Roxygen – A Documentation System for R

Part I: Introduction

Manuel J. A. Eugster

Institut für Statistik
Ludwig-Maximiliams-Universität München

"Directions in Statistical Computing", Copenhagen, 2009

Donald Knuth proposed in "*Literate Programming*" (1984) the combination of a programming language and a documentation language:

> *from "instructing a computer what to do" to "explaining a human beeing what we want a computer to do"*

Literate programming: interleaving code and documentation chunks with `weave` and `tangle`; e.g. Sweave.

Interface documentation: documentation statements as comments; e.g. Doxygen for C/C++ and Javadoc for Java.

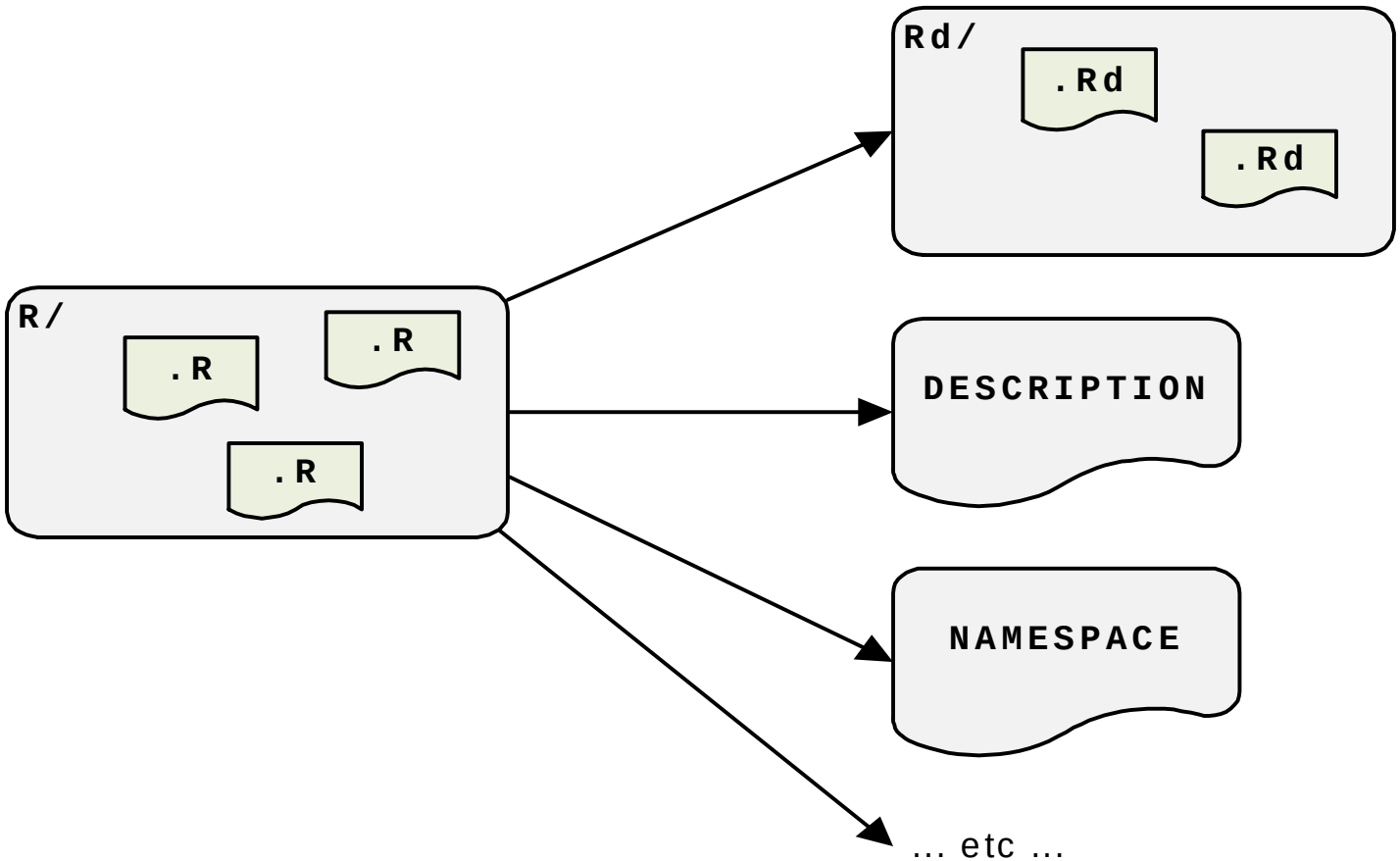
Roxygen enables **in-source** specification of

- documentation and
- package related information.

