

# Package ‘xmapbridge’

April 3, 2025

**Type** Package

**Title** Export plotting files to the xmapBridge for visualisation in X:Map

**Version** 1.65.0

**Date** 2008-12-09

**Author**

Tim Yates <Tim.Yates@cruk.manchester.ac.uk> and Crispin J Miller <Crispin.Miller@cruk.manchester.ac.uk>

**Maintainer** Chris Wirth <Christopher.Wirth@cruk.manchester.ac.uk>

**Depends** R (>= 2.0), methods

**Suggests** RUnit, RColorBrewer

**Description** xmapBridge can plot graphs in the X:Map genome browser. This package exports plotting files in a suitable format.

**License** LGPL-3

**URL** <http://xmap.picr.man.ac.uk>, <http://www.bioconductor.org>

**biocViews** Annotation, ReportWriting, Visualization

**Collate** zzz.R xmapbridge.R

**git\_url** <https://git.bioconductor.org/packages/xmapbridge>

**git\_branch** devel

**git\_last\_commit** 13dcfde

**git\_last\_commit\_date** 2024-10-29

**Repository** Bioconductor 3.21

**Date/Publication** 2025-04-02

## Contents

|                               |   |
|-------------------------------|---|
| xmapbridge-package . . . . .  | 2 |
| exon.data . . . . .           | 3 |
| xmap.col . . . . .            | 3 |
| xmap.files . . . . .          | 4 |
| xmap.plot . . . . .           | 6 |
| xmapbridge-internal . . . . . | 8 |

**Index****9**

---

xmapbridge-package      *Export plotting files to the xmapBridge for visualisation in X:Map*

---

**Description**

The XMapBridge is a Java application which allows you to plot graphs in the X:Map genome browser. This package exports plotting files in a suitable format for the XMapBridge to read.

**Details**

Package: xmapbridge  
Type: Package  
Version: 1.0  
Date: 2008-08-11  
License: LGPL-v3

**Author(s)**

Tim Yates and Crispin J Miller  
Maintainer: Tim Yates <tyates@picr.man.ac.uk>

**References**

Yates T, Okoniewski MJ, Miller CJ. X:Map: annotation and visualization of genome structure for Affymetrix exon array analysis. *Nucleic Acids Res.* 2008 Jan;36(Database issue):D780-6. Epub 2007 Oct 11.  
<http://nar.oxfordjournals.org/cgi/content/full/gkm779v1>

**See Also**

[xmap.plot](#)  
[xmap.points](#)  
[xmap.col](#)  
[xmap.project.new](#)  
[xmap.project.list](#)  
[xmap.project.load](#)  
[xmap.project.save](#)  
[xmap.project.delete](#)  
[xmap.graph.new](#)  
[xmap.graph.list](#)  
[xmap.graph.load](#)  
[xmap.graph.save](#)

```
xmap.graph.delete  
xmap.plot.new  
xmap.plot.list  
xmap.plot.load  
xmap.plot.save  
xmap.plot.delete
```

---

|           |                                  |
|-----------|----------------------------------|
| exon.data | <i>Sample exon array dataset</i> |
|-----------|----------------------------------|

---

**Description**

Data frame containing gene,exon and probeset IDs and probeset target locations for a subset of an Affymetrix exon array dataset comparing two cell lines (mcf7 and mcf10a).

**Usage**

```
data(xmapbridge)
```

**Details**

The object is a subset (80 genes, 1747 probesets) of RMA processed Exon Array data with associated annotation. It was generated using the exonmap bioconductor package. More details on the original dataset can be found in Okoniewski MJ, Miller CJ. Comprehensive Analysis of Affymetrix Exon Arrays Using BioConductor. PLoS Computational Biology 2008, 4(2): e6 doi:10.1371/journal.pcbi.0040006.

**Value**

On loading, creates the object exon.data.

**Author(s)**

Crispin J Miller

---

|          |   |
|----------|---|
| xmap.col | <i>Generate a colour integer for the xmapBridge</i> |
|----------|---|

---

**Description**

Convert an R colour into the form 0xAARRGGBB as used by the xmapBridge

**Usage**

```
xmap.col(col,alpha)
```

## Arguments

|       |  |
|-------|--|
| col   | The R colour to convert                              |
| alpha | Change the colour's alpha value to the one specified |

## Details

Takes one or more hex strings, possibly with an alpha value, as used by R and converts them to an integer of the form 0xAARRGGBB to be used by the xmapBridge. If alpha is specified, then change the colours' alpha values to this, otherwise leave them as they were.

## Author(s)

Crispin J Miller

## See Also

[xmap.plot](#)  
[xmap.plot.new](#)

## Examples

```
require( RColorBrewer )  
transparent.greens <- xmap.col( brewer.pal( 5, "Greens" ), alpha=0x22 )
```

---

|            |   |
|------------|---|
| xmap.files | <i>Underlying xmap methods for manipulating projects, graphs, and plots</i> |
|------------|---|

---

## Description

These methods are used to control xmap.projects, xmap.graphs and xmap.plots

A project contains one or more graphs, and a graph can contain one or more plots (each plot can have a different style)

A simpler method to generate X:Map graphs is to use the xmap.plot and xmap.points methods.

