Introduction to the FLCore package

FLR Core Team

December 2, 2010

Outline

Outline

Outline

Outline

Outline

FLCore structure

Follows S4 paradigm. Objects are instances of a collection of data and methods, where the data structure and the methods to apply are defined by classes.

FLCore structure

Follows S4 paradigm. Objects are instances of a collection of data and methods, where the data structure and the methods to apply are defined by classes.

- Classes - abstract definition of data structure

FLCore structure

Follows S4 paradigm. Objects are instances of a collection of data and methods, where the data structure and the methods to apply are defined by classes.

- Classes - abstract definition of data structure
- Methods - implementation of actions

FLCore structure

Follows S4 paradigm. Objects are instances of a collection of data and methods, where the data structure and the methods to apply are defined by classes.

- Classes - abstract definition of data structure
- Methods - implementation of actions
- Objects - instances of the classes which have data following the class definition and methods depending on its class.

Inside FLCore

Inside FLCore

- Basic classes - *single array*:
  FLArray, FLQuant, FLCohort, FLQuantPoint, FLPar

Inside FLCore

- Basic classes - *single array*:
  FLArray, FLQuant, FLCohort, FLQuantPoint, FLPar
- Composite classes - *complex class composed of basic classes, both FLCore and R*:
  FLComp, FLBiol, FLCatch, FLFleet, FLIndex, FLMetier, FLModel, FLStock

Inside FLCore

- Basic classes - *single array*:
  FLArray, FLQuant, FLCohort, FLQuantPoint, FLPar
- Composite classes - *complex class composed of basic classes, both FLCore and R*:
  FLComp, FLBiol, FLCatch, FLFleet, FLIndex, FLMetier, FLModel, FLStock
- List classes - *list of objects of the same class*:
  FLIst, FLBiols, FLCatches, FLCohorts, FLFleets, FLIndices, FLMetiers, FLQuants, FLStocks

Inside FLCore

- Basic classes - *single array*:
  FLArray, FLQuant, FLCohort, FLQuantPoint, FLPar
- Composite classes - *complex class composed of basic classes, both FLCore and R*:
  FLComp, FLBiol, FLCatch, FLFleet, FLIndex, FLMetier, FLModel, FLStock
- List classes - *list of objects of the same class*:
  FLIst, FLBiols, FLCatches, FLCohorts, FLFleets, FLIndices, FLMetiers, FLQuants, FLStocks
- Model class - *composite classes specific for implementation of models*:
  FLModel, FLGrowth, FLSR

Inside FLCore

- Basic classes - *single array*:
  FLArray, FLQuant, FLCohort, FLQuantPoint, FLPar
- Composite classes - *complex class composed of basic classes, both FLCore and R*:
  FLComp, FLBiol, FLCatch, FLFleet, FLIndex, FLMetier, FLModel, FLStock
- List classes - *list of objects of the same class*:
  FLIst, FLBiols, FLCatches, FLCohorts, FLFleets, FLIndices, FLMetiers, FLQuants, FLStocks
- Model class - *composite classes specific for implementation of models*:
  FLModel, FLGrowth, FLSR
- Methods

Basic classes

	class	parent	nSlots	virtual	child	distance
1	FLArray	array	2	FALSE	FLQuant	1.00
2	FLArray	array	2	FALSE	FLCohort	1.00
3	FLArray	array	2	FALSE	FLQuantPoint	2.00
4	FLCohort	FLArray	2	FALSE		
5	FLQuant	FLArray	2	FALSE	FLQuantPoint	1.00
6	FLQuantPoint	FLQuant	2	FALSE		

FLQuant

Six dimensional array used to store data of a particular type (e.g. catch numbers).

Dimensions are:

FLQuant

Six dimensional array used to store data of a particular type (e.g. catch numbers).

Dimensions are:

1. User defined (age, length etc.)

FLQuant

Six dimensional array used to store data of a particular type (e.g. catch numbers).

