

# Package ‘plan’

August 29, 2016

**Version** 0.4-2

**Date** 2013-09-29

**Title** Tools for project planning

**Author** Dan Kelley <Dan.Kelley@Dal.Ca>

**Maintainer** Dan Kelley <Dan.Kelley@Dal.Ca>

**Depends** R (>= 0.99)

**Description** Supports the creation of burndown charts and gantt diagrams.

**License** GPL (>= 2)

**URL** <http://github.com/dankelley/plan>

**LazyLoad** yes

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2013-09-30 20:15:17

## R topics documented:

|                            |    |
|----------------------------|----|
| as.gantt . . . . .         | 2  |
| burndown . . . . .         | 3  |
| gantt . . . . .            | 4  |
| plan . . . . .             | 4  |
| plot.burndown . . . . .    | 5  |
| plot.gantt . . . . .       | 6  |
| read.burndown . . . . .    | 8  |
| read.gantt . . . . .       | 10 |
| summary.burndown . . . . . | 12 |
| summary.gantt . . . . .    | 13 |
| trim.whitespace . . . . .  | 14 |

|              |           |
|--------------|-----------|
| <b>Index</b> | <b>15</b> |
|--------------|-----------|

---

`as.gantt`*Read a gantt data file*

---

### Description

Read a data file containing gantt information.

### Usage

```
as.gantt(key, description, start, end, done, neededBy)
```

### Arguments

|                          |   |
|--------------------------|---|
| <code>key</code>         | integer key for task, normally 1 for the first task, 2 for the second, etc.                               |
| <code>description</code> | character string describing the task (brief)  |
| <code>start</code>       | start date for task (POSIXt or character string that converts to POSIXt with <a href="#">as.POSIXct</a> ) |
| <code>end</code>         | end date for task (POSIXt or character string that converts to POSIXt with <a href="#">as.POSIXct</a> )   |
| <code>done</code>        | percentage completion for the task  |
| <code>neededBy</code>    | optional key for a dependent task   |

### Details

Creates a gantt object. See documentation for [read.gantt](#), which uses `as.gantt`.

### Value

An object of type "gantt"; for details, see [read.gantt](#).

### Author(s)

Dan Kelley

### See Also

[read.gantt](#), [summary.gantt](#) and [plot.gantt](#)

### Examples

```
library(plan)
arrive <- as.POSIXct("2012-09-05")
month <- 28 * 86400
year <- 12 * month
leave <- arrive + 4 * year
startT1 <- arrive
endT1 <- startT1 + 4 * month
```

```

startT2 <- endT1 + 1
endT2 <- startT2 + 4 * month
startT3 <- arrive + 12 * month
endT3 <- startT3 + 4 * month
startQE <- arrive + 9 * month
endQE <- arrive + 12 * month
QEabsoluteEnd <- arrive + 15 * month
startProposal <- arrive + 15 * month # for example
endProposal <- arrive + 20 * month
startThesisWork <- arrive + 2 * month # assumes no thesis work until 2 months in
endThesisWork <- leave - 4 * month
startThesisWriteup <- leave - 4 * month
endThesisWriteup <- leave
g <- as.gantt(key=1:7, c("Term 1 classes",
  "Term 2 classes",
  "Qualifying Examination",
  "Term 3 classes",
  "Proposal Defence",
  "Thesis Work",
  "Thesis Writing/Defence"),
  c(startT1, startT2, startQE, startT3, startProposal,
    startThesisWork, startThesisWriteup),
  c(endT1, endT2, endQE, endT3, endProposal,
    endThesisWork, endThesisWriteup),
  done=rep(0, 7))
plot(g, xlim=c(arrive, leave))

```

---

burndown

*Sample burndown dataset*


---

### Description

This is sample burndown dataset provided for testing.

### Usage

```
data(burndown)
```

### Format

See [read.burndown](#).

### Author(s)

Dan Kelley

### See Also

[read.burndown](#) may be used to read such objects, [summary.burndown](#) to summarize them, and [plot.burndown](#) to plot them.

