

# pbdPROF

October 16, 2013

---

pbdPROF-package

*MPI Profiling Tools*

---

## Description

This package contains several libraries for profiling MPI codes, as well as some R tools for parsing, analyzing, and plotting their outputs.

## Details

Package:	pbdPROF
Type:	Package
License:	MPL
LazyLoad:	yes

This package requires an MPI library (OpenMPI, MPICH2, or LAM/MPI).

## Author(s)

Wei-Chen Chen, Drew Schmidt, Gaurav Sehrawat, Pragneshkumar Patel, and George Ostrouchov.

## References

Programming with Big Data in R Website: <http://r-pbd.org/>

## Examples

```
## Not run:  
demo(allreduce.fpmapi, "pbdPROF")  
demo(svd.fpmapi, "pbdPROF")  
demo(masterslavePI.fpmapi, "pbdPROF")  
demo(allreduce.mpip, "pbdPROF")  
demo(svd.mpip, "pbdPROF")
```

```
demo(masterslavePI.mpip, "pbdPROF")

## End(Not run)
```

---

Profilers

*Profilers*


---

## Description

Profilers Directly Supported by pbdPROF

## Details

Currently, the **fpmpi** library is fully supported; a version of this library is bundled with the source of the pbdPROF package (see package vignette for more details). Additional libraries can easily be linked with pbdPROF, but these are not yet fully supported. The **mpiP** and **TAU** profilers are expected to be fully supported by the conclusion of Google Summer of Code (~September 30 2013).

## References

Programming with Big Data in R Website: <http://r-pbd.org/>

**fpmpi** website: <http://www.mcs.anl.gov/research/projects/fpmipi/WWW/>

**mpiP** website: <http://mpip.sourceforge.net/>

**TAU** website: <http://www.cs.uoregon.edu/research/tau/home.php>

---

prof-class

*Class prof*


---

## Description

Class for Profiler Output

## Slots

**profiler:** the profiler used stored as character data.

**raw:** raw profiler output (read in via readLines()) cast as the virtual class rawprof.

**parsed:** a dataframe containing the parsed version of the raw slot.

## Details

The `prof` class is a simple container for managing profiler output via R.

The slots are `profiler`, `raw`, and `parsed`. The first, `profiler`, is just a character string consisting of the profiler used to generate the output (e.g., "fpmpi", "mpip", etc.).

Next, the slot `raw` contains the raw output from the profiling library, stored in R as a vector of character data (strings) directly from a `readLines()` call, and cast as the virtual class `rawprof`. In reality, we will cast the raw data as a virtual class of the same name as that found in the `profiler` slot, and `rawprof` is a superclass of each of these classes. This is for internal S3 dispatch. You should not mess around with this.

The final slot, `parsed`, contains a condensed version of the information from the `raw` slot, stored as a dataframe. This is the representation used for plotting.

Unless you really think you know what you are doing, you probably shouldn't directly mess around with the elements of this class. Instead, simply use the `read.prof()` function.

## See Also

[read.prof](#)

---

Example datasets

*Small example datasets*

---

## Description

Sample profiler outputs for testing and package demonstration.

## Format

Each dataset contains information profiled by **fpmpi** for `pbdMPI/demo/allreduce.fpmpi.r`, `pbdDMAT/demo/svd.fpmpi.r`, and `Rmpi/demo/masterslavePI.fpmpi.r`.

Each dataset contains information profiled by **mpip** for `pbdMPI/demo/allreduce.mpip.r`, `pbdDMAT/demo/svd.mpip.r`, and `Rmpi/demo/masterslavePI.mpip.r`.

## Details

Several sample profiler outputs are provided with the package, and utilized in the package demos to show off the functionality of the parsing and plotting methods provided by the package. These small datasets are located in the `pbdPROF/extdata/` folder of the installed package (and `pbdPROF/inst/extdata/` in the package source).

The example datasets are profiled in 2 processors.

