SharedElementProperties** record that specifies the shared properties of a **PageFooter** record.

shared: 0x00 specifies the start token of the **SharedElementProperties** record.

ItemProperties: An array of properties for the **PageFooterElement** record.

printOnFirstPageStart: 0x2C specifies the start token of the **PrintOnFirstPage** property.

printOnFirstPageValue: 0x00 specifies a false value, which indicates that the page footer will not print on the first page.

delimiter: 0xFF specifies the end of the **SharedElementProperties** record.

delimiter: 0xFF specifies the end of the **ElementProperties** record.

ReportItems: A **ReportItem** record specifies items in the report.

Line: A **Line** record specifies a **Line** report item in the **PageFooter** record. See the **Line** example for additional details.

Measurements: A **Measurements** record specifies the measurements of the report items.

measurementsStart: 0x10 specifies the start token of the **Measurements** structure.

offset: 0x00000000000000122 specifies the position of the **pageFooter** token in the RPL stream.

count: 0x00000001 specifies that there is one **Measurement** record.

MeasurementsContent: A **MeasurementsContent** record specifies an array of **Measurement** records.

left: 0x00000000 specifies that the line is placed 0 millimeters to the left of the parent.

top: 0x00000000 specifies that the line is placed 0 millimeters below the parent.

width: 0x00001843 specifies that the line is 152 millimeters wide.

height: 0x0000CC41 specifies that the line is 25.5 millimeters tall.

zIndex: 0x00000000 specifies that the line is drawn first.

State: A **State** property specifies the state information for the **Measurement** record of the report item.

reportElementEndOffset: 0x00000000000000130 specifies the position of the **ReportElementEnd** record of the **Line** report item in the **PageFooterElement** record in the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of the **PageFooterElement** record.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x0000000000000013A specifies the position of the **measurementsStart** token for the **PageFooterElement** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

Measurements: A **Measurements** record specifies the measurements of the **PageContent (RPL 10.3)** record.

measurementsStart: 0x10 specifies the start token of the **Measurements** record.

offset: 0x000000000000002A specifies the position of the **pageContentStart** token in the RPL stream.

count: 0x00000003 specifies that there are three **Measurement** records.

MeasurementsContent: A **MeasurementsContent** record specifies an array of **Measurement** records.

left: 0x00000000 specifies that the body area is placed 0 millimeters to the left of the parent.

top: 0x0000CC41 specifies that the body area is placed 25.5 millimeters below the parent.

width: 0x00001843 specifies that the body area is 152 millimeters wide.

height: 0x00003243 specifies that the body area is 178 millimeters tall.

zIndex: 0x00000000 specifies that the body area is drawn first.

State: A **State** property specifies the state information for the **Measurement** record of the report item.

reportElementEndOffset: 0x000000000000007E specifies the position of the **ReportElementEnd** record for the **Body** record in the RPL stream.

left: 0x00000000 specifies that the page header is placed 0 millimeters to the left of the parent.

top: 0x00000000 specifies that the page header is placed 0 millimeters below the parent.

width: 0x00001843 specifies that the page header is 152 millimeters wide.

height: 0x0000CC41 specifies that the page header is 25.5 millimeters all.

zIndex: 0x00000001 specifies that the page header is drawn second.

State: A **State** property specifies the state information for the **Measurement** record of the page header.

reportElementEndOffset: 0x0000000000000118 specifies the position of the **ReportElementEnd** record for the **PageHeader** record in the RPL stream.

left: 0x00000000 specifies that the page footer is placed 0 millimeters to the left of the parent.

top: 0x00804B43 specifies that the page footer is placed 203.5 millimeters below the parent.

width: 0x00001843 specifies that the page footer is 152 millimeters wide.

height: 0x0000CC41 specifies that the page footer is 25.5 millimeters tall.

zIndex: 0x00000002 specifies that the page footer is drawn third.

State: A **State** property specifies the state information for the **Measurement** record of the page footer.

reportElementEndOffset: 0x0000000000000164 specifies the position of the **ReportElementEnd** record for the **PageFooter** record in the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of **PageContent**.