Dimensions are:

1. User defined (age, length etc.)
2. Year

FLQuant

Six dimensional array used to store data of a particular type (e.g. catch numbers).

Dimensions are:

1. User defined (age, length etc.)
2. Year
3. Unit (substocks, male/female)

FLQuant

Six dimensional array used to store data of a particular type (e.g. catch numbers).

Dimensions are:

1. User defined (age, length etc.)
2. Year
3. Unit (substocks, male/female)
4. Season

FLQuant

Six dimensional array used to store data of a particular type (e.g. catch numbers).

Dimensions are:

1. User defined (age, length etc.)
2. Year
3. Unit (substocks, male/female)
4. Season
5. Area

FLQuant

Six dimensional array used to store data of a particular type (e.g. catch numbers).

Dimensions are:

1. User defined (age, length etc.)
2. Year
3. Unit (substocks, male/female)
4. Season
5. Area
6. Iter

FLQuant example

```
> flq <- window(landings.n(ple4), start = 1995, end = 2001)
> dimnames(flq)

$age
 [1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10"

$year
[1] "1995" "1996" "1997" "1998" "1999" "2000" "2001"

$unit
[1] "unique"

$season
[1] "all"

$area
[1] "unique"

$iter
[1] "1"
```

FLQuant methods

```
> getClassMethods("FLQuant", "package:FLCore")

  [1] "areaMeans"      "areaSums"       "areaVars"       "as.FLQuant"
  [5] "barchart"       "[<-"            "["              "bubbles"
  [9] "bwplot"         "capacity<-"     "catch<-"        "catch.n<-"
 [13] "catch.q<-"      "catch.wt<-"     "coerce"         "crewshare<-"
 [17] "cv"             "dimMeans"       "dimnames<-"     "dims"
 [21] "dimSums"        "dimVars"        "discards<-"     "discards.n<-"
 [25] "discards.sel<-" "discards.wt<-"  "dotplot"        "effort<-"
 [29] "effshare<-"     "E"              "fcost<-"        "fec<-"
 [33] "FLBiol"         "FLCatch"        "FLCohort"       "FLIndex"
 [37] "FLMetier"       "FLQuant"        "FLQuantPoint"   "FLStock"
 [41] "harvest<-"      "harvest.spwn<-" "histogram"      "index<-"
 [45] "index.q<-"      "index.var<-"    "iter<-"         "iterMeans"
 [49] "iters"          "iterVars"       "jacknife"       "landings<-"
 [53] "landings.n<-"   "landings.sel<-" "landings.wt<-"  "loglAR1"
 [57] "mat<-"          "m<-"            "m.spwn<-"       "names"
 [61] "n<-"            "plot"           "price<-"        "print"
 [65] "propagate"      "pv"             "quant"          "quantile"
 [69] "quantMeans"     "quantSums"      "quantTotals"    "quantVars"
 [73] "rec<-"          "r"              "rlnorm"         "rnorm"
 [77] "rpois"          "rSq"            "seasonMeans"    "seasonSums"
 [81] "seasonVars"     "sel.pattern<-"  "setPlusGroup"   "sp"
 [85] "spr0"           "spwn<-"         "ssb<-"          "stock<-"
 [89] "stock.n<-"      "stock.wt<-"     "stripplot"      "sweep"
 [93] "unitMeans"      "units"          "unitSums"       "unitVars"
 [97] "vcost<-"        "window"         "wt<-"           "xyplot"
[101] "yearMeans"      "yearSums"       "yearTotals"     "yearVars"
```

FLQuantPoint

Six dimensional array used to summarize iterations of FLQuant objects.

```
> dimnames(FLQuantPoint(flq))

$age
 [1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10"

$year
[1] "1995" "1996" "1997" "1998" "1999" "2000" "2001"

$unit
[1] "unique"

$season
[1] "all"

$area
[1] "unique"

$iter
[1] "mean"   "median" "var"    "uppq"   "lowq"
```

