**Graph Component**

**Automated Test Procedures**

**Authors:**

Sam Green

Nick Hudson

Stanton Sievers

Jarrod Stormo

Table of Contents

[Axis 9](#_Toc218674953)

[Bar 9](#_Toc218674954)

[Configure 9](#_Toc218674955)

[Crosshairs 9](#_Toc218674956)

[Cget 9](#_Toc218674957)

[Test Procedure – Graph Crosshairs Cget – Valid Option Name 9](#_Toc218674958)

[Test Procedure – Graph Crosshairs Cget – Invalid Option Name 9](#_Toc218674959)

[Configure 9](#_Toc218674960)

[Test Procedure – Graph Crosshairs Configure: Color – Valid Color Name 9](#_Toc218674961)

[Test Procedure – Graph Crosshairs Configure: Color – Invalid Color Name 9](#_Toc218674962)

[Test Procedure – Graph Crosshairs Configure: Dashes – Valid Dash List 9](#_Toc218674963)

[Test Procedure – Graph Crosshairs Configure: Dashes – Empty Dash List 10](#_Toc218674964)

[Test Procedure – Graph Crosshairs Configure: Dashes – Long Dash List 10](#_Toc218674965)

[Test Procedure – Graph Crosshairs Configure: Dashes – Invalid Numerical Dash List 10](#_Toc218674966)

[Test Procedure – Graph Crosshairs Configure: Dashes – Dash List with Characters 10](#_Toc218674967)

[Test Procedure – Graph Crosshairs Configure: Hide – 1 10](#_Toc218674968)

[Test Procedure – Graph Crosshairs Configure: Hide – 0 10](#_Toc218674969)

[Test Procedure – Graph Crosshairs Configure: Hide – True 10](#_Toc218674970)

[Test Procedure – Graph Crosshairs Configure: Hide – False 11](#_Toc218674971)

[Test Procedure – Graph Crosshairs Configure: Hide – Yes 11](#_Toc218674972)

[Test Procedure – Graph Crosshairs Configure: Hide – No 11](#_Toc218674973)

[Test Procedure – Graph Crosshairs Configure: Hide – Invalid Input 11](#_Toc218674974)

[Test Procedure – Graph Crosshairs Configure: Linewidth – Valid Integer Input 11](#_Toc218674975)

[Test Procedure – Graph Crosshairs Configure: Linewidth – Valid Decimal Input 11](#_Toc218674976)

[Test Procedure – Graph Crosshairs Configure: Linewidth – Invalid Numerical Input 11](#_Toc218674977)

[Test Procedure – Graph Crosshairs Configure: Linewidth – Character Input 11](#_Toc218674978)

[Test Procedure – Graph Crosshairs Configure: Position – Valid Coordinates 12](#_Toc218674979)

[Test Procedure – Graph Crosshairs Configure: Position – Invalid Coordinates 12](#_Toc218674980)

[Test Procedure – Graph Crosshairs Configure: Position – Not Coordinate Form 12](#_Toc218674981)

[Off 12](#_Toc218674982)

[On 12](#_Toc218674983)

[Toggle 12](#_Toc218674984)

[Element 12](#_Toc218674985)

[Activate 12](#_Toc218674986)

[Test Procedure – Graph Element Activate: Valid Element Name 12](#_Toc218674987)

[Test Procedure – Graph Element Activate: Invalid Element Name 12](#_Toc218674988)

[Bind 13](#_Toc218674989)

[Test Procedure – Graph Element Bind: Tag, Sequence, Command 13](#_Toc218674990)

[Cget 13](#_Toc218674991)

[Test Procedure – Graph Element Cget: Valid Option Name 13](#_Toc218674992)

[Test Procedure – Graph Element Cget: Invalid Option Name 13](#_Toc218674993)

[Closest 13](#_Toc218674994)

[Test Procedure – Graph Element Closest: Existing Closest Element 13](#_Toc218674995)

[Test Procedure – Graph Element Closest: Existing Closest Element 13](#_Toc218674996)

[Test Procedure – Graph Element Closest: Along – X 13](#_Toc218674997)

[Test Procedure – Graph Element Closest: Along – Y 14](#_Toc218674998)

[Test Procedure – Graph Element Closest: Along – Both 14](#_Toc218674999)

[Test Procedure – Graph Element Closest: Along – Invalid Input 14](#_Toc218675000)

[Test Procedure – Graph Element Closest: Along – Positive Integer Value 14](#_Toc218675001)

[Test Procedure – Graph Element Closest: Along – 1 14](#_Toc218675002)

[Configure 14](#_Toc218675003)

[Test Procedure – Graph Element Configure: Activepen – Valid Pen Name 14](#_Toc218675004)

[Test Procedure – Graph Element Configure: Activepen – Invalid Pen Name 14](#_Toc218675005)

[Test Procedure – Graph Element Configure: Bindtags – Single Tag 15](#_Toc218675006)

[Test Procedure – Graph Element Configure: Bindtags – Tag List 15](#_Toc218675007)

[Test Procedure – Graph Element Configure: Color – Valid Color Name 15](#_Toc218675008)

[Test Procedure – Graph Element Configure: Color – Invalid Color Name 15](#_Toc218675009)

[Test Procedure – Graph Element Configure: Dashes – Valid Dash List 15](#_Toc218675010)

[Test Procedure – Graph Element Configure: Dashes – Empty Dash List 15](#_Toc218675011)

[Test Procedure – Graph Element Configure: Dashes – Long Dash List 16](#_Toc218675012)

[Test Procedure – Graph Element Configure: Dashes – Invalid Numerical Dash List 16](#_Toc218675013)

[Test Procedure – Graph Element Configure: Dashes – Dash List with Characters 16](#_Toc218675014)

[Test Procedure – Graph Element Configure: Data – Valid Coordinate List 16](#_Toc218675015)

[Test Procedure – Graph Element Configure: Data – Invalid Coordinate List 16](#_Toc218675016)

[Test Procedure – Graph Element Configure: Data – Not Proper Form 16](#_Toc218675017)

[Test Procedure – Graph Element Configure: Data – Odd List 16](#_Toc218675018)

[Test Procedure – Graph Element Configure: Fill – Valid Color Name 17](#_Toc218675019)

[Test Procedure – Graph Element Configure: Fill – Invalid Color Name 17](#_Toc218675020)

[Test Procedure – Graph Element Configure: Fill – Defcolor 17](#_Toc218675021)

[Test Procedure – Graph Element Configure: Fill – Empty String 17](#_Toc218675022)

[Test Procedure – Graph Element Configure: Hide – 1 17](#_Toc218675023)

[Test Procedure – Graph Element Configure: Hide – 0 17](#_Toc218675024)

[Test Procedure – Graph Element Configure: Hide – True 17](#_Toc218675025)

[Test Procedure – Graph Element Configure: Hide – False 17](#_Toc218675026)

[Test Procedure – Graph Element Configure: Hide – Yes 18](#_Toc218675027)

[Test Procedure – Graph Element Configure: Hide – No 18](#_Toc218675028)

[Test Procedure – Graph Element Configure: Hide – Invalid Input 18](#_Toc218675029)

[Test Procedure – Graph Element Configure: Label – Valid Text 18](#_Toc218675030)

[Test Procedure – Graph Element Configure: Label – Empty String 18](#_Toc218675031)

[Test Procedure – Graph Element Configure: Label – Default Value 18](#_Toc218675032)

[Test Procedure – Graph Element Configure: Linewidth – Valid Integer Input 18](#_Toc218675033)

[Test Procedure – Graph Element Configure: Linewidth – Valid Decimal Input 19](#_Toc218675034)

[Test Procedure – Graph Element Configure: Linewidth – Invalid Numerical Input 19](#_Toc218675035)

[Test Procedure – Graph Element Configure: Linewidth – Character Input 19](#_Toc218675036)

[Test Procedure – Graph Element Configure: Mapx – Valid Axis Name 19](#_Toc218675037)

[Test Procedure – Graph Element Configure: Mapx – Non-Existent Axis Name 19](#_Toc218675038)

[Test Procedure – Graph Element Configure: Mapx – No Input 19](#_Toc218675039)

[Test Procedure – Graph Element Configure: Mapy – Valid Axis Name 19](#_Toc218675040)

[Test Procedure – Graph Element Configure: Mapy – Non-Existent Axis Name 20](#_Toc218675041)

[Test Procedure – Graph Element Configure: Mapy – No Input 20](#_Toc218675042)

[Test Procedure – Graph Element Configure: Offdash – Valid Color Name 20](#_Toc218675043)

[Test Procedure – Graph Element Configure: Offdash – Invalid Color Name 20](#_Toc218675044)

[Test Procedure – Graph Element Configure: Offdash – Defcolor 20](#_Toc218675045)

[Test Procedure – Graph Element Configure: Offdash – Empty String 20](#_Toc218675046)

[Test Procedure – Graph Element Configure: Outline – Valid Color Name 20](#_Toc218675047)

[Test Procedure – Graph Element Configure: Outline – Invalid Color Name 21](#_Toc218675048)

[Test Procedure – Graph Element Configure: Outline – Defcolor 21](#_Toc218675049)

[Test Procedure – Graph Element Configure: Pen – Valid Pen Name 21](#_Toc218675050)

[Test Procedure – Graph Element Configure: Pen – Invalid Pen Name 21](#_Toc218675051)

[Test Procedure – Graph Element Configure: Outlinewidth – Valid Integer Input 21](#_Toc218675052)

[Test Procedure – Graph Element Configure: Outlinewidth – Valid Decimal Input 21](#_Toc218675053)