---

|       |                             |
|-------|-----------------------------|
| gantt | <i>Sample gantt dataset</i> |
|-------|-----------------------------|

---

**Description**

This is sample gantt dataset provided for testing.

**Usage**

```
data(gantt)
```

**Format**

See [read.gantt](#)

**Author(s)**

Dan Kelley

**See Also**

[read.gantt](#) may be used to read such objects, [summary.gantt](#) to summarize them, and [plot.gantt](#) to plot them.

---

|      |   |
|------|---|
| plan | <i>Plan, a package for project planning</i> |
|------|---|

---

**Description**

This package provides tools for project planning, e.g. burndown charts, gantt diagrams, etc.

**Author(s)**

Dan Kelley

---

plot.burndown                      *Draw a burndown chart*

---

### Description

Plot a burndown object.

### Usage

```
## S3 method for class 'burndown'  
plot(x, col=NULL,  
draw.plan=TRUE, draw.regression=TRUE, draw.lastupdate=FALSE,  
t.stop="",  
y.name="Remaining Effort",  
debug=FALSE, ...)
```

### Arguments

|                 |   |
|-----------------|---|
| x               | an object of class burndown, to be plotted.   |
| col             | list of colours for items, starting with the first key in the file (which will be at the bottom of the chart). If not specified, the <a href="#">hcl</a> scheme will be used, to generate colours that are distinct, that show up reasonably well on a monitor. |
| draw.plan       | boolean, set to TRUE to draw the plan, as a descending line with a horizontal intercept.  |
| draw.regression | boolean, set to TRUE to draw a regression line of actual overall progress.  |
| draw.lastupdate | boolean, set to TRUE to draw the last update (which otherwise requires a sharp eye).  |
| t.stop          | a POSIX time, the maximum time for graph (defaults to deadline if not given).   |
| y.name          | character string, for labelling the vertical axis.  |
| debug           | boolean, set to TRUE to monitor the work.   |
| ...             | extra things handed down to plotting functions.   |

### Details

Plots a burndown chart.

### Value

The object, returned invisibly.

### Author(s)

Dan Kelley

**References**

<http://alistair.cockburn.us/crystal/articles/evabc/earnedvalueandburncharts.htm>.

**See Also**

[read.burndown](#) and [summary.burndown](#).

**Examples**

```
library(plan)
data(burndown)
summary(burndown)
plot(burndown)
```

---

plot.gantt

*Draw a Gantt diagram*

---

**Description**

Plot a Gantt object.

**Usage**

```
## S3 method for class 'gantt'
plot(x,
      xlim,
      time.format = NULL,
      time.labels.by, time.lines.by,
      event.time = NULL, event.label = NULL, event.side=3,
      col.done = gray(0.3), col.notdone = gray(0.9),
      col.event = gray(0.1), col.connector = "black",
      grid.col = "lightgray", grid.lty = "dotted",
      main = "",
      cex=par("cex"),
      debug=FALSE,
      ...)
```

**Arguments**

|                |  |
|----------------|--|
| x              | an object of class gantt.  |
| xlim           | optional range of time axis; if not provided, the range of times in x will be used.  |
| time.format    | format for dates on time axis; defaults to 3-letter month.   |
| time.labels.by | suggested label increment on time axis, e.g. time.labels.by="2 months" to get a two-month interval. If not supplied, the axis will be generated automatically. |

|                            |   |
|----------------------------|---|
| <code>time.lines.by</code> | suggested interval between vertical grid lines on the plot, e.g. <code>time.lines.by="1 week"</code> for weekly. If not supplied, the grid will be generated automatically. |
| <code>event.time</code>    | List of events, e.g. conferences, whose time cannot be altered.   |
| <code>event.label</code>   | list of names of these events.  |
| <code>event.side</code>    | side for event labels.  |
| <code>col.done</code>      | colour of work that has been done already.  |
| <code>col.notdone</code>   | colour of work that has not been done already.  |
| <code>col.event</code>     | colour of events.   |
| <code>col.connector</code> | colour of (optional) connectors between items.  |
| <code>grid.col</code>      | colour for grid   |
| <code>grid.lty</code>      | line type for grid  |
| <code>main</code>          | character string to be used as chart title.   |
| <code>cex</code>           | numeric, font-size factor.  |
| <code>debug</code>         | boolean, set to TRUE to monitor the work.   |
| <code>...</code>           | extra things handed down.   |