Google Summer of Code 2008
project by Peter Danenberg,
mentored by Manuel J. A. Eugster.

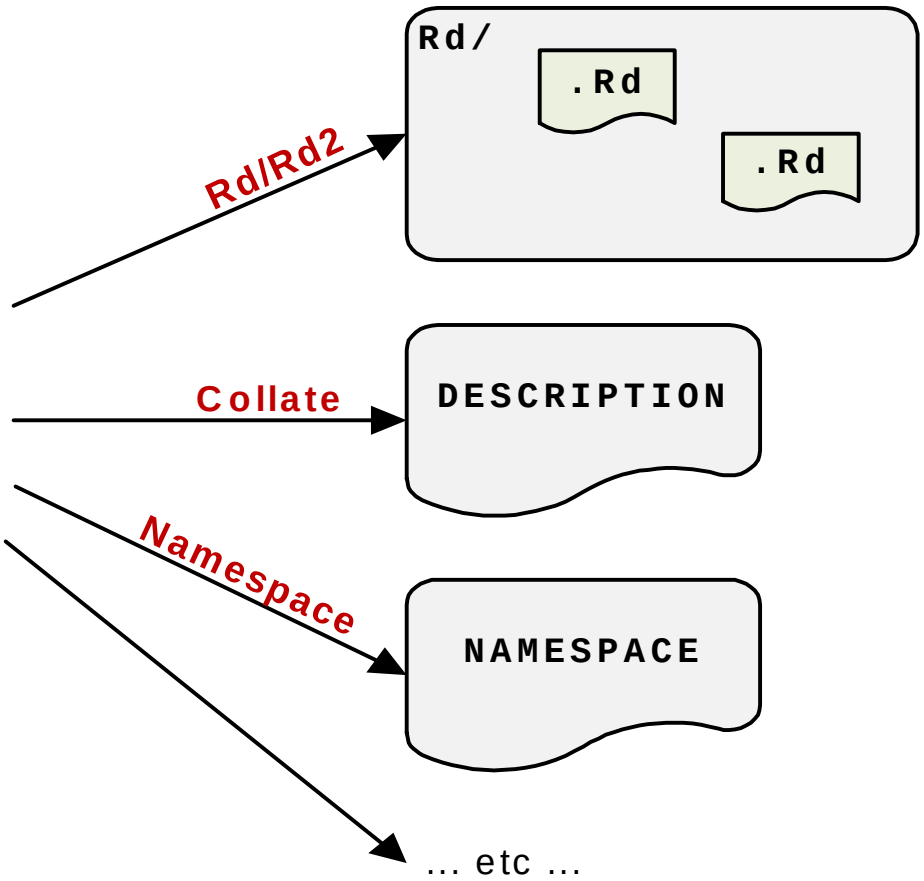
Documentation block in front of "some `R` statement" consisting of textual description and descriptive **tags**.

```
#' Description
#'
#' Details
#'
#' @param a Description
#' @param b Description
#' ...
f <- function(a, b) {
```


Rd/
.Rd
.Rd
R/
.R
.R
.R
DESCRIPTION
NAMESPACE
... etc ...