[Test Procedure – Graph Element Configure: Outlinewidth – Invalid Numerical Input 21](#_Toc218675054)

[Test Procedure – Graph Element Configure: Outlinewidth – Character Input 22](#_Toc218675055)

[Test Procedure – Graph Element Configure: Scalesymbols – 1 22](#_Toc218675056)

[Test Procedure – Graph Element Configure: Scalesymbols – 0 22](#_Toc218675057)

[Test Procedure – Graph Element Configure: Scalesymbols – True 22](#_Toc218675058)

[Test Procedure – Graph Element Configure: Scalesymbols – False 22](#_Toc218675059)

[Test Procedure – Graph Element Configure: Scalesymbols – Yes 22](#_Toc218675060)

[Test Procedure – Graph Element Configure: Scalesymbols – No 22](#_Toc218675061)

[Test Procedure – Graph Element Configure: Scalesymbols – Invalid Input 23](#_Toc218675062)

[Test Procedure – Graph Element Configure: Smooth – Valid Input 23](#_Toc218675063)

[Test Procedure – Graph Element: Smooth – Invalid Input 23](#_Toc218675064)

[Test Procedure – Graph Element: Stylelist – Valid Pen Name 23](#_Toc218675065)

[Test Procedure – Graph Element: Stylelist – Non-Existent Pen 23](#_Toc218675066)

[Test Procedure – Graph Element Configure: Stylelist – Pen and Weights 23](#_Toc218675067)

[Test Procedure – Graph Element Configure: Symbol – Valid Symbol Name 23](#_Toc218675068)

[Test Procedure – Graph Element Configure: Symbol – Invalid Symbol Name 24](#_Toc218675069)

[Test Procedure – Graph Element Configure: Trace – Valid Direction 24](#_Toc218675070)

[Test Procedure – Graph Element Configure: Trace – Invalid Direction 24](#_Toc218675071)

[Test Procedure – Graph Element Configure: Weights – Valid Vector Name 24](#_Toc218675072)

[Test Procedure – Graph Element Configure: Weights – Invalid Vector Name 24](#_Toc218675073)

[Test Procedure – Graph Element Configure: Weights – Valid List 24](#_Toc218675074)

[Test Procedure – Graph Element Configure: Weights – Invalid Input 24](#_Toc218675075)

[Test Procedure – Graph Element Configure: Xdata – Valid Vector Name 25](#_Toc218675076)

[Test Procedure – Graph Element Configure: Xdata – Invalid Vector Name 25](#_Toc218675077)

[Test Procedure – Graph Element Configure: Xdata – Valid List 25](#_Toc218675078)

[Test Procedure – Graph Element Configure: Xdata – Invalid Input 25](#_Toc218675079)

[Test Procedure – Graph Element Configure: Ydata – Valid Vector Name 25](#_Toc218675080)

[Test Procedure – Graph Element Configure: Ydata – Invalid Vector Name 25](#_Toc218675081)

[Test Procedure – Graph Element Configure: Ydata – Valid List 25](#_Toc218675082)

[Test Procedure – Graph Element Configure: Ydata – Invalid Input 26](#_Toc218675083)

[Test Procedure – Graph Element Configure: Pixels – Positive Integer 26](#_Toc218675084)

[Test Procedure – Graph Element Configure: Pixels – Positive Decimal 26](#_Toc218675085)

[Test Procedure – Graph Element Configure: Pixels – Invalid Numerical Input 26](#_Toc218675086)

[Test Procedure – Graph Element Configure: Pixels – Character Input 26](#_Toc218675087)

[Create 26](#_Toc218675088)

[Test Procedure – Graph Element Create: Unique Name 26](#_Toc218675089)

[Test Procedure – Graph Element Create: Existing Name 26](#_Toc218675090)

[Deactivate 27](#_Toc218675091)

[Test Procedure – Graph Element Deactivate: Valid Element Name 27](#_Toc218675092)

[Test Procedure – Graph Element Deactivate: Invalid Element Name 27](#_Toc218675093)

[Delete 27](#_Toc218675094)

[Test Procedure – Graph Element Delete: Single Element Name 27](#_Toc218675095)

[Test Procedure – Graph Element Delete: Multiple Element Names 27](#_Toc218675096)

[Test Procedure – Graph Element Delete: Invalid Element Name 27](#_Toc218675097)

[Exists 27](#_Toc218675098)

[Test Procedure – Graph Element Exists: Existing Element Name 27](#_Toc218675099)

[Test Procedure – Graph Element Exists: Non-Existent Element Name 28](#_Toc218675100)

[Names 28](#_Toc218675101)

[Test Procedure – Graph Element Names: No Pattern 28](#_Toc218675102)

[Test Procedure – Graph Element Names: Exact Pattern 28](#_Toc218675103)

[Test Procedure – Graph Element Names: Wildcard Pattern 28](#_Toc218675104)

[Test Procedure – Graph Element Names: Incorrect Pattern 28](#_Toc218675105)

[Show 28](#_Toc218675106)

[Test Procedure – Graph Element Show: Single Element Name 28](#_Toc218675107)

[Test Procedure – Graph Element Show: Multiple Element Names 28](#_Toc218675108)

[Test Procedure – Graph Element Show: Non-Existent Element Name 29](#_Toc218675109)

[Type 29](#_Toc218675110)

[Test Procedure – Graph Element Type: Valid Line Element Name 29](#_Toc218675111)

[Test Procedure – Graph Element Type: Valid Bar Element Name 29](#_Toc218675112)

[Test Procedure – Graph Element Type: Invalid Element Name 29](#_Toc218675113)

[Extents 29](#_Toc218675114)

[Grid 30](#_Toc218675115)

[Cget 30](#_Toc218675116)

[Configure Color 30](#_Toc218675117)

[Off 30](#_Toc218675118)

[On 30](#_Toc218675119)

[Toggle 30](#_Toc218675120)

[Inside 30](#_Toc218675121)

[Invtransform 30](#_Toc218675122)

[Legend 31](#_Toc218675123)

[Activate 31](#_Toc218675124)

[Deactivate 31](#_Toc218675125)

[Configure 31](#_Toc218675126)

[Bind 31](#_Toc218675127)

[Cget 31](#_Toc218675128)

[Line 31](#_Toc218675129)

[Marker 31](#_Toc218675130)

[Create 31](#_Toc218675131)

[Destroy 31](#_Toc218675132)

[Exists 31](#_Toc218675133)

[Type 31](#_Toc218675134)

[Names 31](#_Toc218675135)

[Configure 31](#_Toc218675136)

[Bind 31](#_Toc218675137)

[Pen 31](#_Toc218675138)

[Create 31](#_Toc218675139)

[Delete 31](#_Toc218675140)

[Cget 31](#_Toc218675141)

[Configure 31](#_Toc218675142)

[Delete 31](#_Toc218675143)

[Names 31](#_Toc218675144)

[Postscript 31](#_Toc218675145)

[Cget 32](#_Toc218675146)

[Configure 32](#_Toc218675147)

[Output 32](#_Toc218675148)

[Snap 32](#_Toc218675149)

[Transform 32](#_Toc218675150)

[Test Case 1 32](#_Toc218675151)

[Test Procedure – Graph Crosshairs: Cget - Valid Option Name 32](#_Toc218675152)

[Xaxis 32](#_Toc218675153)

[X2axis 32](#_Toc218675154)

[Yaxis 32](#_Toc218675155)

[Y2axis 32](#_Toc218675156)

# Axis

# Bar

# Configure

# Crosshairs

## Cget

### Test Procedure – Graph Crosshairs Cget – Valid Option Name

**Test Case 1**

**Purpose –** Ensure the *crosshairs cget* command works correctly when given a valid crosshairs configuration option name.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.cget.1.1

### Test Procedure – Graph Crosshairs Cget – Invalid Option Name

**Test Case 1**

**Purpose –** Ensure the *crosshairs cget* command works correctly when given an invalid crosshairs configuration option name.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.cget.1.2

## Configure

### Test Procedure – Graph Crosshairs Configure: Color – Valid Color Name

**Test Case 1**

**Purpose –** Ensure the *crosshairs configure -color* command works correctly when given a valid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.1.1

### Test Procedure – Graph Crosshairs Configure: Color – Invalid Color Name

**Test Case 1**

**Purpose –** Ensure the *crosshairs configure -color* command works correctly when given an invalid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.1.2

### Test Procedure – Graph Crosshairs Configure: Dashes – Valid Dash List

**Test Case 2**

**Purpose –** Ensure the *crosshairs configure -dashes* command works correctly when given a valid dash list.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.2.1

### Test Procedure – Graph Crosshairs Configure: Dashes – Empty Dash List

**Test Case 2**

**Purpose –** Ensure the *crosshairs configure -dashes* command works correctly when given an empty dash list.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.2.2

### Test Procedure – Graph Crosshairs Configure: Dashes – Long Dash List

**Test Case 2**

**Purpose –** Ensure the *crosshairs configure -dashes* command works correctly when given a dash list that is too long.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.2.3

### Test Procedure – Graph Crosshairs Configure: Dashes – Invalid Numerical Dash List

**Test Case 2**

**Purpose –** Ensure the *crosshairs configure -dashes* command works correctly when given an invalid numerical dash list.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.2.4

### Test Procedure – Graph Crosshairs Configure: Dashes – Dash List with Characters

**Test Case 2**

**Purpose –** Ensure the *crosshairs configure -dashes* command works correctly when given a dash list with characters.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.2.5

### Test Procedure – Graph Crosshairs Configure: Hide – 1

**Test Case 3**

**Purpose –** Ensure the *crosshairs configure -hide* command works correctly when given 1.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.3.1

