Programming with R

John M. Chambers

August 8, 2007

Agenda

- What *is* our programming?
- Some principles,

  concepts, and techniques.
- How do they apply?

Bob’s Question:

“Just what is it that you really do, anyway?”

What is Programming?

(generally, or with R)

Science?

Art?

Engineering?

Craft?

Science?

No, we support science with “scientific” computing, but science is about learning, we’re about creating.

Art?

So Knuth said. Sort of--we have choices & aesthetics, but the software does have to work, too.

Engineering?

Closer, maybe: making things that work. But we have more choices, not just “applied science”.

Craft?

Making things with “dexterity + art”. Perhaps the closest fit. Maybe a new kind of “high craft” ?

What is Programming?

In the end, serious programming is a new kind of activity.

It has grown to huge importance in 50 years or less.

Valuable programming for data analysis: Are there general principles?

UseRs and ProgrammeRs

We're both, at different times.

The transition to programming should be easy and gradual.

Still, after a while the programming starts to have some value, and then we need principles to help increase that value.

Two Principles for Programming (with R)

Enable effective and rapid exploration of data (The Mission)

Provide trustworthy software (The Prime Directive)

or, “First, Tell no Lies”

Implications?

Many. Here’s a general one.

For trustworthy software, open-source systems are better, other things being equal.

Why? To allow drilling down to unknown depths in unexpected places.

Programming with R
(or other systems)

Useful to narrow down in two stages.

Concepts: a few key ideas;

Techniques: the many specific "recipes" for desired results.

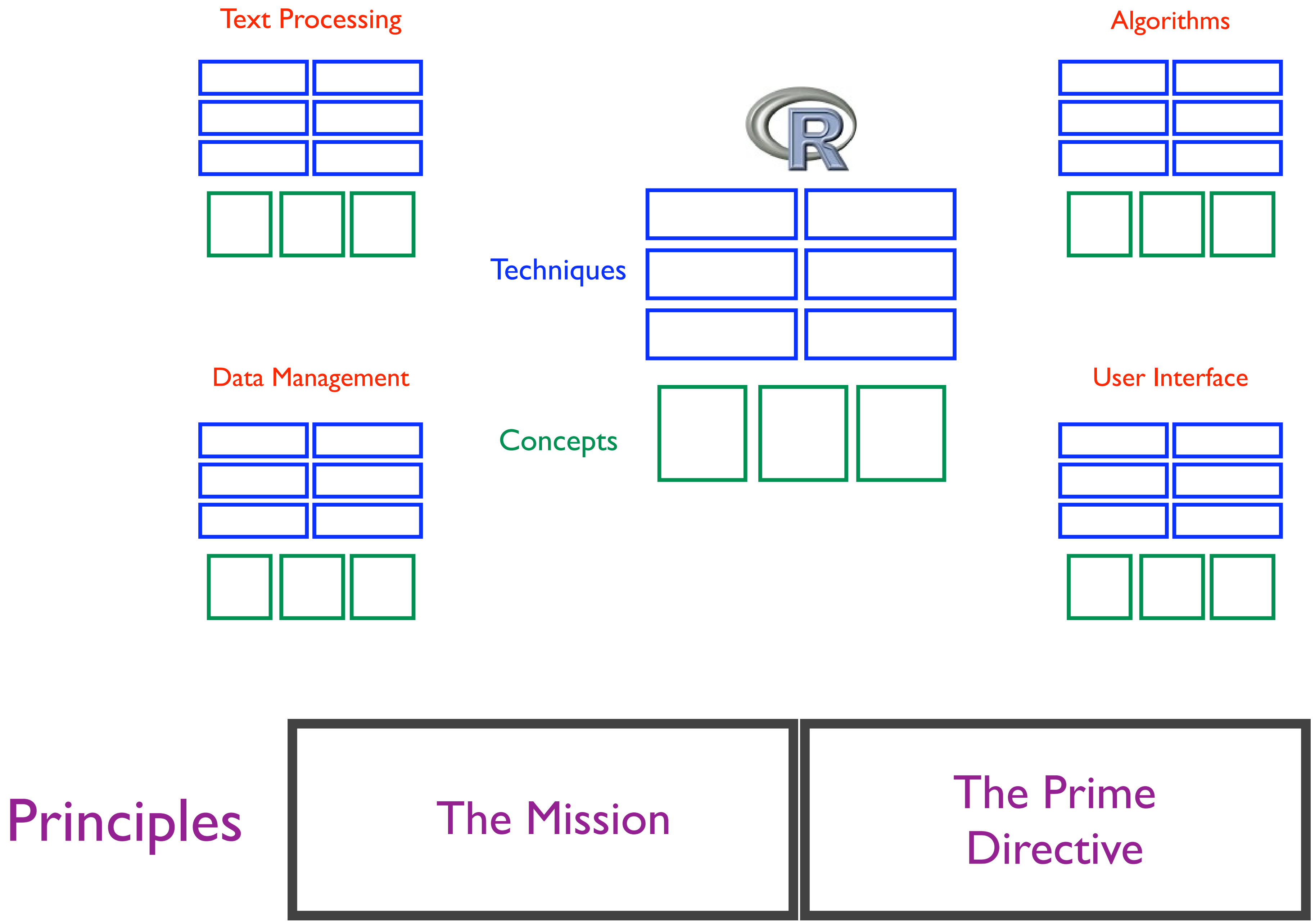
Key Concepts in R
(For general programming)

