
Elixir Report Logical Model

Elixir Technology Pte Ltd

Copyright © 2006-2008 Elixir Technology Pte Ltd

Table of Contents

Introduction	1
Shorthand Notation	2
Attributes	2
Collections	2
Elements	3
Annotation	3
Body	3
BodyPart	3
ByteArray	4
Column	4
Columns	4
Detail	4
Empty	4
FontMetrics	4
Fragment	5
Glint	5
Grid	5
GroupFooter	5
GroupHeader	5
Image	6
Line	6
LogicalElement	7
LogicalElementHolder	7
LogicalModelElement	7
LogicalReport	7
Page	8
PageBreak	8
PageFooter	8
PageHeader	8
PageSequence	8
Rectangle	9
Reference	9
ReportFooter	9
ReportHeader	10
RTF	10
SizableElement	10
SVG	10
Text	10
TOCList	11

Introduction

This document describes the internal object model of Elixir Report. The object model consists of a number of classes that are all in the `com.elixirtech.report2.logical.model` package. The information is generated directly from the tool. While we endeavour to ensure this API will remain unchanged, it is

possible that changes will be needed in future releases to correct errors or to add significant functionality.

All location and dimension values are in twips, where we assume 1 twip = 1/20 of a pixel and there are 72 pixels per inch (giving 1440 twips per inch). Font sizes are also measured in twips.

Shorthand Notation

Attributes

Where an object is shown to contain another object, for example every Style has a String called Name, this implies there are methods called `getName()` and `setName(String name)`. In JavaScript it is also allowable to access the property directly as `style.Name`, which is internally mapped into calls to the `get` or `set` method, depending on the context. The only exception to this pattern is Boolean values, where an attribute of type Boolean called Enabled is accessed by `isEnabled()` and `setEnabled(Boolean b)`. This is consistent with Java bean naming conventions.

Collections

Many objects hold collections of other objects. In this object model these are Lists and Maps. Each List and Map is exposed through an API described here.

List collections are documented as in this example:

```
List: Item : List<CubeLevel>
```

This shorthand defines the following functions:

- `int getItemCount()`
- `void addItem(CubeLevel item)`
- `void addItem(int idx, CubeLevel item)`
- `void addAllItems(Collection collection)`
- `int indexOfItem(CubeLevel item)`
- `void removeItem(CubeLevel item)`
- `void removeAllItems(Collection collection)`
- `void removeAllItems()`
- `CubeLevel getItem(int idx)`
- `java.util.Iterator getItemIterator()`

Each shorthand can be trivially expanded into these ten functions, varying the name of the contents (Item in this case) and the type of the contents (CubeLevel in this case).

Map collections are documented as in this example:

```
Map: RenderDetails : Map<String,RenderDetails> key is value.getMimeType()
```

This shorthand defines the following functions:

- `void addRenderDetails(RenderDetails renderDetails)`

- void removeRenderDetails(RenderDetails renderDetails)
- RenderDetails getRenderDetails(String mimeType)
- java.util.Iterator getRenderDetailsIterator()
- int getRenderDetailsCount()

Each shorthand can be trivially expanded into these five functions, varying the name of the key type and value type (String and RenderDetails respectively in this case). The add method requires more explanation, the key is not passed in to the function, as with a typical Map.put(key,value) call, instead the key is derived by invoking a method on the value. In this example, the key is derived by calling the function getMimeType() on the RenderDetails, which returns a String to be used as the key. This is done inside the API, so you just have to add a RenderDetails and the key will be determined automatically.

Elements

Annotation

Annotation extends LogicalElementHolder (view LogicalElementHolder)

Implicit Style Name: annotation

- Visible : boolean
- Page : int
- Author : String
- Timestamp : String

Body

Body extends LogicalModelElement (view LogicalModelElement)

Implicit Style Name: body

List: BodyPart : List<BodyPart>

BodyPart

BodyPart extends LogicalElementHolder (view LogicalElementHolder)

Implicit Style Name: body-part

- SpaceAfter : int
- ForceNewPage : String
- Caption : String
- VerticalAlign : String
- KeepWithNext : boolean

ByteArray

ByteArray extends LogicalModelElement (view LogicalModelElement)

Implicit Style Name: bytes

Column

Column extends LogicalModelElement (view LogicalModelElement)

Implicit Style Name: column

- Width : int
- SpaceAfter : int

List: BodyPart : List<BodyPart>

Columns

Columns extends BodyPart (view BodyPart)

Implicit Style Name: columns

List: Column : List<Column>

Detail

Detail extends BodyPart (view BodyPart)

Implicit Style Name: detail

- TableOfContents : boolean
- TOCTitle : String
- TOCLevel : int

Empty

Empty extends LogicalElement (view LogicalElement)

Implicit Style Name: empty

FontMetrics

Implicit Style Name: fontmetrics

- Height : int
- Leading : int
- Ascent : int
- Descent : int

Fragment

Fragment extends LogicalElementHolder (view LogicalElementHolder)

Implicit Style Name: fragment

Glint

Glint extends LogicalElement (view LogicalElement)

Implicit Style Name: glint

- ImageResolution : int
- ByteArray : ByteArray (view ByteArray)

Grid

Grid extends Text (view Text)