### Test Procedure – Graph Crosshairs Configure: Hide – 0

**Test Case 3**

**Purpose –** Ensure the *crosshairs configure -hide* command works correctly when given 0.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs..4.2

### Test Procedure – Graph Crosshairs Configure: Hide – True

**Test Case 3**

**Purpose –** Ensure the *crosshairs configure -hide* command works correctly when given true.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.3.3

### Test Procedure – Graph Crosshairs Configure: Hide – False

**Test Case 3**

**Purpose –** Ensure the *crosshairs configure -hide* command works correctly when given false.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.3.4

### Test Procedure – Graph Crosshairs Configure: Hide – Yes

**Test Case 3**

**Purpose –** Ensure the *crosshairs configure -hide* command works correctly when given yes.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.3.5

### Test Procedure – Graph Crosshairs Configure: Hide – No

**Test Case 3**

**Purpose –** Ensure the *crosshairs configure -hide* command works correctly when given no.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.3.6

### Test Procedure – Graph Crosshairs Configure: Hide – Invalid Input

**Test Case 3**

**Purpose –** Ensure the *crosshairs configure -hide* command works correctly when given an invalid input value.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.3.7

### Test Procedure – Graph Crosshairs Configure: Linewidth – Valid Integer Input

**Test Case 4**

**Purpose –** Ensure the *crosshairs configure -linewidth* command works correctly when given an integer pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.4.1

### Test Procedure – Graph Crosshairs Configure: Linewidth – Valid Decimal Input

**Test Case 4**

**Purpose –** Ensure the *crosshairs configure -linewidth* command works correctly when given a decimal pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.4.2

### Test Procedure – Graph Crosshairs Configure: Linewidth – Invalid Numerical Input

**Test Case 4**

**Purpose –** Ensure the *crosshairs configure -linewidth* command works correctly when given an invalid numerical pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.4.3

### Test Procedure – Graph Crosshairs Configure: Linewidth – Character Input

**Test Case 4**

**Purpose –** Ensure the *crosshairs configure -linewidth* command works correctly when given a character as input.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.4.4

### Test Procedure – Graph Crosshairs Configure: Position – Valid Coordinates

**Test Case 5**

**Purpose –** Ensure the *crosshairs configure -position* command works correctly when given valid coordinates as input.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.5.1

### Test Procedure – Graph Crosshairs Configure: Position – Invalid Coordinates

**Test Case 5**

**Purpose –** Ensure the *crosshairs configure -position* command works correctly when given invalid coordinates as input.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.5.2

### Test Procedure – Graph Crosshairs Configure: Position – Not Coordinate Form

**Test Case 5**

**Purpose –** Ensure the *crosshairs configure -position* command works correctly when given input that is not in coordinate form (@x,y).

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.configure.5.3

## Off

## On

## Toggle

# Element

## Activate

### Test Procedure – Graph Element Activate: Valid Element Name

**Test Case 1**

**Purpose –** Ensure the *element activate* command works correctly when given a valid element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.activate.1.1

### Test Procedure – Graph Element Activate: Invalid Element Name

**Test Case 1**

**Purpose –** Ensure the *element activate* command works correctly when given an invalid element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.activate.1.2

## Bind

### Test Procedure – Graph Element Bind: Tag, Sequence, Command

**Test Case 1**

**Purpose –** Ensure the *element bind* command works correctly when given a tag name, an event sequence, and a command.

**Special Requirements –** None

**TclTest –** RBC.graph.element.bind.1.1

## Cget

### Test Procedure – Graph Element Cget: Valid Option Name

**Test Case 1**

**Purpose –** Ensure the *element cget* command works correctly when given a valid element configuration option name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.cget.1.1

### Test Procedure – Graph Element Cget: Invalid Option Name

**Test Case 1**

**Purpose –** Ensure the *crosshairs cget* command works correctly when given an invalid element configuration option name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.cget.1.2

## Closest

### Test Procedure – Graph Element Closest: Existing Closest Element

**Test Case 1**

**Purpose –** Ensure the *element closest* command works correctly when a closest element exists.

**Special Requirements –** None

**TclTest –** RBC.graph.element.closest.1.1

### Test Procedure – Graph Element Closest: Existing Closest Element

**Test Case 1**

**Purpose –** Ensure the *element closest* command works correctly when a closest element does not exist.

**Special Requirements –** None

**TclTest –** RBC.graph.element.closest.1.2

### Test Procedure – Graph Element Closest: Along – X

**Test Case 2**

**Purpose –** Ensure the *element closest -along* command works correctly when given x as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.closest.2.1

### Test Procedure – Graph Element Closest: Along – Y

**Test Case 2**

**Purpose –** Ensure the *element closest -along* command works correctly when given y as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.closest.2.2

### Test Procedure – Graph Element Closest: Along – Both

**Test Case 2**

**Purpose –** Ensure the *element closest -along* command works correctly when given both as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.closest.2.3

### Test Procedure – Graph Element Closest: Along – Invalid Input

**Test Case 2**

**Purpose –** Ensure the *element closest -along* command works correctly when given invalid input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.closest.2.4

### Test Procedure – Graph Element Closest: Along – Positive Integer Value

**Test Case 3**

**Purpose –** Ensure the *element closest -halo* command works correctly when given a positive integer value as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.closest.3.1

### Test Procedure – Graph Element Closest: Along – 1

**Test Case 4**

**Purpose –** Ensure the *element closest -along* command works correctly when given 1 as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.11.1

## Configure

### Test Procedure – Graph Element Configure: Activepen – Valid Pen Name

**Test Case 1**

**Purpose –** Ensure the *element configure -activepen* command works correctly when given a valid pen name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.1.1

### Test Procedure – Graph Element Configure: Activepen – Invalid Pen Name

**Test Case 1**

**Purpose –** Ensure the *element configure -activepen* command works correctly when given an invalid pen name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.1.2

### Test Procedure – Graph Element Configure: Bindtags – Single Tag

**Test Case 2**

**Purpose –** Ensure the *element configure -bindtags* command works correctly when given a single tag name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.2.1

### Test Procedure – Graph Element Configure: Bindtags – Tag List

**Test Case 2**

**Purpose –** Ensure the *element configure -bindtags* command works correctly when given a list of tag names.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.2.2

### Test Procedure – Graph Element Configure: Color – Valid Color Name

**Test Case 3**

**Purpose –** Ensure the *element configure -color* command works correctly when given a valid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.3.1

### Test Procedure – Graph Element Configure: Color – Invalid Color Name

**Test Case 3**

**Purpose –** Ensure the *element configure -color* command works correctly when given an invalid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.3.2

### Test Procedure – Graph Element Configure: Dashes – Valid Dash List

**Test Case 4**

**Purpose –** Ensure the *element configure -dashes* command works correctly when given a valid dash list.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.4.1

### Test Procedure – Graph Element Configure: Dashes – Empty Dash List

**Test Case 4**

**Purpose –** Ensure the *element configure -dashes* command works correctly when given an empty dash list.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.4.2

### Test Procedure – Graph Element Configure: Dashes – Long Dash List

**Test Case 4**

**Purpose –** Ensure the *element configure -dashes* command works correctly when given a dash list that is too long.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.4.3

### Test Procedure – Graph Element Configure: Dashes – Invalid Numerical Dash List

**Test Case 4**

**Purpose –** Ensure the *element configure -dashes* command works correctly when given an invalid numerical dash list.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.4.4

### Test Procedure – Graph Element Configure: Dashes – Dash List with Characters

**Test Case 4**

**Purpose –** Ensure the *element configure -dashes* command works correctly when given a dash list with characters.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.4.5

### Test Procedure – Graph Element Configure: Data – Valid Coordinate List

**Test Case 5**

**Purpose –** Ensure the *element configure -data* command works correctly when given a valid coordinate list as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.5.1

### Test Procedure – Graph Element Configure: Data – Invalid Coordinate List

**Test Case 5**

**Purpose –** Ensure the *element configure -data* command works correctly when given an invalid coordinate list as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.5.2

### Test Procedure – Graph Element Configure: Data – Not Proper Form

**Test Case 5**

**Purpose –** Ensure the *element configure -data* command works correctly when given input that is not in the proper form (x1 y1 x2 y1).