Function calls

Objects and names.

Corresponding techniques:

- creating a function object;
- debugging (`trace` and `recover`)


Text Processing
Algorithms
Techniques
Data Management
User Interface
Concepts
Principles
The Mission
The Prime Directive

Programming with R
(or other systems)

Useful to narrow down in two stages.

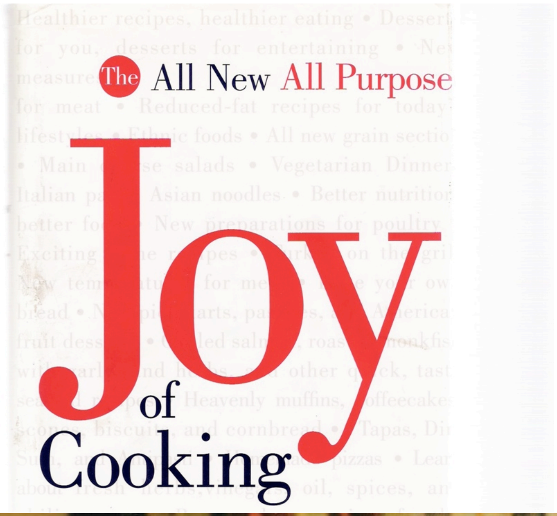
Concepts: a few key ideas;

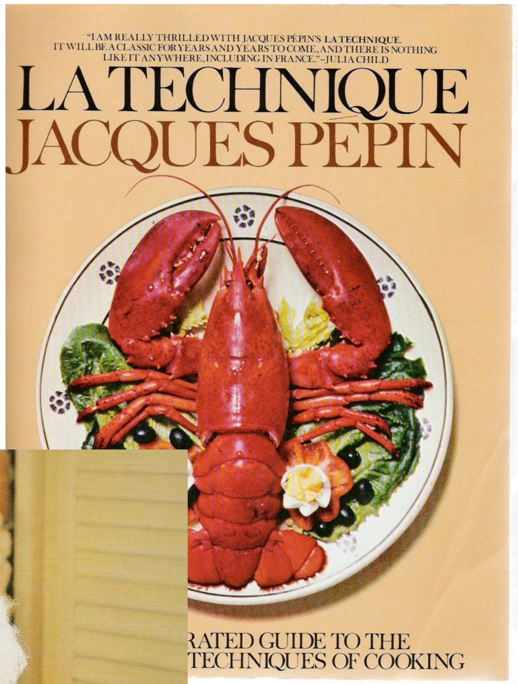
Techniques: the many specific "recipes" for desired results.