**Details**

Plots a gantt chart, possibly with events superimposed.

**Value**

The gantt object, returned invisibly.

**Note**

The defaults work well for projects that take a year or two. Consider adjusting `time.labels.by` and `time.lines.by` for projects that are much shorter or longer.

**Author(s)**

Dan Kelley

**References**

Gantt diagrams are described on wikipedia [http://en.wikipedia.org/wiki/Gantt\\_Chart](http://en.wikipedia.org/wiki/Gantt_Chart).

**See Also**

Use [read.gantt](#) to read gantt data, and [summary.gantt](#) to summarize them.

## Examples

```
library(plan)
data(gantt)
summary(gantt)
plot(gantt)
# Add a couple of event
event.label <- c("Proposal", "AGU")
event.time <- c("2008-01-28", "2008-12-10")
plot(gantt, event.label=event.label, event.time=event.time)
```

---

|               |                                |
|---------------|--------------------------------|
| read.burndown | <i>Scan burndown data file</i> |
|---------------|--------------------------------|

---

## Description

Read a data file containing burndown information.

## Usage

```
read.burndown(file, debug=FALSE)
```

## Arguments

|       |   |
|-------|---|
| file  | a connection or a character string giving the name of the file to load. |
| debug | boolean, set to TRUE to print debugging information.                    |

## Details

Reads a burndown dataset.

A strict format is required, in which the following items must be present, in the stated order, and with nothing else in the file. An example is given after the description.

- Line 1: contains two comma-separated items: the string `Start`, and a time expressed in ISO 8601 format (YYYY-MM-DD or YYYY-MM-DD hh:mm:ss). This line indicates the start of the project.
- Line 2: as Line 1, but the string is to be `End`, and the line indicates the deadline for the project.
- Line 3: a header line for a "task" list, comprising the following three words separated by commas: `Key`, `Description`, and `Effort`.
- Lines 4 to N: data lines, each containing three items: a numeric index "`Key`" for the task, a short "`Description`" of the task, and the estimated "`Effort`" for this task, expressed as a number. The keys must be distinct, and they must match the keys in the progress table (see below). The description should be short enough to give a reasonable-size legend as created by [plot.burndown](#). The effort may be expressed in any convenient unit, e.g. the number of hours or days for the task, or as a percentage of the overall task.



- Line N+1: a header line for the "Progress" list, comprising the following four words separated by commas: Key, Done, and Time.
- Line N+2 to end: data lines holding Progress items. Each "Key" must match a key in the task list. The "Done" column holds the percentage of the task that has been completed. The "Time" is in ISO 8601 format, as described above.

Executing the code

```
library(plan)
data(burndown)
print(summary(burndown))
```

will create the following, which may be read with read.burndown:

```
Start,      2006-04-08 12:00:00
Deadline, 2006-04-11 20:00:00
Key, Description,          Effort
  1, code read.burndown(),    4
  2, code summary.burndown(), 1
  3, code plot.burndown(),   5
  4, create R package,       2
  5, write documentation,    2
  6, set up website,         1
Key, Done,  Time
  1,   5, 2006-04-08 13:00:00
  2,   5, 2006-04-08 13:30:00
  1,  10, 2006-04-08 14:00:00
  2,  50, 2006-04-08 15:00:00
  4,   5, 2006-04-08 19:30:00
  5,   5, 2006-04-08 20:00:00
  4, 100, 2006-04-08 21:16:00
  1,  50, 2006-04-09 09:10:00
  3,   5, 2006-04-09 09:41:00
  3,  30, 2006-04-09 10:18:00
  3,  80, 2006-04-09 11:00:00
  2,  60, 2006-04-09 12:00:00
  2, 100, 2006-04-09 12:10:00
  1,  70, 2006-04-09 12:30:00
  5,  30, 2006-04-09 13:50:00
  5,  90, 2006-04-09 14:20:00
  5, 100, 2006-04-09 14:30:00
  1, 100, 2006-04-09 14:35:00
  3, 100, 2006-04-09 14:40:00
  6, 100, 2006-04-09 16:00:00
```