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.5.3

### Test Procedure – Graph Element Configure: Data – Odd List

**Test Case 5**

**Purpose –** Ensure the *element configure -data* command works correctly when given input that does not have an even number of values.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.5.4

### Test Procedure – Graph Element Configure: Fill – Valid Color Name

**Test Case 6**

**Purpose –** Ensure the *element configure -fill* command works correctly when given a valid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.6.1

### Test Procedure – Graph Element Configure: Fill – Invalid Color Name

**Test Case 6**

**Purpose –** Ensure the *element configure -fill* command works correctly when given an invalid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.6.2

### Test Procedure – Graph Element Configure: Fill – Defcolor

**Test Case 6**

**Purpose –** Ensure the *element configure -fill* command works correctly when given ‘defcolor’ as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.6.3

### Test Procedure – Graph Element Configure: Fill – Empty String

**Test Case 6**

**Purpose –** Ensure the *element configure -fill* command works correctly when given “” as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.6.4

### Test Procedure – Graph Element Configure: Hide – 1

**Test Case 7**

**Purpose –** Ensure the *element configure -hide* command works correctly when given 1.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.7.1

### Test Procedure – Graph Element Configure: Hide – 0

**Test Case 7**

**Purpose –** Ensure the *element configure -hide* command works correctly when given 0.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.7.2

### Test Procedure – Graph Element Configure: Hide – True

**Test Case 7**

**Purpose –** Ensure the *element configure -hide* command works correctly when given true.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.7.3

### Test Procedure – Graph Element Configure: Hide – False

**Test Case 7**

**Purpose –** Ensure the *element configure -hide* command works correctly when given false.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.7.4

### Test Procedure – Graph Element Configure: Hide – Yes

**Test Case 7**

**Purpose –** Ensure the *element configure -hide* command works correctly when given yes.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.7.5

### Test Procedure – Graph Element Configure: Hide – No

**Test Case 7**

**Purpose –** Ensure the *element configure -hide* command works correctly when given no.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.7.6

### Test Procedure – Graph Element Configure: Hide – Invalid Input

**Test Case 7**

**Purpose –** Ensure the *element configure -hide* command works correctly when given an invalid input value.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.7.7

### Test Procedure – Graph Element Configure: Label – Valid Text

**Test Case 8**

**Purpose –** Ensure the *element configure -label* command works correctly when given valid text.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.8.1

### Test Procedure – Graph Element Configure: Label – Empty String

**Test Case 8**

**Purpose –** Ensure the *element configure -label* command works correctly when given the empty string as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.8.2

### Test Procedure – Graph Element Configure: Label – Default Value

**Test Case 8**

**Purpose –** Ensure the *element configure -label* command does not change the default value when not given any input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.8.3

### Test Procedure – Graph Element Configure: Linewidth – Valid Integer Input

**Test Case 9**

**Purpose –** Ensure the *element configure -linewidth* command works correctly when given an integer pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.9.1

### Test Procedure – Graph Element Configure: Linewidth – Valid Decimal Input

**Test Case 9**

**Purpose –** Ensure the *element configure -linewidth* command works correctly when given a decimal pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.9.2

### Test Procedure – Graph Element Configure: Linewidth – Invalid Numerical Input

**Test Case 9**

**Purpose –** Ensure the *element configure -linewidth* command works correctly when given an invalid numerical pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.9.3

### Test Procedure – Graph Element Configure: Linewidth – Character Input

**Test Case 9**

**Purpose –** Ensure the *element configure -linewidth* command works correctly when given a character as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.9.4

### Test Procedure – Graph Element Configure: Mapx – Valid Axis Name

**Test Case 10**

**Purpose –** Ensure the *element configure -mapx* command works correctly when given valid axis name as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.10.1

### Test Procedure – Graph Element Configure: Mapx – Non-Existent Axis Name

**Test Case 10**

**Purpose –** Ensure the *element configure -mapx* command works correctly when given a non-existent axis name as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.10.2

### Test Procedure – Graph Element Configure: Mapx – No Input

**Test Case 10**

**Purpose –** Ensure the *element configure -mapx* command works correctly when not input is given.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.10.3

### Test Procedure – Graph Element Configure: Mapy – Valid Axis Name

**Test Case 11**

**Purpose –** Ensure the *element configure -mapy* command works correctly when given valid axis name as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.11.1

### Test Procedure – Graph Element Configure: Mapy – Non-Existent Axis Name

**Test Case 11**

**Purpose –** Ensure the *element configure -mapy* command works correctly when given a non-existent axis name as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.11.2

### Test Procedure – Graph Element Configure: Mapy – No Input

**Test Case 11**

**Purpose –** Ensure the *element configure -mapy* command works correctly when not input is given.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.11.3

### Test Procedure – Graph Element Configure: Offdash – Valid Color Name

**Test Case 12**

**Purpose –** Ensure the *element configure -offdash* command works correctly when given a valid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.12.1

### Test Procedure – Graph Element Configure: Offdash – Invalid Color Name

**Test Case 12**

**Purpose –** Ensure the *element configure -offdash* command works correctly when given an invalid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.12.2

### Test Procedure – Graph Element Configure: Offdash – Defcolor

**Test Case 12**

**Purpose –** Ensure the *element configure -offdash* command works correctly when given ‘defcolor’ as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.12.3

### Test Procedure – Graph Element Configure: Offdash – Empty String

**Test Case 12**

**Purpose –** Ensure the *element configure -offdash* command works correctly when given “” as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.12.4

### Test Procedure – Graph Element Configure: Outline – Valid Color Name

**Test Case 13**

**Purpose –** Ensure the *element configure -outline* command works correctly when given a valid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.13.1

### Test Procedure – Graph Element Configure: Outline – Invalid Color Name

**Test Case 13**

**Purpose –** Ensure the *element configure -outline* command works correctly when given an invalid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.13.2

### Test Procedure – Graph Element Configure: Outline – Defcolor

**Test Case 13**

**Purpose –** Ensure the *element configure -outline* command works correctly when given ‘defcolor’ as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.13.3

### Test Procedure – Graph Element Configure: Pen – Valid Pen Name

**Test Case 14**

**Purpose –** Ensure the *element configure -pen* command works correctly when given a valid pen name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.14.1

### Test Procedure – Graph Element Configure: Pen – Invalid Pen Name

**Test Case 14**

**Purpose –** Ensure the *element configure -pen* command works correctly when given an invalid pen name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.14.2

### Test Procedure – Graph Element Configure: Outlinewidth – Valid Integer Input

**Test Case 15**

**Purpose –** Ensure the *element configure -outlinewidth* command works correctly when given an integer pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.15.1

### Test Procedure – Graph Element Configure: Outlinewidth – Valid Decimal Input

**Test Case 15**

**Purpose –** Ensure the *element configure -outlinewidth* command works correctly when given a decimal pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.15.2

### Test Procedure – Graph Element Configure: Outlinewidth – Invalid Numerical Input

**Test Case 15**

**Purpose –** Ensure the *element configure -outlinewidth* command works correctly when given an invalid numerical pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.15.3

### Test Procedure – Graph Element Configure: Outlinewidth – Character Input

**Test Case 15**

**Purpose –** Ensure the *element configure -outlinewidth* command works correctly when given a character as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.15.4

### Test Procedure – Graph Element Configure: Scalesymbols – 1

**Test Case 16**

**Purpose –** Ensure the *element configure -scalesymbols* command works correctly when given 1.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.16.1

### Test Procedure – Graph Element Configure: Scalesymbols – 0

**Test Case 16**

**Purpose –** Ensure the *element configure -scalesymbols* command works correctly when given 0.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.16.2

### Test Procedure – Graph Element Configure: Scalesymbols – True

**Test Case 16**

**Purpose –** Ensure the *element configure -scalesymbols* command works correctly when given true.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.16.3

### Test Procedure – Graph Element Configure: Scalesymbols – False

**Test Case 16**

**Purpose –** Ensure the *element configure -scalesymbols* command works correctly when given false.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.16.4

### Test Procedure – Graph Element Configure: Scalesymbols – Yes

**Test Case 16**

**Purpose –** Ensure the *element configure -scalesymbols* command works correctly when given yes.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.16.5

### Test Procedure – Graph Element Configure: Scalesymbols – No

**Test Case 16**

**Purpose –** Ensure the *element configure -scalesymbols* command works correctly when given no.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.16.6

### Test Procedure – Graph Element Configure: Scalesymbols – Invalid Input

**Test Case 16**

**Purpose –** Ensure the *element configure -scalesymbols* command works correctly when given an invalid input value.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.16.7

### Test Procedure – Graph Element Configure: Smooth – Valid Input

**Test Case 17**

**Purpose –** Ensure the *element configure -smooth* command works correctly when given valid input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.17.1

### Test Procedure – Graph Element: Smooth – Invalid Input

**Test Case 17**

**Purpose –** Ensure the *element configure -smooth* command works correctly when given invalid input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.17.2

### Test Procedure – Graph Element: Stylelist – Valid Pen Name

**Test Case 18**

**Purpose –** Ensure the *element configure -styles* command works correctly when given a valid pen name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.18.1

### Test Procedure – Graph Element: Stylelist – Non-Existent Pen

**Test Case 18**

**Purpose –** Ensure the *element configure -styles* command works correctly when given an invalid pen name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.18.2

### Test Procedure – Graph Element Configure: Stylelist – Pen and Weights

**Test Case 18**

**Purpose –** Ensure the *element configure -styles* command works correctly when given a valid pen name and a minimum and maximum element weight range.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.18.3

### Test Procedure – Graph Element Configure: Symbol – Valid Symbol Name

**Test Case 19**

**Purpose –** Ensure the *element configure -symbol* command works correctly when given a valid symbol name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.19.1

### Test Procedure – Graph Element Configure: Symbol – Invalid Symbol Name

**Test Case 19**

**Purpose –** Ensure the *element configure -symbol* command works correctly when given an invalid symbol name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.19.2

### Test Procedure – Graph Element Configure: Trace – Valid Direction

**Test Case 20  
Purpose –** Ensure the *element configure -trace* command works correctly when given a valid direction.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.20.1

### Test Procedure – Graph Element Configure: Trace – Invalid Direction

**Test Case 20**

**Purpose –** Ensure the *element configure -trace* command works correctly when given an invalid direction.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.20.2

### Test Procedure – Graph Element Configure: Weights – Valid Vector Name

**Test Case 21**

**Purpose –** Ensure the *element configure -weights* command works correctly when given a valid vector name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.21.1

### Test Procedure – Graph Element Configure: Weights – Invalid Vector Name

**Test Case 21**

**Purpose –** Ensure the *element configure -weights* command works correctly when given an invalid vector name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.21.2

### Test Procedure – Graph Element Configure: Weights – Valid List

**Test Case 21**

**Purpose –** Ensure the *element configure -weights* command works correctly when given a valid list of numerical values.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.21.3