The xmap.debug method was introduced as otherwise the vignette would not print out id values as it was supposed to. For some reason Sweave ignores the show and as.character methods being overridden, and simply does its own thing.

## Usage

```
xmap.debug( idobject, newlines=FALSE )  
  
xmap.project.new( name )  
xmap.project.list()  
xmap.project.load( projectid )  
xmap.project.save( projectid, data )
```

```

xmap.project.delete( projectid )

xmap.graph.new( projectid, name, desc, min, max, chr, start, stop, ylab="value",
               species=c("homo_sapiens", "mus_musculus", "rattus_norvegicus") )
xmap.graph.list( projectid )
xmap.graph.load( graphid )
xmap.graph.save( graphid, data )
xmap.graph.delete( graphid )

xmap.plot.new( graphid, name, x, y,
              type=c("scatter", "line", "bar", "step", "area", "steparea"),
              col = NULL, dp = 2 )
xmap.plot.list( graphid )
xmap.plot.load( plotid )
xmap.plot.save( plotid, data, x, y, dp = 2 )
xmap.plot.delete( plotid )

```

## Arguments

|           |   |
|-----------|---|
| idobject  | An id object for an xmapbridge project, graph or plot   |
| newlines  | Should the resultant string be nicely formatyted with newline characters?   |
| name      | The name for this object (stored in the NAME variable of the list returned by load)   |
| projectid | The projectid (returned by xmap.project.new and as a list from xmap.project.list)   |
| data      | This is a list object for this particular object type   |
| desc      | The description for this item   |
| min       | The minimum y value for the graph   |
| max       | The maximum y value for the graph   |
| chr       | A string representing the chromosome this graph is to be drawn on   |
| start     | The BP that this graph should start on  |
| stop      | The BP that this graph reaches until. Must be greater than start  |
| ylab      | The label for the yaxis of this graph   |
| species   | The species that this graph is to be drawn on   |
| graphid   | The graphid returned by xmap.graph.new or xmap.graph.list   |
| type      | The type of graph you wish to draw  |
| col       | The colour of the graph defined as an integer in the format 0xAARRGGBB. See <a href="#">xmap.col</a>  |
| x         | The points for the x-axis of this plot. These points must be offsets from the start of the graph, and not their actual chromosomal location |
| y         | The points for the y-axis of this plot. Points where min>y or y >max will not be displayed  |
| dp        | The number of decimal places you want to save for the y-coordinates   |
| plotid    | The plotid which is returned by xmap.plot.new or xmap.plot.list   |

**Value**

xmap.graph.new returns the graphid of the created graph xmap.graph.list returns a vector of all graphs contained within the given project xmap.graph.load returns a list containing the parameters for the given graph xmap.graph.save

**Author(s)**

Tim Yates

**See Also**

[xmap.plot](#)  
[xmap.points](#)  
[xmap.col](#)  
<http://xmap.picr.man.ac.uk>

**Examples**

```
# Create a new project
projectid <- xmap.project.new( "A quick project" )

# Create a graph in this project covering 100K bases on chromosome 1
graphid <- xmap.graph.new( projectid, "A quick graph", "This is a quick graph",
                           -10, 10, "1", 5000, 105000, species="homo_sapiens" )

# Make our datasets
x <- seq( 1000, 100000, by = 1000 )
y <- runif( length( x ), -10, 10 )

# Then add a plot to this graph
xmap.plot.new( graphid, "Graph 1", x, y, type="area", col=0xAAFFEE00 )

# And another one, in a different colour
a <- xmap.plot.new( graphid, "Graph 2", x, y, type="scatter", col=0xAA0000FF )

cat( xmap.debug( a, newlines=TRUE ) )

# At this point, we should be able to see the graphs using the XMapBridge application

# Clean up our project file...
xmap.project.delete( projectid )
```

---

xmap.plot

*Utility methods for quick graph generation*

---

**Description**

These methods attempt to provide a simpler interface for generating X:Map graphs.

**Usage**

```
xmap.plot( x, y, species=c("homo_sapiens", "mus_musculus", "rattus_norvegicus"),
           chr, xlim, ylim, main, sub,
           type=c("scatter", "line", "bar", "step", "area", "steparea"),
           xlab, ylab = "value", col, dp=2, projectid )
xmap.points( x, y, type=c("scatter", "line", "bar", "step", "area", "steparea"),
             xlab, col, dp=2, graphid )
```