FLQuantPoint example

```
> flq[1:4, 1:5]

An object of class "FLQuant"
, , unit = unique, season = all, area = unique

   year
age 1995   1996   1997   1998   1999
  1   7751   1104    892    196    549
  2  36575  42496  42855  30401   8689
  3  81398  64382  86948  68920 155971
  4  78370  46359  43669  56329  39857

units:  thousands

> FLQuantPoint(flq[1:4, 1:5])

An object of class "FLQuantPoint":
-- median:
, , unit = unique, season = all, area = unique

   year
age  1995  1996  1997  1998   1999
  1  7751  1104   892   196    549
  2 36575 42496 42855 30401   8689
  3 81398 64382 86948 68920 155971
  4 78370 46359 43669 56329  39857

units:  NA
```

FLQuantPoint methods

```
> getClassMethods("FLQuantPoint", "package:FLCore")

 [1] "[<-"      "["        "coerce"   "FLQuant"  "lowq<-"   "lowq"
 [7] "mean"     "mean<-"   "median<-" "median"   "plot"     "quantile"
[13] "rgamma"   "rlnorm"   "rnorm"    "show"     "summary"  "uppq<-"
[19] "uppq"     "var<-"    "var"
```

FLCohort

Six dimensional array used to store cohort data.

```
> dimnames(FLCohort(flq))

$age
 [1] "1"  "2"  "3"  "4"  "5"  "6"  "7"  "8"  "9"  "10"

$cohort
 [1] "1985" "1986" "1987" "1988" "1989" "1990" "1991" "1992" "1993" "1994"
[11] "1995" "1996" "1997" "1998" "1999" "2000"

$unit
[1] "unique"

$season
[1] "all"

$area
[1] "unique"

$iter
[1] "1"
```

FLCohort example

```
> flq[1:4, 1:5]

An object of class "FLQuant"
, , unit = unique, season = all, area = unique

   year
age 1995   1996   1997   1998   1999
  1   7751   1104    892    196    549
  2  36575  42496  42855  30401   8689
  3  81398  64382  86948  68920 155971
  4  78370  46359  43669  56329  39857

units:  thousands

> FLCohort(flq[1:4, 1:5])

An object of class "FLCohort"
, , unit = unique, season = all, area = unique

   cohort
age 1991   1992   1993   1994   1995   1996   1997   1998
  1     NA     NA     NA   7751   1104    892    196    549
  2     NA     NA  36575  42496  42855  30401   8689     NA
  3     NA  81398  64382  86948  68920 155971     NA     NA
  4  78370  46359  43669  56329  39857     NA     NA     NA

units:  thousands
```

FLCohort methods

```
> getClassMethods("FLCohort", "package:FLCore")

 [1] "bubbles"    "ccplot"     "coerce"     "dimnames<-" "dims"
 [6] "flc2flq"    "FLCohort"   "iter<-"     "plot"       "propagate"
[11] "show"       "xyplot"
```

FLPar

A two dimensional array used to store parameter's data.

```
> dimnames(new("FLPar"))

$param
[1] ""

$iter
[1] "1"
```

FLPar methods

```
> getClassMethods("FLPar", "package:FLCore")

 [1] "ab"            "Arith"         "as.data.frame" "[<-"
 [5] "["             "coerce"        "convertFLPar"  "densityplot"
 [9] "dims"          "FLPar"         "fmle"          "histogram"
[13] "iter<-"        "iter"          "mean"          "median"
[17] "names<-"       "names"         "params<-"      "plot"
[21] "propagate"     "show"          "splom"         "summary"
[25] "sv"            "sweep"         "units<-"       "units"
[29] "var"
```

Composite classes

Classes that use FLQuant classes to define their slots.

	class	parent	nSlots	virtual	child	distance
1	FLBiol	FLComp	8	FALSE		
2	FLCatch	FLComp	13	FALSE		
3	FLFleet	FLComp	8	FALSE		
4	FLIndex	FLComp	12	FALSE		
5	FLMetier	FLComp	7	FALSE		
6	FLModel	FLComp	14	FALSE	FLSR	1.00
7	FLStock	FLComp	20	FALSE		