### Test Procedure – Graph Element Configure: Weights – Invalid Input

**Test Case 21**

**Purpose –** Ensure the *element configure -weights* command works correctly when given an invalid list.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.21.4

### Test Procedure – Graph Element Configure: Xdata – Valid Vector Name

**Test Case 22**

**Purpose –** Ensure the *element configure -xdata* command works correctly when given a valid vector name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.22.1

### Test Procedure – Graph Element Configure: Xdata – Invalid Vector Name

**Test Case 22**

**Purpose –** Ensure the *element configure -xdata* command works correctly when given an invalid vector name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.22.2

### Test Procedure – Graph Element Configure: Xdata – Valid List

**Test Case 22**

**Purpose –** Ensure the *element configure -xdata* command works correctly when given a valid list of numerical values.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.22.3

### Test Procedure – Graph Element Configure: Xdata – Invalid Input

**Test Case 22**

**Purpose –** Ensure the *element configure -xdata* command works correctly when given an invalid list.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.22.4

### Test Procedure – Graph Element Configure: Ydata – Valid Vector Name

**Test Case 23**

**Purpose –** Ensure the *element configure -ydata* command works correctly when given a valid vector name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.23.1

### Test Procedure – Graph Element Configure: Ydata – Invalid Vector Name

**Test Case 23**

**Purpose –** Ensure the *element configure -ydata* command works correctly when given an invalid vector name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.23.2

### Test Procedure – Graph Element Configure: Ydata – Valid List

**Test Case 23**

**Purpose –** Ensure the *element configure -ydata* command works correctly when given a valid list of numerical values.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.23.3

### Test Procedure – Graph Element Configure: Ydata – Invalid Input

**Test Case 23**

**Purpose –** Ensure the *element configure -ydata* command works correctly when given an invalid list.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.23.4

### Test Procedure – Graph Element Configure: Pixels – Positive Integer

**Test Case 24**

**Purpose –** Ensure the *element configure -pixels* command works correctly when given a positive integer.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.24.1

### Test Procedure – Graph Element Configure: Pixels – Positive Decimal

**Test Case 24**

**Purpose –** Ensure the *element configure -pixels* command works correctly when given a positive decimal.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.24.2

### Test Procedure – Graph Element Configure: Pixels – Invalid Numerical Input

**Test Case 24**

**Purpose –** Ensure the *element configure -pixels* command works correctly when given invalid numerical input (e.g. a negative value).

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.24.3

### Test Procedure – Graph Element Configure: Pixels – Character Input

**Test Case 24**

**Purpose –** Ensure the *element configure -pixels* command works correctly when given characters as input.

**Special Requirements –** None

**TclTest –** RBC.graph.element.configure.24.4

## Create

### Test Procedure – Graph Element Create: Unique Name

**Test Case 1**

**Purpose –** Ensure the *element create* command works correctly when given a unique element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.create.1.1

### Test Procedure – Graph Element Create: Existing Name

**Test Case 1**

**Purpose –** Ensure the *element create* command works correctly when given an existing element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.create.1.2

## Deactivate

### Test Procedure – Graph Element Deactivate: Valid Element Name

**Test Case 1**

**Purpose –** Ensure the *element deactivate* command works correctly when given a valid element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.deactivate.1.1

### Test Procedure – Graph Element Deactivate: Invalid Element Name

**Test Case 1**

**Purpose –** Ensure the *element deactivate* command works correctly when given an invalid element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.deactivate.1.2

## Delete

### Test Procedure – Graph Element Delete: Single Element Name

**Test Case 1**

**Purpose –** Ensure the *element delete* command works correctly when given a single element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.delete.1.1

### Test Procedure – Graph Element Delete: Multiple Element Names

**Test Case 1**

**Purpose –** Ensure the *element delete* command works correctly when given multiple element names.

**Special Requirements –** None

**TclTest –** RBC.graph.element.delete.1.2

### Test Procedure – Graph Element Delete: Invalid Element Name

**Test Case 1**

**Purpose –** Ensure the *element delete* command works correctly when given an invalid element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.delete.1.3

## Exists

### Test Procedure – Graph Element Exists: Existing Element Name

**Test Case 1**

**Purpose –** Ensure the *element delete* command works correctly when given an existing element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.exists.1.1

### Test Procedure – Graph Element Exists: Non-Existent Element Name

**Test Case 1**

**Purpose –** Ensure the *element delete* command works correctly when given a non-existent element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.exists.1.2

## Names

### Test Procedure – Graph Element Names: No Pattern

**Test Case 1**

**Purpose –** Ensure the *element names* command works correctly when no pattern is given.

**Special Requirements –** None

**TclTest –** RBC.graph.element.names.1.1

### Test Procedure – Graph Element Names: Exact Pattern

**Test Case 1**

**Purpose –** Ensure the *element names* command works correctly when an exact pattern is given.

**Special Requirements –** None

**TclTest –** RBC.graph.element.names.1.2

### Test Procedure – Graph Element Names: Wildcard Pattern

**Test Case 1**

**Purpose –** Ensure the *element names* command works correctly when a wildcard pattern is given.

**Special Requirements –** None

**TclTest –** RBC.graph.element.names.1.3

### Test Procedure – Graph Element Names: Incorrect Pattern

**Test Case 1**

**Purpose –** Ensure the *element names* command works correctly when an incorrect pattern is given.

**Special Requirements –** None

**TclTest –** RBC.graph.element.names.1.4

## Show

### Test Procedure – Graph Element Show: Single Element Name

**Test Case 1**

**Purpose –** Ensure the *element show* command works correctly when given a single element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.show.1.1

### Test Procedure – Graph Element Show: Multiple Element Names

**Test Case 1**

**Purpose –** Ensure the *element show* command works correctly when given multiple element names.

**Special Requirements –** None

**TclTest –** RBC.graph.element.show.1.2

### Test Procedure – Graph Element Show: Non-Existent Element Name

**Test Case 1**

**Purpose –** Ensure the *element show* command works correctly when given a non-existent element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.show.1.3

## Type

### Test Procedure – Graph Element Type: Valid Line Element Name

**Test Case 1**

**Purpose –** Ensure the *element type* command works correctly when given the name of a valid line element.

**Special Requirements –** None

**TclTest –** RBC.graph.element.type.1.1

### Test Procedure – Graph Element Type: Valid Bar Element Name

**Test Case 1**

**Purpose –** Ensure the *element type* command works correctly when given the name of a valid bar element.

**Special Requirements –** None

**TclTest –** RBC.graph.element.type.1.2

### Test Procedure – Graph Element Type: Invalid Element Name

**Test Case 1**

**Purpose –** Ensure the *element type* command works correctly when given an invalid element name.

**Special Requirements –** None

**TclTest –** RBC.graph.element.type.1.3

# Grid

## Cget

### Test Procedure – Graph Grid Cget: Valid Option Name

**Test Case 1**

**Purpose –** Ensure the *grid cget* command works correctly when given a valid grid configuration option name.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.1.1

### Test Procedure – Graph Grid Cget: Invalid Option Name

**Test Case 1**

**Purpose –** Ensure the *grid cget* command works correctly when given an invalid grid configuration option name.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.1.2

## Configure

### Test Procedure – Graph Grid Configure: Color – Valid Color Name

**Test Case 1**

**Purpose –** Ensure the *grid configure -color* command works correctly when given a valid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.1.1

### Test Procedure – Graph Grid Configure: Color – Invalid Color Name

**Test Case 1**

**Purpose –** Ensure the *grid configure -color* command works correctly when given an invalid color name.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.1.2

### Test Procedure – Graph Grid Configure: Dashes – Valid Dash List

**Test Case 2**

**Purpose –** Ensure the *grid configure -dashes* command works correctly when given a valid dash list.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.2.1

### Test Procedure – Graph Grid Configure: Color – Empty Dash List

**Test Case 2**

**Purpose –** Ensure the *grid configure -dashes* command works correctly when given an empty dash list.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.2.2

### Test Procedure – Graph Grid Configure: Dashes – Long Dash List

**Test Case 2**

**Purpose –** Ensure the *grid configure -dashes* command works correctly when given a dash list that is too long.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.2.3

### Test Procedure – Graph Grid Configure: Dashes – Invalid Numerical Dash List

**Test Case 2**

**Purpose –** Ensure the *grid configure -dashes* command works correctly when given an invalid numerical dash list.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.2.4

### Test Procedure – Graph Grid Configure: Dashes – Dash List with Characters

**Test Case 2**

**Purpose –** Ensure the *grid configure -dashes* command works correctly when given a dash list with characters.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.2.5

### Test Procedure – Graph Grid Configure: Hide – 1

**Test Case 2**

**Purpose –** Ensure the *grid configure -hide* command works correctly when given 1.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.3.1

### Test Procedure – Graph Grid Configure: Hide – 0

**Test Case 3**

**Purpose –** Ensure the *grid configure -hide* command works correctly when given 0.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.3.2

### Test Procedure – Graph Grid Configure: Hide – True

**Test Case 3**

**Purpose –** Ensure the *grid configure -hide* command works correctly when given true.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.3.3

### Test Procedure – Graph Grid Configure: Hide – False

**Test Case 3**

**Purpose –** Ensure the *grid configure -hide* command works correctly when given false.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.3.4

### Test Procedure – Graph Grid Configure: Hide – Yes

**Test Case 3**

**Purpose –** Ensure the *grid configure -hide* command works correctly when given yes.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.3.5

### Test Procedure – Graph Grid Configure: Hide – No

**Test Case 3**

**Purpose –** Ensure the *grid configure -hide* command works correctly when given no.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.3.6