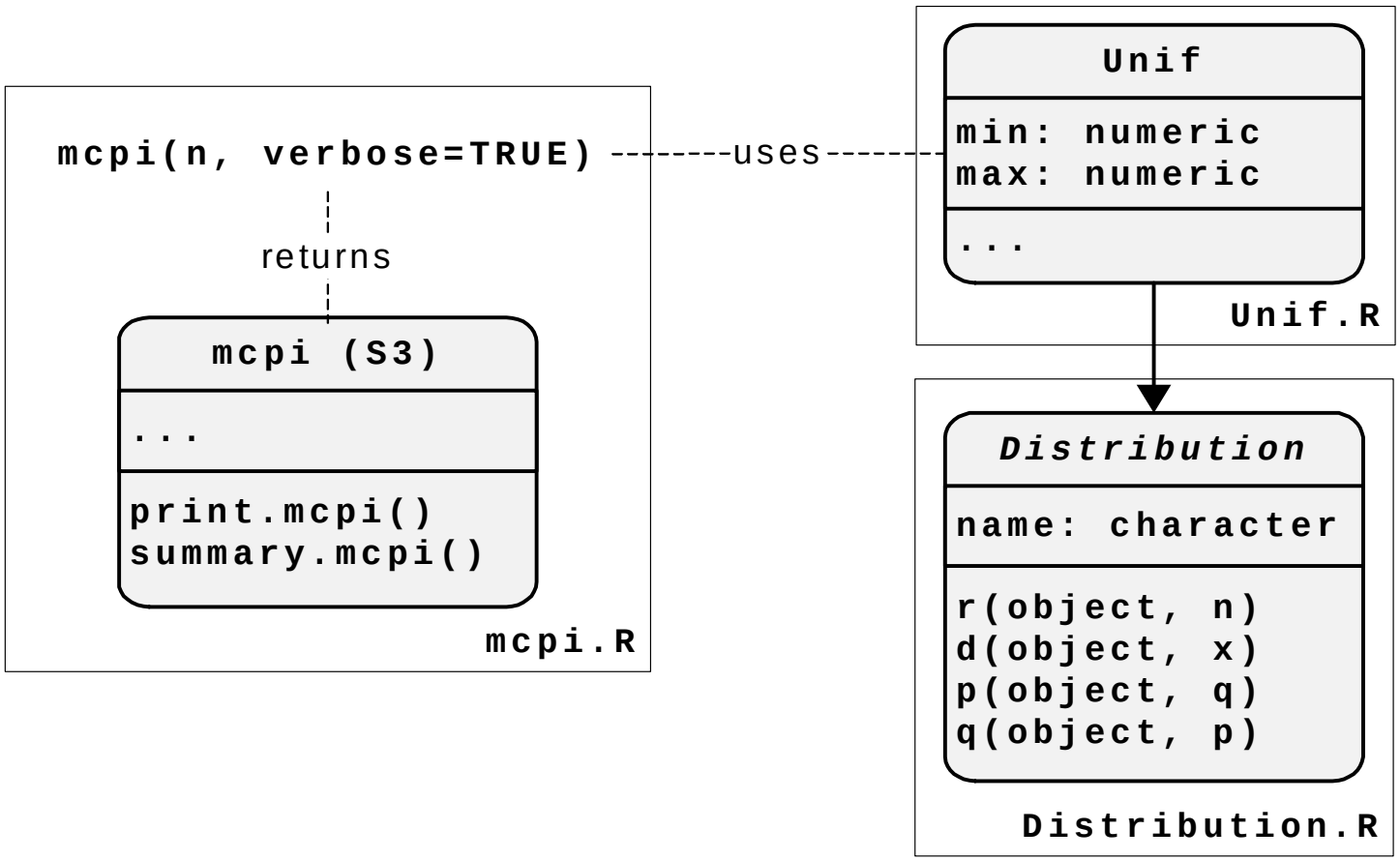
Roclets understand a set of tags and process them to some outcome.

```
#' Description
#'
#' Details
#'
#' @param a Description
#' @param b Description
#' ...
f <- function(a, b) {
```



A sample package

Monte-Carlo π approximation



In-source documentation

The Rd and Rd2 roclets

The **Rd** roclet is the original implementation supporting functions and S3 constructs.

The **Rd2** roclet is a new implementation using the `Rd` structure defined by `parse_Rd()` (`R` $\geq$ 2.9). It additionally supports basic S4 constructs documentation.

```
> rd <- make.Rd2.roclet(subdir='man')
> do.call(rd$parse, list.files('R/'))
```

```
#' Monte-Carlo PI approximation.
#'
#' If a circle of radius \eqn{R} is inscribed inside ...
#'
#' @param n Number of trials
#' @param verbose Print information during execution
#' @return S3 \code{mcpi} object; a list consisting of
#'   \item{pi}{the approximation of pi}
#'   \item{n}{the number of trials}
#'   \item{hits}{the number of hits}
#' @example mcpi/sandbox/mcpi.R
#' @references \url{http://www.eveandersson.com/pi/...}
#' ...
mcpi <- function(n, verbose=FALSE) {
```