### Value

A burndown object.

**Author(s)**

Dan Kelley

**See Also**

[summary.burndown](#) and [plot.burndown](#).

**Examples**

```
## Not run:
library(plan)
b <- read.burndown("burndown.dat")
summary(b)
plot(b)

## End(Not run)
```

---

read.gantt

*Read a gantt data file*

---

**Description**

Read a data file containing gantt information.

**Usage**

```
read.gantt(file, debug=FALSE)
```

**Arguments**

`file` a connection or a character string giving the name of the file to load.  
`debug` boolean, set to TRUE to print debugging information.

**Details**

Reads a gantt dataset.

The data format is strict, and deviations from it may lead to error messages that are difficult to understand.

The first line is a header, and must contain the words `Key`, `Description`, `Start`, `End`, `Done`, and `NeededBy`, written exactly in this way, with commas separating the words. (Blanks are ignored in this line.)

Additional lines indicate the details of each of several sub-projects, in comma-separated items, as follows:

- A key for the task. These must be distinct, and are typically just the numbers 1, 2, 3, etc.
- A description of the task. (This may not contain commas!)

- The start time for the task, in ISO 8601 format (YYYY-MM-DD or YYYY-MM-DD hh:mm:ss).
- The end time for the task, in the same format as the starting time.
- A number indicating the percentage of this task that has been completed to date.
- A space-separated optional list of numbers that indicate the keys of other tasks that depend on this one. This list is ignored in the present version of read.gantt.

Executing the code

```
library(plan)
data(gantt)
print(summary(gantt))
```

will create the following sample file, which may be read with [read.gantt](#):

| Key, | Description,            | Start,      | End,        | Done, | NeededBy |
|------|-------------------------|-------------|-------------|-------|----------|
| 1,   | Assemble equipment,     | 2008-01-01, | 2008-03-28, | 90    |          |
| 2,   | Test methods,           | 2008-02-28, | 2008-03-28, | 30    |          |
| 3,   | Field sampling,         | 2008-04-01, | 2008-08-14, | 0     |          |
| 4,   | Analyse field data,     | 2008-06-30, | 2008-11-14, | 0     |          |
| 5,   | Write methods chapter,  | 2008-08-14, | 2008-11-14, | 0     |          |
| 6,   | Write results chapter,  | 2008-10-14, | 2009-01-15, | 0     |          |
| 7,   | Write other chapters,   | 2008-12-10, | 2009-02-28, | 0     |          |
| 8,   | Committee reads thesis, | 2009-02-28, | 2009-03-14, | 0     |          |
| 9,   | Revise thesis,          | 2009-03-15, | 2009-03-30, | 0     |          |
| 10,  | Thesis on display,      | 2009-04-01, | 2009-04-15, | 0     |          |
| 11,  | Defend thesis,          | 2009-04-16, | 2009-04-17, | 0     |          |
| 12,  | Finalize thesis,        | 2009-04-18, | 2009-05-07, | 0     |          |

### Value

An object of type "gantt", which is a data frame containing "description" (a character description of the task) "start" (the task's start time), "end" (the task's end time), "progress" (a number giving the percent progress on this item, or NA if none given), and "needed.by" (a number giving the indices of other tasks that rely on this task, or NA if none given).