Implicit Style Name: grid

- WritingMode : String
- GlyphOrientation : String
- LineGridMode : String
- LineGridProgression : String
- BoxBorderColor : String
- BoxBorderSide : boolean
- BoxBorderTop : boolean
- BoxBorderBottom : boolean
- BoxBorderHeight : int
- CharacterOffsetX : int
- CharacterOffsetY : int

GroupFooter

GroupFooter extends BodyPart (view BodyPart)

Implicit Style Name: group-footer

- TableOfContents : boolean

GroupHeader

GroupHeader extends BodyPart (view BodyPart)

Implicit Style Name: group-header

- TableOfContents : boolean

- TOCTitle : String
- RepeatSection : boolean
- TOCLevel : int

Image

Image extends LogicalElement (view LogicalElement)

Implicit Style Name: image

- Picture : String
- PictureType : String
- SizeMode : String
- HorizontalAlign : String
- VerticalAlign : String
- URL : String
- URLLDescription : String
- URLTarget : String

Line

Line extends LogicalElement (view LogicalElement)

Implicit Style Name: line

- LineColor : String
- X1 : int
- Y1 : int
- X2 : int
- Y2 : int
- Divisor : int
- LineWidth : int
- LineStyle : String
- ArrowStyle1 : String
- ArrowWidth1 : int
- ArrowColor1 : String
- ArrowStyle2 : String
- ArrowWidth2 : int
- ArrowColor2 : String

LogicalElement

LogicalElement extends SizableElement (view SizableElement)

Implicit Style Name: element

- Left : int
- Top : int
- Width : int

LogicalElementHolder

LogicalElementHolder extends SizableElement (view SizableElement)

Implicit Style Name: element-holder

- BackgroundColor : String
- Width : int
- URL : String
- URDescription : String
- URLTarget : String

List: LogicalElement : List<LogicalElement>

LogicalModelElement

Implicit Style Name: log

- Id : String

LogicalReport

Implicit Style Name: report

- Phase : String
- Caption : String
- ModelName : String
- Name : String
- Locale : Locale
- RecordSource : String
- PageHeaderFlags : String
- PageFooterFlags : String
- Width : int
- RecordCount : int

- XmlBase : String

Map: PageSetup : Map<String,com.elixirtech.report2.raw.model.PageSetup>key is value.getName()

Map: Parameter : Map<String,com.elixirtech.report2.raw.model.Parameter>key is value.getKey()

List: PageSequence : List<PageSequence>

Page

Page extends LogicalModelElement (view LogicalModelElement)

Implicit Style Name: page

- Name : String
- Watermark : Page (view Page)
- PageHeader : PageHeader (view PageHeader)
- Body : Body (view Body)
- PageFooter : PageFooter (view PageFooter)

List: Annotation : List<Annotation>

PageBreak

PageBreak extends LogicalElement (view LogicalElement)

Implicit Style Name: page-break

- ResetPageCount : boolean

PageFooter

PageFooter extends LogicalElementHolder (view LogicalElementHolder)

Implicit Style Name: page-footer

PageHeader

PageHeader extends LogicalElementHolder (view LogicalElementHolder)

Implicit Style Name: page-header

PageSequence

PageSequence extends LogicalModelElement (view LogicalModelElement)

Implicit Style Name: page-sequence

- MasterName : String
- ResetPageCount : boolean
- KeepBlankPages : boolean

List: Page : List<Page>

Rectangle

Rectangle extends LogicalElement (view LogicalElement)

Implicit Style Name: rect

- BackgroundColor : String
- BorderStyle : String
- BorderWidth : int
- BorderColor : String
- BorderLeft : boolean
- BorderTop : boolean
- BorderRight : boolean
- BorderBottom : boolean
- BorderRadius : int
- PaddingLeft : int
- PaddingRight : int
- PaddingTop : int
- PaddingBottom : int
- CurvedClip : boolean
- PaddedClip : boolean
- TableOfContents : boolean
- TOCTitle : String
- TOCLevel : int

List: LogicalElement : List<LogicalElement>

Reference

Reference extends LogicalElement (view LogicalElement)

Implicit Style Name: reference

- Title : String
- RefId : String

ReportFooter

ReportFooter extends BodyPart (view BodyPart)

Implicit Style Name: report-footer

- TableOfContents : boolean

ReportHeader

ReportHeader extends BodyPart (view BodyPart)

Implicit Style Name: report-header

- TableOfContents : boolean
- TOCTitle : String
- TOCLevel : int

RTF

RTF extends LogicalElement (view LogicalElement)

Implicit Style Name: rtf

- ShowAsImage : boolean

SizableElement

SizableElement extends LogicalModelElement (view LogicalModelElement)

Implicit Style Name: element

- Height : int
- KeepTogether : boolean
- Split : String

SVG

SVG extends LogicalElement (view LogicalElement)

Implicit Style Name: svg

- URL : String
- BackgroundStyle : String
- BackgroundColor : String
- ImageResolution : int

Text

Text extends LogicalElement (view LogicalElement)

Implicit Style Name: text

- FontName : String
- FontSize : int
- FontBold : boolean

- FontItalic : boolean
- FontColor : String
- FontUnderline : boolean
- FontStrikethrough : boolean
- TextAlign : String
- URL : String
- URLLDescription : String
- URLTarget : String
- Orientation : int

TOCList

TOCList extends BodyPart (view BodyPart)

Implicit Style Name: toc-list