?mcpi

```
#' @param x A \code{mcpi} object
#' @param ... Ignored
#' @method print mcpi
#' @rdname mcpi
#' ...
print.mcpi <- function(x, ...) {

#' @param object A \code{mcpi} object
#' @param ... Ignored
#' @method summary mcpi
#' @rdname mcpi
#' ...
summary.mcpi <- function(object, ...) {
```

?mcpi, ?print.mcpi, ?summary.mcpi

```
#' @title The uniform distribution.
#' @slot min Lower limit of the distribution
#' @slot max Upper limit of the distribution
#' ...
setClass('Unif',
         contains=c('Distribution'),
         representation=representation(
           min='numeric',
           max='numeric'),
         prototype=prototype(
           name='Uniform distribution',
           min=0,
           max=1))
```

class?Unif

Base documentation file for "static documentation"; it is merged with documentation computed by Roxygen.

man/Unif-class.Rd

```
\description{
  The uniform distribution has density
  \deqn{d(x) = \frac{1}{max - min}} for
  \eqn{min \leq x \leq max}.
}
\author{Manuel J. A. Eugster}
```

class?Unif

```
#' Density function.
#' @param object A \code{\linkS4class{Distribution}} object
#' @param x Vector of quantiles
#' ...
setGeneric('d',
function(object, n, ...) {

#' ...
#' @rdname d-methods
setMethod('d', signature=signature(object='Unif', x='numeric'),
function(object, x, log=FALSE) {
```

methods?d

Analoge for generics `r`, `p` and `q` and their methods.

Odds and ends of in-source documentation:

- Reduces the flexibility of writing Rd files, but enables standardized documentation.
- The new Rd structure defined by `parse_Rd()` allows the development of an Rd API and might give back some of the flexibility.
- With the new help system proposed by Duncon Murdoch for R 2.10 a lot of things with S4 will become easier; e.g., finding all methods for a specific class.

In-source "NAMESPACE" definition

The namespace roclet

The **namespace** roclet enables export, import and `useDynLib` directives.

```
> ns <- make.namespace.roclet(outfile='NAMESPACE')
> do.call(ns$parse, list.files('R/'))
```

```
#' ...
#' @export
mcpi <- function(n, verbose=FALSE) {

#' ...
#' @S3method print mcpi
print.mcpi <- function(x, ...) {

#' ...
#' @exportClass Unif
setClass('Unif',

#' ...
#' @exportMethod r
setGeneric('r', function(object, n, ...) {

#' ...
#' @importFrom stats runif
setMethod('r', signature=signature(object='Unif', n='numeric'),
function(object, n)
  return(runif(n, object@min, object@max)))
```