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x000000000000016F specifies the position of the **measurementsStart** token for the **PageContent** record in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

OffsetsArrayElement: An [OffsetsArrayElement](#) record specifies the **offset** for each page.

offsetsArrayElementStart: 0x12 specifies the start token of the **OffsetsArrayElement** structure.

offset: 0x0000000000000012 specifies the position of the **reportStart** token in the RPL stream.

count: 0x00000001 specifies that there is one **PageContent** record.

OffsetsPageContent: An **OffsetsPageContent** record specifies an array with offsets to each **reportElementEnd** field of a **PageContent** record in the RPL stream.

offset: 0x00000000000001D3 specifies the position of the **ReportElementEnd** record for the **PageContent** record in the RPL stream.

ReportElementEnd: A **ReportElementEnd** record specifies the end of the [Report](#).

elementEnd: 0xFE specifies the start token of the **ReportElementEnd** record.

offset: 0x00000000000001DD specifies the position of the **offsetsArrayElementStart** token in the RPL stream.

delimiter: 0xFF specifies the end of the **ReportElementEnd** record.

Version: A Version structure specifies the RPL version.

majorVersion: 0x0A specifies that the major version is 10.

minorVersion: 0x03 specifies that the minor version is 3.

build: 0x00000000 specifies that the build is 0.

4 Security Considerations

None.

Preliminary

5 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products and technologies:

- Microsoft® SQL Server® 2008 R2 CTP2
- Microsoft® Visual Studio® 2010

Exceptions, if any, are noted below. Unless otherwise specified, any statement of optional behavior in this specification prescribed using the terms SHOULD or SHOULD NOT implies that the aforementioned Microsoft products' behavior is in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that these Microsoft products do not follow the prescription.

<1> [Section 2.2.1](#): **RPLStamp** value is currently defined as "RPLIF". This value is not validated by Microsoft ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<2> [Section 2.2.2](#): Version 0x03 corresponds to SQL Server 2008, and versions 0x04, 0x05, and 0x06 correspond to SQL Server 2009.

<3> [Section 2.2.13.1](#): While reading the **BodyAreaElement** structure, Microsoft ReportViewer controls do not enforce the validation that the number of elements MUST NOT be greater than the value of the **ColumnCount** property specified in the **SectionProperties** record. ReportViewer controls are the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010.

<4> [Section 2.2.13.2](#): While reading the **BodyAreaElement** structure, Microsoft ReportViewer controls do not enforce the validation that the number of elements MUST NOT be greater than the value of the **ColumnCount** property specified in the **Page** record. ReportViewer controls are the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010.

<5> [Section 2.2.22](#): The parent record is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<6> [Section 2.2.22](#): The parent record is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<7> [Section 2.2.31](#): The restriction that **Chart**, **Map**, **GaugePanel**, **SubReport**, and **Tablix** records MUST NOT be part of **PageHeaderElement** and **PageFooterElement** is not imposed by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<8> [Section 2.2.40](#): The number of coordinates based on shape type is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<9> [Section 2.2.56](#): The value of the **Width** property is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<10> [Section 2.2.58](#): The **Height** value is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<11> [Section 2.2.58](#): **SharedLayoutRow** and **UseSharedLayoutRow** being set at the same time is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<12> [Section 2.2.58](#): **SharedLayoutRow** and **UseSharedLayoutRow** being set at the same time is not validated by the Microsoft RPL Reader.

<13> [Section 2.3.9](#): ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010) do not validate that the **ToggleItem** property MUST NOT be used in **PageHeader** and **PageFooter**.

<14> [Section 2.3.28](#): When a property is part of both shared and nonshared **ElementProperties** structures, the nonshared property takes precedence over the shared property.

<15> [Section 2.3.33](#): When a property is part of both shared and nonshared **ElementProperties** structures, the nonshared property takes precedence over the shared property.

<16> [Section 2.3.34](#): When a property is part of both shared and nonshared **ElementProperties** structures, the nonshared property takes precedence over the shared property.

<17> [Section 2.3.36](#): ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010) do not validate whether the **IsSimple** property specifies a **RichTextBox** with one or multiple **TextRun** records.

<18> [Section 2.3.38](#): The Microsoft implementation uses (C/M/G)_UniqueName_PageNumber, where C is used for **Chart**, M is used for **Map**, and G is used for **Gauge**.