### Test Procedure – Graph Grid Configure: Hide – Invalid Input

**Test Case 3**

**Purpose –** Ensure the *grid configure -hide* command works correctly when given an invalid input value.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.3.7

### Test Procedure – Graph Grid Configure: Linewidth – Valid Integer Input

**Test Case 4**

**Purpose –** Ensure the *grid configure -linewidth* command works correctly when given an integer pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.4.1

### Test Procedure – Graph Grid Configure: Linewidth – Valid Decimal Input

**Test Case 4**

**Purpose –** Ensure the *grid configure -linewidth* command works correctly when given a decimal pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.4.2

### Test Procedure – Graph Grid Configure: Linewidth – Invalid Numerical Input

**Test Case 4**

**Purpose –** Ensure the *grid configure -linewidth* command works correctly when given an invalid numerical pixel value.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.4.3

### Test Procedure – Graph Grid Configure: Linewidth – Character Input

**Test Case 4**

**Purpose –** Ensure the *grid configure -linewidth* command works correctly when given a character as input.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.4.4

#### Test Case 5

### Test Procedure – Graph Grid Configure: Mapx – Valid Axis Name

**Purpose –** Ensure the *grid configure -mapx* command works correctly when given valid axis name as input.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.5.1

### Test Procedure – Graph Grid Configure: Mapx – Non-Existent Axis Name

**Purpose –** Ensure the *grid configure -mapx* command works correctly when given a non-existent axis name as input.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.5.2

### Test Procedure – Graph Grid Configure: Mapx – Empty String

**Purpose –** Ensure the *grid configure -mapx* command works correctly when given the empty string as input.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.5.3

### Test Procedure – Graph Grid Configure: Mapx – No Input

**Purpose –** Ensure the *grid configure -mapx* command works correctly when not input is given.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.5.4

#### Test Case 6

### Test Procedure – Graph Grid Configure: Mapy – Valid Axis Name

**Purpose –** Ensure the *grid configure -mapy* command works correctly when given valid axis name as input.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.6.1

### Test Procedure – Graph Grid Configure: Mapy – Non-Existent Axis Name

**Purpose –** Ensure the *grid configure -mapy* command works correctly when given a non-existent axis name as input.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.6.2

### Test Procedure – Graph Grid Configure: Mapy – Empty String

**Purpose –** Ensure the *grid configure -mapy* command works correctly when given the empty string as input.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.6.3

### Test Procedure – Graph Grid Configure: Mapy – No Input

**Purpose –** Ensure the *grid configure -mapy* command works correctly when not input is given.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.6.4

#### Test Case 7

### Test Procedure – Graph Grid Configure: Minor – 1

**Purpose –** Ensure the *grid configure -minor* command works correctly when given 1.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.7.1

### Test Procedure – Graph Grid Configure: Minor – 0

**Purpose –** Ensure the *grid configure -minor* command works correctly when given 0.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.7.2

### Test Procedure – Graph Grid Configure: Minor – True

**Purpose –** Ensure the *grid configure -minor* command works correctly when given true.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.7.3

### Test Procedure – Graph Grid Configure: Minor – False

**Purpose –** Ensure the *grid configure -minor* command works correctly when given false.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.7.4

### Test Procedure – Graph Grid Configure: Minor – Yes

**Purpose –** Ensure the *grid configure -minor* command works correctly when given yes.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.7.5

### Test Procedure – Graph Grid Configure: Minor – No

**Purpose –** Ensure the *grid configure -minor* command works correctly when given no.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.7.6

### Test Procedure – Graph Grid Configure: Minor – Invalid Input

**Purpose –** Ensure the *grid configure -minor* command works correctly when given an invalid input value.

**Special Requirements –** None

**TclTest –** RBC.graph.grid.configure.7.7

# Legend

## Activate (Test Case 1)

### Test Procedure – Activate All Returns All Activated Elements

**Purpose –** Ensure that activating all legend elements returns all legend elements.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.1.1

### Test Procedure – Activate Returns Subset of Activated Elements

**Purpose –** Ensure that activating a subset of legend elements returns only the activated elements.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.1.2

### Test Procedure – Activate Returns All Activated Elements

**Purpose –** Ensure that activating a legend element returns all active legend elements not only the ones that were just activated.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.1.3

## Bind (Test Case 4)

### Test Procedure – Bind Legend Element

**Purpose –** Ensure that bindings can be created for a legend element.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.4.1

### Test Procedure – Bind Query for Sequence and Element

**Purpose –** Ensure that bindings can be queried for a sequence and legend element.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.4.2

### Test Procedure – Bind Command Append

**Purpose –** Ensure that bindings can be appended for a sequence and legend element.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.4.3

### Test Procedure – Bind Command Overwrite

**Purpose –** Ensure that bindings are overwritten for a sequence and legend element.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.4.4

### Test Procedure – Bind Query for Element

**Purpose –** Ensure that bound sequences can be queried for just a legend element.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.4.5

### Test Procedure – Bind Arbitrary Tag Name

**Purpose –** Ensure that bindings can be created for an arbitrary tag.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.4.6

## Cget (Test Case 5)

### Test Procedure – Cget Default Option

**Purpose –** Ensure that cget works for default values.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.5.1

### Test Procedure – Cget Configured Option

**Purpose –** Ensure that cget works with an explicitly set option.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.5.2

## Deactivate (Test Case 2)

### Test Procedure – Deactivate All Elements

**Purpose –** Ensure that deactivating all legend elements works properly.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.2.1

### Test Procedure – Deactivate Subset of Elements

**Purpose –** Ensure that deactivating a subset of legend elements deactivates only that subset.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.2.2

## Configure (Test Case 3)

### Test Procedure – Configure Active Background

**Purpose –** Ensure that the activebackground configuration works for valid colors.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.1

### Test Procedure – Configure Active Border Width

**Purpose –** Ensure that the activeborderwidth configuration works for valid widths.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.2

### Test Procedure – Configure Active Foreground

**Purpose –** Ensure that the activeforeground configuration works for valid colors.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.3

### Test Procedure – Configure Active Relief Raised

**Purpose –** Ensure that the activerelief configuration works for raised reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.4

### Test Procedure – Configure Active Relief Flat

**Purpose –** Ensure that the activerelief configuration works for flat reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.5

### Test Procedure – Configure Active Relief Grooved

**Purpose –** Ensure that the activerelief configuration works for grooved reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.6

### Test Procedure – Configure Active Relief Ridged

**Purpose –** Ensure that the activerelief configuration works for ridged reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.7

### Test Procedure – Configure Active Relief Solid

**Purpose –** Ensure that the activerelief configuration works for solid reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.8

### Test Procedure – Configure Active Relief Sunken

**Purpose –** Ensure that the activerelief configuration works for sunken reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.9

### Test Procedure – Configure Anchor Center

**Purpose –** Ensure that the anchor configuration works for center anchor.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.10

### Test Procedure – Configure Anchor North

**Purpose –** Ensure that the anchor configuration works for north anchor.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.11

### Test Procedure – Configure Anchor Northeast

**Purpose –** Ensure that the anchor configuration works for northeast anchor.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.12

### Test Procedure – Configure Anchor Northwest

**Purpose –** Ensure that the anchor configuration works for northwest anchor.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.13

### Test Procedure – Configure Anchor South

**Purpose –** Ensure that the anchor configuration works for south anchor.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.14

### Test Procedure – Configure Anchor Southeast

**Purpose –** Ensure that the anchor configuration works for southeast anchor.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.15

### Test Procedure – Configure Anchor Southwest

**Purpose –** Ensure that the anchor configuration works for southwest anchor.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.16

### Test Procedure – Configure Anchor East

**Purpose –** Ensure that the anchor configuration works for east anchor.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.17

### Test Procedure – Configure Anchor West

**Purpose –** Ensure that the anchor configuration works for west anchor.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.18

### Test Procedure – Configure No Background

**Purpose –** Ensure that the background configuration works for no background.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.19

### Test Procedure – Configure Background

**Purpose –** Ensure that the background configuration works for valid colors.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.20

### Test Procedure – Configure Borderwidth

**Purpose –** Ensure that the borderwidth configuration works for valid width.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.21

### Test Procedure – Configure Font

**Purpose –** Ensure that the font configuration works for valid font strings.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.22

### Test Procedure – Configure Foreground

**Purpose –** Ensure that the foreground configuration works for valid colors.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.23

### Test Procedure – Configure Hidden

**Purpose –** Ensure that the hide configuration works for true.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.24

### Test Procedure – Configure Not Hidden

**Purpose –** Ensure that the hide configuration works for false.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.25

### Test Procedure – Configure Ipadx Single

**Purpose –** Ensure that the ipadx configuration works for a single value.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.26

### Test Procedure – Configure Ipadx Multiple

**Purpose –** Ensure that the ipadx configuration works for multiple values.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.27

### Test Procedure – Configure Ipady Single

**Purpose –** Ensure that the ipady configuration works for a single value.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.28

### Test Procedure – Configure Ipady Multiple

**Purpose –** Ensure that the ipady configuration works for multiple values.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.29

### Test Procedure – Configure Padx Single

**Purpose –** Ensure that the padx configuration works for a single value.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.30

### Test Procedure – Configure Padx Multiple

**Purpose –** Ensure that the padx configuration works for multiple values.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.31

### Test Procedure – Configure Pady Single

**Purpose –** Ensure that the pady configuration works for a single value.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.32

### Test Procedure – Configure Pady Multiple

**Purpose –** Ensure that the pady configuration works for multiple values.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.33