---

`read.prof`*Reading Profiling Outputs*

---

**Description**

Reader for profiler outputs.

**Usage**

```
read.prof(file.name, ...)
```

**Arguments**

<code>file.name</code>	a full file name.
<code>...</code>	options for <code>readLines</code>

**Details**

This function reads in profiling outputs from MPI-using R code and stores the output in a `prof` class object. The reading is managed by the `base::readLines()` function. The user does not need to specify the type of profiler output being used (e.g., whether the profiler text is from **fpm**, **mpiP**, etc.).

Additionally, this method automatically parses the output into a condensed, manageable dataframe (the `parsed` slot of the `prof` class).

**Value**

A `prof` class object.

**See Also**

[prof-class](#)

**Examples**

```
## Not run:
library(pbdPROF)

fn <- system.file("data/allreduce.fpm", package = "pbdPROF")
da <- read.prof(fn, lib.type = "fpm")

da

## End(Not run)
```

---

**Print***Printing*

---

**Description**

Print and show methods for prof class objects.

**Usage**

```
## S4 method for signature prof
print(x, ...)
## S4 method for signature prof
show(object)
```

**Arguments**

x, object	prof class object
...	extra arguments

**Methods**

```
signature(x = "prof")
```

**See Also**

[prof-class](#), [read.prof](#)

---

**Plot***Plotting*

---

**Description**

Plot methods for prof class objects.

**Usage**

```
## S4 method for signature prof
plot(x, ..., which=1L:4L, show.title=TRUE,
      plot.type="timing", label, bar.label=FALSE)
## S3 method for class prof
autoplot(object, ...)
```

**Arguments**

<code>x</code>	prof class object
<code>object</code>	prof class object
<code>...</code>	extra arguments
<code>which</code>	Vector consisting of a subset of the integers 1, 2, 3, 4. Determines which plots will be produced.
<code>show.title</code>	Logical; determines whether or not the plot title will be shown.
<code>plot.type</code>	for profiling mpiP including "timing", "stats1", "stat2", "messages1", "message2", and "counts".
<code>label</code>	The label for the plot title.
<code>bar.label</code>	show labeled legend.

**Methods**

`signature(x = "prof")`

**See Also**

[prof-class](#), [read.prof](#)

# Index

## \*Topic **Classes**

prof-class, [2](#)

## \*Topic **Methods**

Plot, [5](#)

Print, [5](#)

## \*Topic **Package**

pbDPROF-package, [1](#)

## \*Topic **data**

Example datasets, [3](#)

## \*Topic **utility**

read prof, [4](#)

allreduce.fpmPI (Example datasets), [3](#)

allreduce.mPI (Example datasets), [3](#)

autoplot.prof (Plot), [5](#)

Example datasets, [3](#)

fpmPI-class (prof-class), [2](#)

fpmPI\_example (Example datasets), [3](#)

masterslavePI.fpmPI (Example datasets),  
[3](#)

masterslavePI.mPI (Example datasets), [3](#)

mPI-class (prof-class), [2](#)

mPI\_example (Example datasets), [3](#)

pbDPROF (pbDPROF-package), [1](#)

pbDPROF-package, [1](#)

Plot, [5](#)

plot (Plot), [5](#)

plot, prof-method (Plot), [5](#)

plot-method (Plot), [5](#)

Print, [5](#)

print (Print), [5](#)

print, prof-method (Print), [5](#)

print-method (Print), [5](#)

prof (prof-class), [2](#)

prof-class, [2](#)

Profilers, [2](#)

rawprof-class (prof-class), [2](#)

read prof, [4](#)

read.prof, [3](#), [5](#), [6](#)

read.prof (read prof), [4](#)

show (Print), [5](#)

show, prof-method (Print), [5](#)

show-method (Print), [5](#)

svd.fpmPI (Example datasets), [3](#)

svd.mPI (Example datasets), [3](#)

tau-class (prof-class), [2](#)