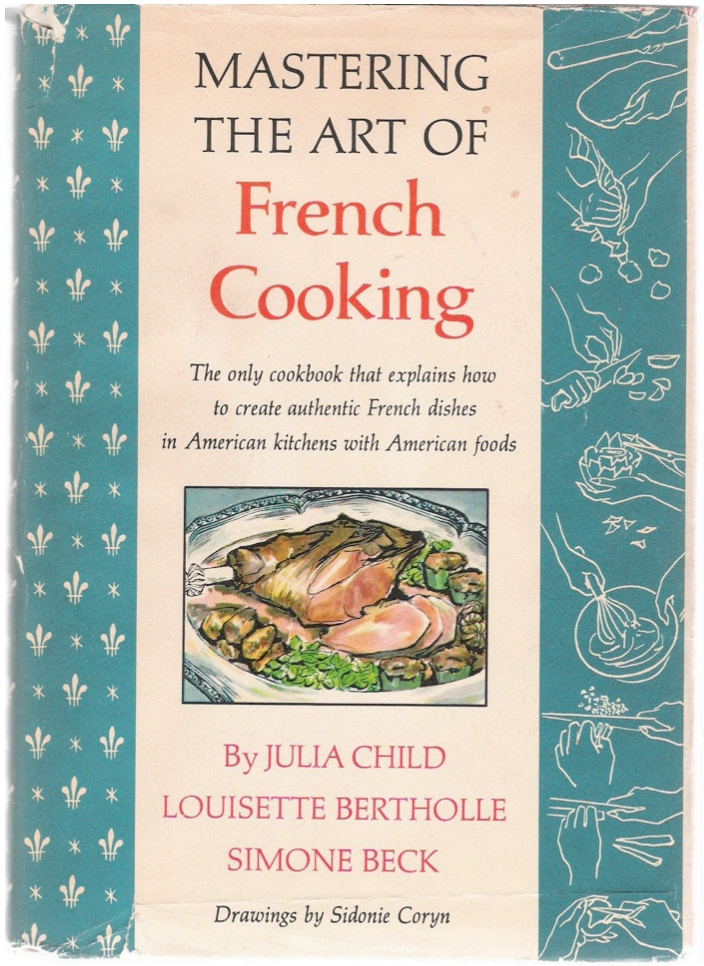
The distinction is useful, but NOT rigid:

techniques can lead to concepts.

Recipes, techniques, concepts?




"I AM REALLY THRILLED WITH JACQUES PÉPIN'S LA TECHNIQUE. IT WILL BE A CLASSIC FOR YEARS AND YEARS TO COME, AND THERE IS NOTHING LIKE IT ANYWHERE, INCLUDING IN FRANCE." –JULIA CHILD
LA TECHNIQUE
JACQUES PÉPIN
RATED GUIDE TO THE
TECHNIQUES OF COOKING


MASTERING THE ART OF
French Cooking
The only cookbook that explains how to create authentic French dishes in American kitchens with American foods
By JULIA CHILD
LOUISETTE BERTHOLLE
SIMONE BECK
Drawings by Sidonie Coryn

Classes and Methods in R

(no prior experience needed)

Why this example? a good serious area: Real "Programming with R"

Good to address some confusion with OOP languages. We are different.

Classes and Methods in R (Background)

Functions and objects, again, are the essence.

Everything happens as a function call & everything is an object.

Only one language, for data analysis and for programming.

Classes and Methods in R (Key Concepts)

Functions: Separate their purpose (functionality) from methods : this defines a generic function.

Objects: organize similar objects as a class of objects, define methods for classes of arguments to functions.

Classes and Methods in R
(Techniques to implement the Concepts)

- Define classes of objects that represent shared structures in objects;
- Define methods for important functions when those objects occur as arguments.

Classes and Methods in R Related to Both Principles

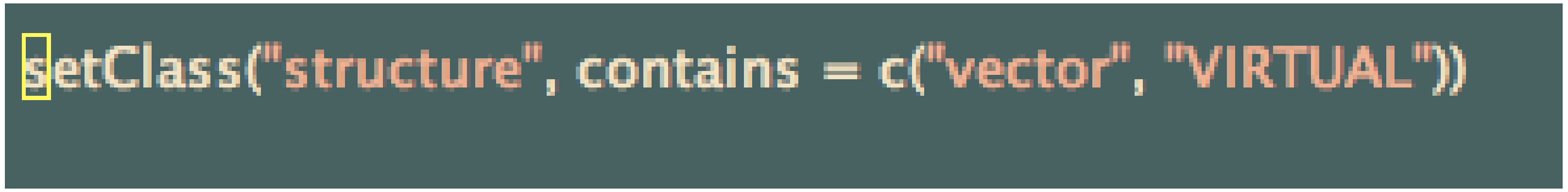
- Helping programmers make gradual extensions incrementally (the Mission)
- More potential for validation; simpler implementations (the Prime Directive)