<19> [Section 2.3.45](#): When a property is part of both shared and nonshared **ElementProperties** structures, the nonshared property takes precedence over the shared property.

<20> [Section 2.3.45](#): ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010) do not validate the **MIMEType**.

<21> [Section 2.3.46](#): When a property is part of both shared and nonshared **ElementProperties** structures, the nonshared property takes precedence over the shared property.

<22> [Section 2.3.47](#): When a property is part of both shared and nonshared **ElementProperties** structures, the nonshared property takes precedence over the shared property.

<23> [Section 2.3.47](#): The value of **Width**, **Height**, **HorizontalResolution**, **VerticalResolution**, and **RawFormat** being present, all together, is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<24> [Section 2.3.48](#): When a property is part of both shared and nonshared **ElementProperties** structures, the nonshared property takes precedence over the shared property.

<25> [Section 2.3.48](#): The value of **Width**, **Height**, **HorizontalResolution**, **VerticalResolution**, and **RawFormat** being present, all together, is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<26> [Section 2.3.49](#): When a property is part of both shared and nonshared **ElementProperties** structures, the nonshared property takes precedence over the shared property.

<27> [Section 2.3.49](#): The value of **Width, Height, HorizontalResolution, VerticalResolution,** and **RawFormat,** all together, being present is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<28> [Section 2.3.50](#): When a property is part of both shared and nonshared **ElementProperties** structures, the nonshared property takes precedence over the shared property.

<29> [Section 2.3.50](#): The value of **Width, Height, HorizontalResolution, VerticalResolution,** and **RawFormat,** all together, being present is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<30> [Section 2.3.51](#): When a property is part of both shared and nonshared **ElementProperties** structures, the nonshared property takes precedence over the shared property.

<31> [Section 2.3.51](#): The value of **Width, Height, HorizontalResolution, VerticalResolution,** and **RawFormat,** all together, being present is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<32> [Section 2.3.61](#): The **BorderStyle** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<33> [Section 2.3.62](#): The **BorderStyleLeft** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<34> [Section 2.3.63](#): The value of the **BorderStyleRight** property is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<35> [Section 2.3.64](#): The **BorderStyleTop** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<36> [Section 2.3.65](#): The **BorderStyleBottom** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<37> [Section 2.3.75](#): The **FontStyle** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<38> [Section 2.3.78](#): The **FontWeight** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<39> [Section 2.3.80](#): The **TextDecoration** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<40> [Section 2.3.81](#): The **TextAlign** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<41> [Section 2.3.82](#): The **VerticalAlign** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<42> [Section 2.3.83](#): The **Color** value is not validated against the rules specified in the description of **Color** property.

<43> [Section 2.3.85](#): The **Direction** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<44> [Section 2.3.86](#): The **WritingMode** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<45> [Section 2.3.87](#): The **UnicodeBiDi** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<46> [Section 2.3.89](#): The **BackgroundColor** value is not validated against the rules specified in the description of **Color** property.

<47> [Section 2.3.90](#): The **BackgroundRepeat** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<48> [Section 2.3.92](#): The **NumeralVariant** value is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<49> [Section 2.3.93](#): The **Calendar** value is not validated against the enumeration by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<50> [Section 2.3.94](#): The **ColumnHeaderRows** value is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<51> [Section 2.3.95](#): The **RowHeaderColumns** value is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<52> [Section 2.3.96](#): The **ColsBeforeRowHeader** value is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<53> [Section 2.3.105](#): The **ColumnIndex** value is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<54> [Section 2.3.106](#): The **RowIndex** value is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

<55> [Section 2.3.111](#): The **ParagraphNumber** value is not validated by ReportViewer controls (the ReportViewer Windows Forms control and the ReportViewer ASP.NET Web server control that ship with Visual Studio 2010).

Preliminary

6 Change Tracking

This section identifies changes made to [MS-RPL] protocol documentation between August 2009 and November 2009 releases. Changes are classed as major, minor, or editorial.

Major changes affect protocol interoperability or implementation. Examples of major changes are:

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

Minor changes do not affect protocol interoperability or implementation. Examples are updates to fix technical accuracy or ambiguity at the sentence, paragraph, or table level.

Editorial changes apply to grammatical, formatting, and style issues.