FLStock

Represents a fish stock and comprises a number of slots.

```
> showClass("FLStock")
```

```
Class "FLStock" [package "FLCore"]

Slots:

Name:         catch      catch.n     catch.wt     discards   discards.n
Class:      FLQuant      FLQuant      FLQuant      FLQuant      FLQuant

Name:   discards.wt     landings   landings.n  landings.wt        stock
Class:      FLQuant      FLQuant      FLQuant      FLQuant      FLQuant

Name:       stock.n     stock.wt            m          mat      harvest
Class:      FLQuant      FLQuant      FLQuant      FLQuant      FLQuant

Name:  harvest.spwn       m.spwn         name         desc        range
Class:      FLQuant      FLQuant    character    character      numeric

Extends: "FLComp"
```

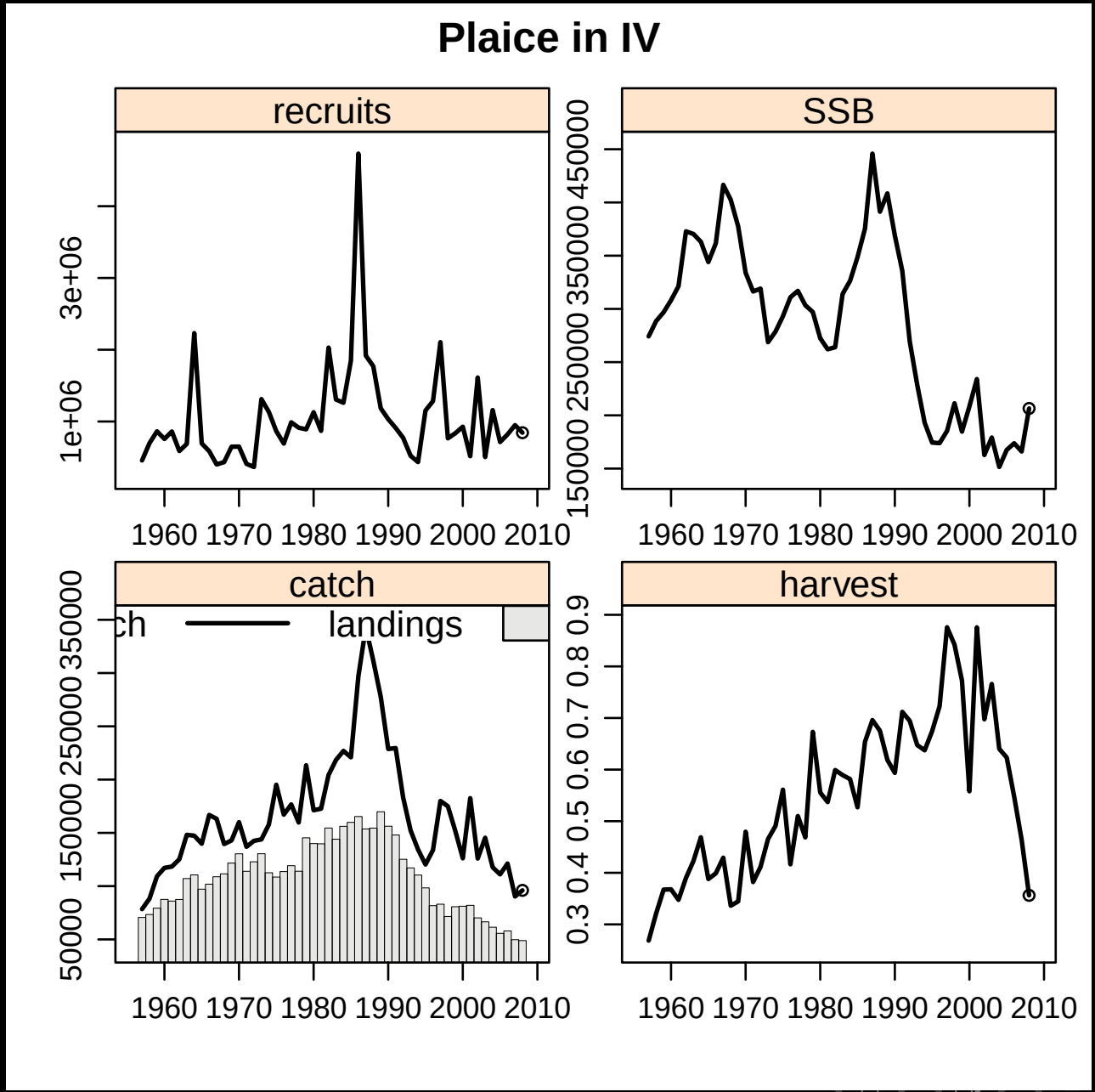
FLStock example

```
> summary(ple4)

An object of class "FLStock"

Name: Plaice in IV
Description: Imported from a VPA file. ( N:\Projecten\ICES WG\Demersale werkgroep WGNSSK\2009\st
Range:         min        max        pgroup        minyear        maxyear        minfbar
        1         10         10         1957         2008         2         6
Quant: age

catch          : [ 1 52 1 1 1 1 ], units =  tonnes
catch.n        : [ 10 52 1 1 1 1 ], units =  thousands
catch.wt       : [ 10 52 1 1 1 1 ], units =  kg
discards       : [ 1 52 1 1 1 1 ], units =  tonnes
discards.n     : [ 10 52 1 1 1 1 ], units =  thousands
discards.wt    : [ 10 52 1 1 1 1 ], units =  kg
landings       : [ 1 52 1 1 1 1 ], units =  tonnes
landings.n     : [ 10 52 1 1 1 1 ], units =  thousands
landings.wt    : [ 10 52 1 1 1 1 ], units =  kg
stock          : [ 1 52 1 1 1 1 ], units =  tonnes
stock.n        : [ 10 52 1 1 1 1 ], units =  thousands
stock.wt       : [ 10 52 1 1 1 1 ], units =  kg
m              : [ 10 52 1 1 1 1 ], units =  NA
mat            : [ 10 52 1 1 1 1 ], units =  NA
harvest        : [ 10 52 1 1 1 1 ], units =  f
harvest.spwn   : [ 10 52 1 1 1 1 ], units =  NA
m.spwn         : [ 10 52 1 1 1 1 ], units =  NA
```