### Test Procedure – Configure Position Right Margin

**Purpose –** Ensure that the position configuration works for rightmargin.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.34

### Test Procedure – Configure Position Left Margin

**Purpose –** Ensure that the position configuration works for leftmargin.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.35

### Test Procedure – Configure Position Top Margin

**Purpose –** Ensure that the position configuration works for topmargin.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.36

### Test Procedure – Configure Position Bottom Margin

**Purpose –** Ensure that the position configuration works for bottommargin.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.37

### Test Procedure – Configure Position Plot Area

**Purpose –** Ensure that the position configuration works for plotarea.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.38

### Test Procedure – Configure Position a Point

**Purpose –** Ensure that the position configuration works for a point.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.39

### Test Procedure – Configure Raised

**Purpose –** Ensure that the raised configuration works for true.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.40

### Test Procedure – Configure Not Raised

**Purpose –** Ensure that the raised configuration works for false.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.41

### Test Procedure – Configure Relief Raised

**Purpose –** Ensure that the relief configuration works for raised reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.42

### Test Procedure – Configure Relief Flat

**Purpose –** Ensure that the relief configuration works for flat reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.43

### Test Procedure – Configure Relief Grooved

**Purpose –** Ensure that the relief configuration works for grooved reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.44

### Test Procedure – Configure Relief Ridged

**Purpose –** Ensure that the relief configuration works for ridged reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.45

### Test Procedure – Configure Relief Solid

**Purpose –** Ensure that the relief configuration works for solid reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.46

### Test Procedure – Configure Relief Sunken

**Purpose –** Ensure that the relief configuration works for sunken reliefs.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.47

### Test Procedure – Configure Shadow

**Purpose –** Ensure that the shadow configuration works for valid colors.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.48

### Test Procedure – Configure Shadow and Depth

**Purpose –** Ensure that the shadow configuration works for a shadow and a depth.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.49

### Test Procedure – Configure No Shadow

**Purpose –** Ensure that the shadow configuration works for no shadow.

**Special Requirements –** None

**TclTest –** RBC.graph.legend.3.50

# Line

# Marker

## Bind

#### Test Case 13

### Test Procedure – Bind Create

**Purpose –** Ensure that bindings can be created for a marker

**Special Requirements –** None

**TclTest –** RBC.graph.marker.bind.1.1

### Test Procedure – Bind Query for Sequence and Marker

**Purpose –** Ensure that bindings can be queried for a sequence and marker

**Special Requirements –** None

**TclTest –** RBC.graph.marker.bind.1.2

### Test Procedure – Bind Command Append

**Purpose –** Ensure that bindings can be appended for a sequence and marker

**Special Requirements –** None

**TclTest –** RBC.graph.marker.bind.1.3

### Test Procedure – Bind Command Overwrite

**Purpose –** Ensure that bindings are overwritten for a sequence and marker

**Special Requirements –** None

**TclTest –** RBC.graph.marker.bind.1.4

### Test Procedure – Bind Query for Element

**Purpose –** Ensure that bound sequences can be queried for just a marker

**Special Requirements –** None

**TclTest –** RBC.graph.marker.bind.1.5

## Create

### Test Procedure – Graph Marker Exists

**Purpose –** Ensure creating a marker actually creates the marker on the graph.

**Special Requirements –** None

**TclTest –** RBC.graph.marker.1.1

## Destroy

### Test Procedure – Deletes a Single Marker

**Purpose –** Ensure the marker delete command works correctly when given a single existing marker name

**Special Requirements –** None

**TclTest –** RBC.graph.marker.2.1

### Test Procedure – Deletes multiple markers

**Purpose –** Ensure the marker delete command works correctly when given multiple existing marker name

**Special Requirements –** None

**TclTest –** RBC.graph.marker.2.2

## Exists

### Test Procedure – Non-existant marker

**Purpose –** Ensure that a marker that doesn't exist is recognized by exists

**Special Requirements –** None

**TclTest –** RBC.graph.marker.3.1

### Test Procedure – Existant marker

**Purpose –** Ensure that a marker that an existing marker is recognized by exists

**Special Requirements –** None

**TclTest –** RBC.graph.marker.3.2

## Type

### Test Procedure – Type of a Text Marker

**Purpose –** Ensure a text marker is of the correct type (TextMarker).

**Special Requirements –** None

**TclTest –** RBC.graph.marker.4.1

### Test Procedure – Type of a Line Marker

**Purpose –** Ensure a line marker is of the correct type (LineMarker).

**Special Requirements –** None

**TclTest –** RBC.graph.marker.4.2

### Test Procedure – Type of a Bitmap Marker

**Purpose –** Ensure a bitmap marker is of the correct type (BitmapMarker).

**Special Requirements –** None

**TclTest –** RBC.graph.marker.4.3

### Test Procedure – Type of a Image Marker

**Purpose –** Ensure a image marker is of the correct type (ImageMarker).

**Special Requirements –** None

**TclTest –** RBC.graph.marker.4.4

### Test Procedure – Type of a Polygon Marker

**Purpose –** Ensure a polygon marker is of the correct type (PolygonMarker).

**Special Requirements –** None

**TclTest –** RBC.graph.marker.4.5

### Test Procedure – Type of a Window Marker

**Purpose –** Ensure a window marker is of the correct type (WindowMarker).

**Special Requirements –** None

**TclTest –** RBC.graph.marker.4.6

## Names

### Test Procedure – Marker Names: No Pattern

**Purpose –** Ensure the marker names command works correctly when no pattern is given.

**Special Requirements –** None

**TclTest –** RBC.graph.marker.5.1

### Test Procedure – Marker Names: Exact Pattern

**Purpose –** Ensure the marker names command works correctly when an exact pattern is given.

**Special Requirements –** None

**TclTest –** RBC.graph.marker.5.2

### Test Procedure – Marker Names: Wildcard Pattern

**Purpose –** Ensure the marker names command works correctly when a wildcard pattern is given.

**Special Requirements –** None

**TclTest –** RBC.graph.marker.5.3

### Test Procedure – Marker Names: Incorrect Pattern

**Purpose –** Ensure the marker names command works correctly when an incorrect pattern is given.

**Special Requirements –** None

**TclTest –** RBC.graph.marker.5.4

## Configure

#### Test Case 6

### Test Procedure – Configure Coords

**Purpose –** Ensure configuring coords changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.6.1

### Test Procedure – Configure Element

**Purpose –** Ensure configuring element changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.6.2

### Test Procedure – Configure Hide

**Purpose –** Ensure configuring hide changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.6.3

### Test Procedure – Configure Under

**Purpose –** Ensure configuring under changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.6.4

### Test Procedure – Configure XOffset

**Purpose –** Ensure configuring xoffset changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.6.5

### Test Procedure – Configure YOffset

**Purpose –** Ensure configuring yoffset changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.6.6

### Test Procedure – Configure Mapx

**Purpose –** Ensure configuring mapx changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.6.7

### Test Procedure – Configure Mapy

**Purpose –** Ensure configuring mapy changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.6.8

#### Test Case 7

### Test Procedure – Configure Anchor

**Purpose –** Ensure configuring anchor changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.7.1

### Test Procedure – Configure Background

**Purpose –** Ensure configuring background changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.7.2

### Test Procedure – Configure Font

**Purpose –** Ensure configuring font changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.7.3

### Test Procedure – Configure Fill

**Purpose –** Ensure configuring fill changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.7.4

### Test Procedure – Configure Foreground

**Purpose –** Ensure configuring foreground changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.7.5

### Test Procedure – Configure Justify

**Purpose –** Ensure configuring justify changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.7.6

### Test Procedure – Configure Outline

**Purpose –** Ensure configuring outline changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.7.7

### Test Procedure – Configure Padx

**Purpose –** Ensure configuring padx changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.7.8

### Test Procedure – Configure Pady

**Purpose –** Ensure configuring pady changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.7.9

### Test Procedure – Configure Rotate

**Purpose –** Ensure configuring rotate changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.7.10

### Test Procedure – Configure Text

**Purpose –** Ensure configuring text changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.7.11

#### Test Case 8

### Test Procedure – Configure Dashes

**Purpose –** Ensure configuring dashes changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.8.1

### Test Procedure – Configure Fill

**Purpose –** Ensure configuring fill changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.8.2

### Test Procedure – Configure Linewidth

**Purpose –** Ensure configuring linewidth changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.8.3

### Test Procedure – Configure Outline

**Purpose –** Ensure configuring outline changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.8.4

### Test Procedure – Configure Xor

**Purpose –** Ensure configuring xor changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.8.5

#### Test Case 9

### Test Procedure – Configure Background

**Purpose –** Ensure configuring background changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.9.1

### Test Procedure – Configure Bitmap

**Purpose –** Ensure configuring bitmap changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.9.2

### Test Procedure – Configure Fill

**Purpose –** Ensure configuring fill changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.9.3

### Test Procedure – Configure Foreground

**Purpose –** Ensure configuring foreground changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.9.4

### Test Procedure – Configure Outline

**Purpose –** Ensure configuring outline changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.9.5

### Test Procedure – Configure Rotate

**Purpose –** Ensure configuring rotate changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.9.6

#### Test Case 10

### Test Procedure – Configure Anchor

**Purpose –** Ensure configuring Anchor changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.10.1

### Test Procedure – Configure Image

**Purpose –** Ensure configuring image changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.10.2

#### Test Case 11

### Test Procedure – Configure Dashes

**Purpose –** Ensure configuring dashes changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.11.1

### Test Procedure – Configure Fill

**Purpose –** Ensure configuring fill changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.11.2

### Test Procedure – Configure Linewidth

**Purpose –** Ensure configuring linewidth changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.11.3

### Test Procedure – Configure Outline

**Purpose –** Ensure configuring outline changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.11.4

### Test Procedure – Configure Stipple

**Purpose –** Ensure configuring stipple changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.11.5

#### Test Case 12

### Test Procedure – Configure Anchor

**Purpose –** Ensure configuring anchor changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.12.1

### Test Procedure – Configure Height

**Purpose –** Ensure configuring height changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.12.2

### Test Procedure – Configure Width

**Purpose –** Ensure configuring width changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.12.3

### Test Procedure – Configure Window

**Purpose –** Ensure configuring window changes the corresponding option value pair

**Special Requirements –** None

**TclTest –** RBC.graph.marker.12.4

# Pen

## Create

#### Test Case 1

### Test Procedure – Create

**Purpose –** Ensure that pen creation works when passed only a pen name.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.1.1