Example:
Vector Structures

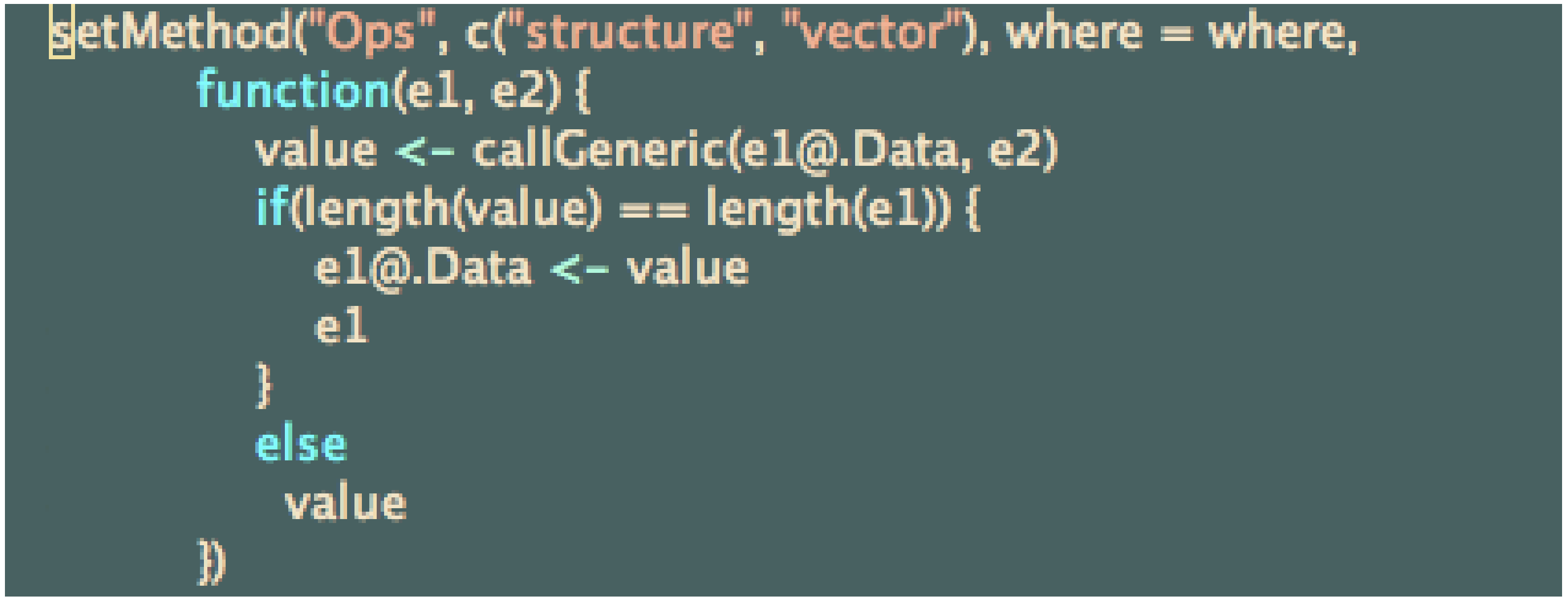
- Concept: Objects with structure (in space, time, layout) independent of individual values (e.g., matrix)
- Implies `x +1`, `x > 0`, `log(x)` structures, but `x[1:10]`, `sample(x, 5)` are not.
- Not consistent in R.

Example:
Vector Structures

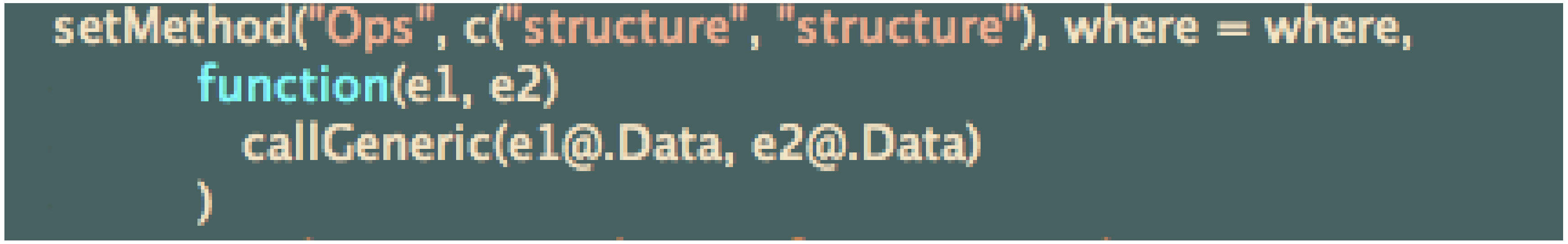
- Techniques: Define a formal class, "structure" which all formal structure classes extend.
- Define methods for groups of functions that implement the rules. (VERY simple).



```r
setClass("structure", contains = c("vector", "VIRTUAL"))
```



```r
setMethod("Ops", c("structure", "vector"), where = where,
          function(e1, e2) {
            value <- callGeneric(e1@.Data, e2)
            if(length(value) == length(e1)) {
              e1@.Data <- value
              e1
            }
            else
             value
          })
```



```r
setMethod("Ops", c("structure", "structure"), where = where,
          function(e1, e2)
            callGeneric(e1@.Data, e2@.Data)
          )
```