No changes means that the document is identical to its last release.

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

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

- Content removed for template compliance.
- Obsolete document removed.

Editorial changes always have the revision type "Editorially updated."

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

Protocol syntax refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.

Protocol revision refers to changes made to a protocol that affect the bits that are sent over the wire.

Changes are listed in the following table. If you need further information, please contact protocol@microsoft.com.

| Section | Tracking number (if applicable) and description | Major change (Y or N) | Revision Type |
|---|---|-----------------------|---|
| 1.6 Versioning and Localization | 340010 Added information for new version 10.6. | N | New content added due to protocol revision. |
| 1.6 Versioning and Localization | 340010 Added information for new version 10.6. | N | New content added due to protocol revision. |
| 2.2.2 Version | 340010 Added new field value. | N | Content updated due to protocol revision. |
| 2.2.5.1 PageContent (RPL 10.6) | 340010 Added information for new version 10.6. | N | New content added due to protocol revision. |

7 Index

A

[Augmented Backus-Naur Form \(ABNF\)](#) 15

B

[BackgroundRepeatTypes enumeration](#) 166

[big-endian](#) 13

binary records

[about](#) 15

[RPL file structure](#) 15

[RPL stream](#) 15

Boolean data type ([section 2.1.1](#) 15, [section 2.1.3](#) 15)

[BorderStyles enumeration](#) 165

[byte boundary](#) 15

[byte ordering](#) 13

C

[Calendars enumeration](#) 165

[Change tracking](#) 231

D

data types

[about](#) 15

[values](#) 15

[Datetime data type](#) 15

[Directions enumeration](#) 164

E

[Enumerations](#) 161

F

fields

[unaligned](#) 15

[Float data type](#) 15

[FontStyles enumeration](#) 162

[FontWeights enumeration](#) 163

I

[ImageRawFormat enumeration](#) 162

Int32 data type ([section 2.1.1](#) 15, [section 2.1.3](#) 15)

[Int64 data type](#) 15

L

[left properties](#) 15

[ListStyles enumeration](#) 166

[little-endian](#) 13

[Long data type](#) 15

M

[MarkupStyles enumeration](#) 166

N

[numeric values](#) 15

O

[offsets](#) 15

P

[Properties](#) 84

R

[record type](#) 15

[rendering runtime sizes](#) 15

[report hierarchy](#) 15

[report layout](#) 15

[Report structure](#) 16

[report structures](#) 16

[ReportViewer controls](#) 14

RPL

[about](#) 12

version ([section 1.6](#) 14, [section 2.2.2](#) 18)

RPL stream

[about](#) 15

[file structure](#) 15

[protocols](#) 14

[reading strings](#) 16

[root record](#) 16

[Unicode characters](#) 13

[well formatted grammar](#) 14

writing strings to ([section 2.1.3.1](#) 16, [section 2.1.3.2](#) 16)

[RPLSize structure](#) 170

S

[ShapeType enumeration](#) 162

[Sizings enumeration](#) 161

[SortOptions enumeration](#) 161

[StateValues structure](#) 167

[String data type](#) 15

structure example

[Chart](#) 174

[GaugePanel](#) 176

[Image](#) 173

[Line](#) 172

[Report](#) 193

[Report 2008](#) 209

[RichTextBox](#) 180

[SubReport](#) 178

[Tablix](#) 186

structures

[enumerations](#) 161

[examples](#) 172

[properties](#) 84
[report structures](#) 16
[unaligned fields](#) 15

T

[TablixMemberDefStateValues structure](#) 169
[TablixMemberStateValues structure](#) 168
[TextAlignments enumeration](#) 163
[TextDecorations enumeration](#) 163
[top properties](#) 15
[Tracking changes](#) 231
[TypeCode enumeration](#) 167

U

[unaligned fields](#) 15
[Unicode characters](#) 13
[UnicodeBiDiTypes enumeration](#) 164
UTF-16 encoding ([section 1.4](#) 13, [section 2.1.3.1](#) 16)

V

version
RPL ([section 1.6](#) 14, [section 2.2.2](#) 18)
[Version structure](#) 18
[VerticalAlignments enumeration](#) 164

W

[WritingModes enumeration](#) 164