NAMESPACE

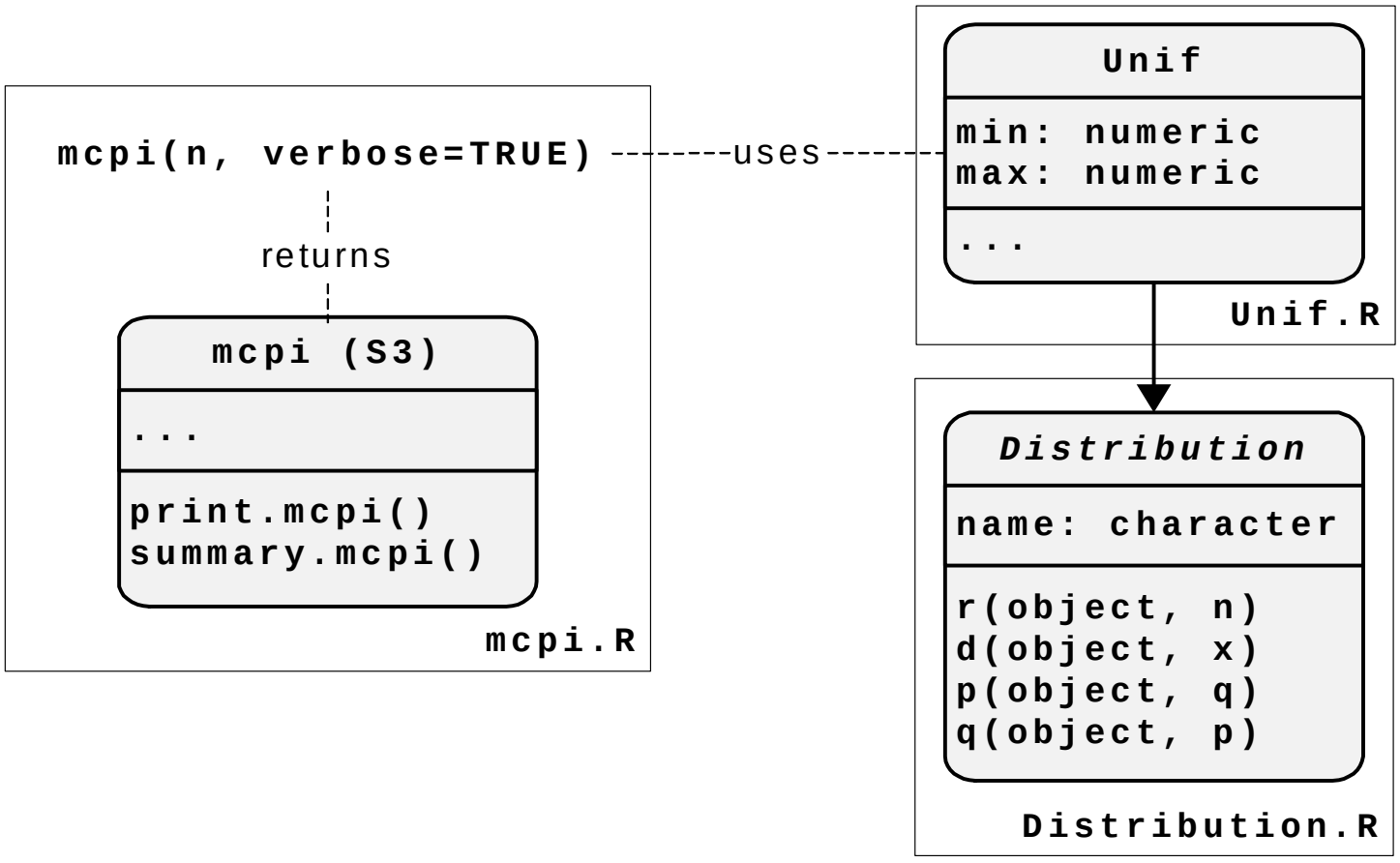
```
exportMethods(r)
exportMethods(d)
exportMethods(q)
exportMethods(p)
exportClasses(Unif)
export(Unif)
importFrom(stats, runif)
importFrom(stats, dunif)
importFrom(stats, qunif)
importFrom(stats, punif)
export(mcpi)
S3method(print, mcpi)
S3method(summary, mcpi)
```

In-source “Collate” definition

The collate roclet

The **collate** roclet provides an include directive for source files to specify dependencies. It topologically sorts the dependencies and writes the `Collate` field in the `DESCRIPTION` file.

```
> co <- make.collate.roclet('DESCRIPTION')
> do.call(co$parse, list.files('R/'))
```


mcpi(n, verbose=TRUE)
uses
returns
mcpi (S3)
...
print.mcpi()
summary.mcpi()
mcpi.R
Unif
min: numeric
max: numeric
...
Unif.R
Distribution
name: character
r(object, n)
d(object, x)
p(object, q)
q(object, p)
Distribution.R

R/Unif.R

```
#' @include Distribution.R
roxygen()

...
```

R/mcpi.R

```
#' @include Unif.R
{}

...
```

DESCRIPTION

```
Package: mcpi
Version: 0.1
...
Collate: 'Distribution.R' 'Unif.R' 'mcpi.R'
```

"Roxygenize"

Process a package with the Rd/Rd2, namespace and collate roclets.

Within R:

```
> roxygenize(package.dir='mcpi',
+            roxygen.dir='mcpi.roxygen')
```

Using `R CMD`:

```
$ R CMD roxygen mcpi mcpi.roxygen
```

Perspective

- Complete S4 integration.
- `proto` integration.
- Roxygen 1.0 along with the proceedings article.
- Support of Roxygen on R-Forge.

`http://www.roxygen.org`