### Test Procedure – Create Non-overwriting

**Purpose –** Ensure that pen creation will not overwrite an existing pen name.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.1.2

### Test Procedure – Default Pens

**Purpose –** Ensure that two default pens are created automatically when a graph is created

**Special Requirements –** None

**TclTest –** RBC.graph.pen.1.3

### Test Procedure – Creation with Single Option

**Purpose –** Ensure that pen creation works when passed a single option-value pair.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.1.4

### Test Procedure – Creation with Multiple Options

**Purpose –** Ensure that pen creation works when passed option-value pairs.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.1.5

## Delete

#### Test Case 2

### Test Procedure – Deleting Single Pen

**Purpose –** Ensure that pen deletion works when deleting a single pen.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.2.1

### Test Procedure – Deleting Multiple Pens

**Purpose –** Ensure that pen deletion works when deleting multiple pens.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.2.2

## Configure

#### Test Case 3

### Test Procedure – Configure Color

**Purpose –** Ensure that the color configuration works for valid colors.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.1

### Test Procedure – Configure Dashes

**Purpose –** Ensure that the dashes configuration works for valid dashes.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.2

### Test Procedure – Configure No Dashes

**Purpose –** Ensure that the dashes configuration works for no dashes.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.3

### Test Procedure – Configure Fill Color

**Purpose –** Ensure that the fill configuration works for valid fill colors.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.4

### Test Procedure – Configure No Fill

**Purpose –** Ensure that the fill configuration works for no fill.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.5

### Test Procedure – Configure Line Width

**Purpose –** Ensure that the linewidth configuration works for valid linewidths.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.6

### Test Procedure – Configure Off Dash Color

**Purpose –** Ensure that the offdash configuration works for valid offdash colors.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.7

### Test Procedure – Configure Outline Color

**Purpose –** Ensure that the outline configuration works for valid outline colors.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.8

### Test Procedure – Configure Outline Default Color

**Purpose –** Ensure that the outline configuration works for defcolor (the same color as the color configuration option).

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.9

### Test Procedure – Configure Outline Width

**Purpose –** Ensure that the outlinewidth configuration works for valid widths.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.10

### Test Procedure – Configure No Outline

**Purpose –** Ensure that the outlinewidth configuration works for no width.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.11

### Test Procedure – Configure Symbol Size

**Purpose –** Ensure that the pixels configuration works for valid symbol sizes.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.12

### Test Procedure – Configure No Symbol (Size Zero)

**Purpose –** Ensure that the pixels configuration works for symbole size zero.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.13

### Test Procedure – Configure Square Symbol

**Purpose –** Ensure that the symbol configuration works for square.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.14

### Test Procedure – Configure Circle Symbol

**Purpose –** Ensure that the symbol configuration works for circle.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.15

### Test Procedure – Configure Diamond Symbol

**Purpose –** Ensure that the symbol configuration works for diamond.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.16

### Test Procedure – Configure Plus Symbol

**Purpose –** Ensure that the symbol configuration works for plus.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.17

### Test Procedure – Configure Cross Symbol

**Purpose –** Ensure that the symbol configuration works for cross.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.18

### Test Procedure – Configure Splus Symbol

**Purpose –** Ensure that the symbol configuration works for splus.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.19

### Test Procedure – Configure Scross Symbol

**Purpose –** Ensure that the symbol configuration works for scross.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.20

### Test Procedure – Configure Triangle Symbol

**Purpose –** Ensure that the symbol configuration works for triangle.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.21

### Test Procedure – Configure No Symbol

**Purpose –** Ensure that the symbol configuration works for no symbol.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.22

### Test Procedure – Configure Bitmap Symbol

**Purpose –** Ensure that the symbol configuration works for bitmap symbols.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.23

### Test Procedure – Configure Default Type

**Purpose –** Ensure that the default type of a pen is set.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.24

### Test Procedure – Configure Type

**Purpose –** Ensure that the type configuration works.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.3.25

## Cget

#### Test Case 5

### Test Procedure – Cget

**Purpose –** Ensure that cget works with an explicitly set option.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.5.1

### Test Procedure – Cget Default

**Purpose –** Ensure that cget works with a default value.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.5.2

## Delete

## Names

#### Test Case 4

### Test Procedure – Names No Pattern

**Purpose –** Ensure that names works with no given pattern.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.4.1

### Test Procedure – Names with Pattern

**Purpose –** Ensure that names works with a pattern parameter.

**Special Requirements –** None

**TclTest –** RBC.graph.pen.4.2

# Postscript

## Cget

#### Test Case 1

### Test Procedure – Cget

**Purpose –** Ensure that cget works for default values.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.1.1

### Test Procedure – Cget Default

**Purpose –** Ensure that cget works with an explicitly set option.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.1.2

## Configure

#### Test Case 2

### Test Procedure – Configure Center False

**Purpose –** Ensure that the center configuration works for valid booleans.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.1

### Test Procedure – Configure Center True

**Purpose –** Ensure that the center configuration works for valid booleans.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.2

### Test Procedure – Configure Colormode Color

**Purpose –** Ensure that the colormode configuration works for color mode.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.3

### Test Procedure – Configure Colormode Gray

**Purpose –** Ensure that the colormode configuration works for gray mode.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.4

### Test Procedure – Configure Colormode Greyscale

**Purpose –** Ensure that the colormode configuration works for grayscale mode.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.5

### Test Procedure – Configure Colormode Mono

**Purpose –** Ensure that the colormode configuration works for mono mode.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.6

### Test Procedure – Configure Colormode Monochrome

**Purpose –** Ensure that the colormode configuration works for monochrome mode.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.7

### Test Procedure – Configure Decorations False

**Purpose –** Ensure that the decorations configuration works.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.8

### Test Procedure – Configure Decorations True

**Purpose –** Ensure that the decorations configuration works.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.9

### Test Procedure – Configure Height Zero

**Purpose –** Ensure that the height configuration works for zero.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.10

### Test Procedure – Configure Height

**Purpose –** Ensure that the height configuration works for valid values.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.11

### Test Procedure – Configure Landscape False

**Purpose –** Ensure that the landscape configuration works.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.12

### Test Procedure – Configure Landscape True

**Purpose –** Ensure that the landscape configuration works.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.13

### Test Procedure – Configure Maxpect Zero

**Purpose –** Ensure that the maxpect configuration works.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.14

### Test Procedure – Configure Maxpect

**Purpose –** Ensure that the maxpect configuration works.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.15

### Test Procedure – Configure Padx Single Value

**Purpose –** Ensure that the padx configuration works for a single value.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.16

### Test Procedure – Configure Padx Multiple Values

**Purpose –** Ensure that the padx configuration works for multiple values.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.17

### Test Procedure – Configure Pady Single Value

**Purpose –** Ensure that the pady configuration works for a single value.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.18

### Test Procedure – Configure Pady Multiple Values

**Purpose –** Ensure that the pady configuration works for multiple values.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.19

### Test Procedure – Configure Paper Height

**Purpose –** Ensure that the paperheight configuration works for valid values.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.20

### Test Procedure – Configure Paper Width

**Purpose –** Ensure that the paperwidth configuration works for valid values.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.21

### Test Procedure – Configure Width Zero

**Purpose –** Ensure that the width configuration works for zero.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.22

### Test Procedure – Configure Width

**Purpose –** Ensure that the width configuration works for valid values.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.2.23

## Output

#### Test Case 3

### Test Procedure – Output to Console

**Purpose –** Ensure that output will print postscript to the console.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.3.1

### Test Procedure – Output to File

**Purpose –** Ensure that output will print postscript to a file.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.3.2

### Test Procedure – Output to Console with Options

**Purpose –** Ensure that output will print postscript to the console with option-value pairs.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.3.3

### Test Procedure – Output to File with Options

**Purpose –** Ensure that output will print postscript to a file with option-value pairs.

**Special Requirements –** None

**TclTest –** RBC.graph.postscript.3.4

# Snap

# Transform

## Test Case 1

**Test Case ID –** RBC.graph.crosshairs.1

**Test Item –** The *crosshairs cget* function of the *graph* BLT component.

**Input Specification –** The name of a crosshairs configure option.

**Output Specification –** The value of the given crosshairs configure option name.

**Special Procedural Requirements –** A graph widget can be created.

**Inter-case Dependencies –**

### Test Procedure – Graph Crosshairs: Cget - Valid Option Name

**Purpose –** Ensure the *crosshairs cget* command works correctly when given a valid crosshairs configuration option name.

**Special Requirements –** None

**TclTest –** RBC.graph.crosshairs.1.1

# Xaxis

# X2axis

# Yaxis

# Y2axis