**Arguments**

|           |   |
|-----------|---|
| x         | The points for the x-axis of this plot. These points must be offsets from the start of the graph, and not their actual chromosomal location   |
| y         | The points for the y-axis of this plot. Points where min>y or y >max will not be displayed  |
| species   | The species that this graph is to be drawn on   |
| chr       | A string representing the chromosome this graph is to be drawn on   |
| xlim      | A range object describing the min and max BP for this graph. If omitted, we call range on the x parameter   |
| ylim      | The maximum and minimum Y value for the graph. If omitted, we call range on the y parameter   |
| main      | The main title  |
| sub       | A sub-title for this graph  |
| type      | The type of graph you wish to draw  |
| xlab      | The label for this plot   |
| ylab      | The label for the yaxis of this graph   |
| col       | The colour of the graph defined as an integer in the format 0xAARRGGBB. See <a href="#">xmap.col</a>  |
| dp        | The number of decimal places you want to save for the y-coordinates   |
| projectid | The projectid (returned by <a href="#">xmap.project.new</a> and as a list from <a href="#">xmap.project.list</a> . If this is not passed, on the first time of calling a new project will be created, and will be used on all subsequent calls) |
| graphid   | The graphid returned by <a href="#">xmap.graph.new</a> or <a href="#">xmap.graph.list</a> . If this is not passed, it will use the last graph generated by <a href="#">xmap.plot</a> .  |

**Author(s)**

Tim Yates

**See Also**

[xmap.col](#)  
[xmap.project.new](#)  
[xmap.project.list](#)  
[xmap.project.load](#)

```
xmap.project.save  
xmap.project.delete  
xmap.graph.new  
xmap.graph.list  
xmap.graph.load  
xmap.graph.save  
xmap.graph.delete  
xmap.plot.new  
xmap.plot.list  
xmap.plot.load  
xmap.plot.save  
xmap.plot.delete
```

### Examples

```
x <- seq( 1000, 100000, by = 1000 )  
y <- runif( length( x ), -10, 10 )  
  
# This will create a new project, and graph, and add a plot to it  
xmap.plot( x, y, species="homo_sapiens", "1", main="A quick graph",  
           sub="Sub heading", type="area", xlab="Plot 1", col=0xAAFFBB00 )  
  
# This will add another plot to the same graph  
xmap.points( x, y, type="area", xlab="Plot 2", col=0xAA0000FF )
```

---

xmapbridge-internal    *Internal xmapbridge objects*

---

### Description

Internal xmapbridge objects.

### Details

These are not to be called by the user.



# Index

!=, Graph, Graph-method (xmap.files), 4  
!=, Plot, Plot-method (xmap.files), 4  
!=, Project, Project-method (xmap.files),  
4

\* **internal**  
xmapbridge-internal, 8

\* **misc**  
exon.data, 3

\* **package**  
xmapbridge-package, 2

.get.xmap.cache.dir  
(xmapbridge-internal), 8

.make.xmap.child.file  
(xmapbridge-internal), 8

.make.xmap.project.folder  
(xmapbridge-internal), 8

.xmap.env (xmapbridge-internal), 8

.xmap.graph.valid  
(xmapbridge-internal), 8

.xmap.project.valid  
(xmapbridge-internal), 8

.xmap.read.file.info  
(xmapbridge-internal), 8

.xmap.write.file.info  
(xmapbridge-internal), 8

==, Graph, Graph-method (xmap.files), 4  
==, Plot, Plot-method (xmap.files), 4  
==, Project, Project-method (xmap.files),  
4

as.character, Graph-method (xmap.files),  
4

as.character, Plot-method (xmap.files), 4

as.character, Project-method  
(xmap.files), 4

exon.data, 3

xmap.col, 2, 3, 5–7

xmap.debug (xmap.files), 4

xmap.files, 4

xmap.graph.delete, 3, 8

xmap.graph.delete (xmap.files), 4

xmap.graph.list, 2, 7, 8

xmap.graph.list (xmap.files), 4

xmap.graph.load, 2, 8

xmap.graph.load (xmap.files), 4

xmap.graph.new, 2, 7, 8

xmap.graph.new (xmap.files), 4

xmap.graph.save, 2, 8

xmap.graph.save (xmap.files), 4

xmap.plot, 2, 4, 6, 6, 7

xmap.plot.delete, 3, 8

xmap.plot.delete (xmap.files), 4

xmap.plot.list, 3, 8

xmap.plot.list (xmap.files), 4

xmap.plot.load, 3, 8

xmap.plot.load (xmap.files), 4

xmap.plot.new, 3, 4, 8

xmap.plot.new (xmap.files), 4

xmap.plot.save, 3, 8

xmap.plot.save (xmap.files), 4

xmap.points, 2, 6

xmap.points (xmap.plot), 6

xmap.project.delete, 2, 8

xmap.project.delete (xmap.files), 4

xmap.project.list, 2, 7

xmap.project.list (xmap.files), 4

xmap.project.load, 2, 7

xmap.project.load (xmap.files), 4

xmap.project.new, 2, 7

xmap.project.new (xmap.files), 4

xmap.project.save, 2, 8

xmap.project.save (xmap.files), 4

xmapbridge (xmapbridge-package), 2

xmapbridge-internal, 8

xmapbridge-package, 2