### Author(s)

Dan Kelley

### See Also

[summary.gantt](#) and [plot.gantt](#)

### Examples

```
## Not run:
library(plan)
gantt <- read.gantt("demo/gantt.dat")
summary(gantt)
```

```
plot(gantt)

## End(Not run)
```

---

|                  |                                    |
|------------------|------------------------------------|
| summary.burndown | <i>Summarize a burndown object</i> |
|------------------|------------------------------------|

---

## Description

Summarizes a burndown object.

## Usage

```
## S3 method for class 'burndown'
summary(object, ...)
## S3 method for class 'summary.burndown'
print(x, ...)
```

## Arguments

|        |  |
|--------|--|
| object | an object of class burndown, e.g. as read by <a href="#">read.burndown</a> . |
| x      | an object of class summary.burndown, as created by summary.burndown.         |
| ...    | extra arguments (not used in this version).                                  |

## Details

Prints a summary of a burndown dataset.

## Value

None.

## Author(s)

Dan Kelley

## See Also

The burndown object may be read with [read.burndown](#) and plotted with [plot.burndown](#).

## Examples

```
library(plan)
data(burndown)
summary(burndown)
```

---

|               |                                 |
|---------------|---------------------------------|
| summary.gantt | <i>Summarize a gantt object</i> |
|---------------|---------------------------------|

---

### Description

Summarizes a gantt object.

### Usage

```
## S3 method for class 'gantt'  
summary(object, ...)  
## S3 method for class 'summary.gantt'  
print(x, ...)
```

### Arguments

|        |  |
|--------|--|
| object | an object of class gantt, e.g. as read by <a href="#">read.gantt</a> . |
| x      | an object of class summary.gantt, as created by summary.gantt.         |
| ...    | extra arguments (not used in this version).                            |

### Details

Prints a summary of a gantt dataset.

### Value

None.

### Author(s)

Dan Kelley

### References

<http://alistair.cockburn.us/crystal/articles/evabc/earnedvalueandburncharts.htm>.

### See Also

The gantt object may be read with [read.gantt](#).

### Examples

```
library(plan)  
data(gantt)  
summary(gantt)
```

---

|                 |  |
|-----------------|--|
| trim.whitespace | <i>Trim leading/trailing whitespace from character strings</i> |
|-----------------|--|

---

**Description**

Trim leading and trailing whitespace from character strings. Used by [read.gantt](#) and [read.burndown](#).

**Usage**

```
trim.whitespace(x)
```

**Arguments**

x a character string, or vector of character strings.

**Value**

As x, but with leading and trailing space removed

**Author(s)**

Dan Kelley

**Examples**

```
library(plan)
x <- c(" hellow there", "ba bye  ", " buddy  ")
print(x)
print(trim.whitespace(x))
```

# Index

## \*Topic **datasets**

burndown, [3](#)

gantt, [4](#)

## \*Topic **misc**

as.gantt, [2](#)

plan, [4](#)

plot.burndown, [5](#)

plot.gantt, [6](#)

read.burndown, [8](#)

read.gantt, [10](#)

summary.burndown, [12](#)

summary.gantt, [13](#)

trim.whitespace, [14](#)

as.gantt, [2](#)

as.POSIXct, [2](#)

burndown, [3](#)

gantt, [4](#)

hcl, [5](#)

plan, [4](#)

plot.burndown, [3](#), [5](#), [8](#), [10](#), [12](#)

plot.gantt, [2](#), [4](#), [6](#), [11](#)

print.summary.burndown

(summary.burndown), [12](#)

print.summary.gantt(summary.gantt), [13](#)

read.burndown, [3](#), [6](#), [8](#), [12](#), [14](#)

read.gantt, [2](#), [4](#), [7](#), [10](#), [11](#), [13](#), [14](#)

summary.burndown, [3](#), [6](#), [10](#), [12](#)

summary.gantt, [2](#), [4](#), [7](#), [11](#), [13](#)

trim.whitespace, [14](#)