FLStock methods

```
> getClassMethods("FLStock", "package:FLCore")

 [1] "as.FLBiol"        "as.FLSR"          "[<-"              "["
 [5] "+"                "catch<-"          "catch"            "catch.n<-"
 [9] "catch.n"          "catch.wt<-"       "catch.wt"         "coerce"
[13] "computeCatch"     "computeDiscards"  "computeLandings"  "computeStock"
[17] "dimnames<-"       "discards<-"       "discards"         "discards.n<-"
[21] "discards.n"       "discards.wt<-"    "discards.wt"      "expand"
[25] "fapex"            "fbar"             "harvest<-"        "harvest"
[29] "harvest.spwn<-"   "harvest.spwn"     "landings<-"       "landings"
[33] "landings.n<-"     "landings.n"       "landings.wt<-"    "landings.wt"
[37] "mat<-"            "mat"              "m<-"              "m"
[41] "m.spwn<-"         "m.spwn"           "plot"             "rec"
[45] "r"                "setPlusGroup"     "sp"               "spr0"
[49] "ssb"              "ssbpurec"         "stock<-"          "stock"
[53] "stock.n<-"        "stock.n"          "stock.wt<-"       "stock.wt"
[57] "summary"          "survprob"         "trim"             "tsb"
```

FLStock plot



FLBiol

Represents a biological population

```
> showClass("FLBiol")

Class "FLBiol" [package "FLCore"]

Slots:

Name:          n         m        wt       fec      spwn      name      desc
Class:   FLQuant   FLQuant   FLQuant   FLQuant   FLQuant character character

Name:      range
Class:   numeric

Extends: "FLComp"
```

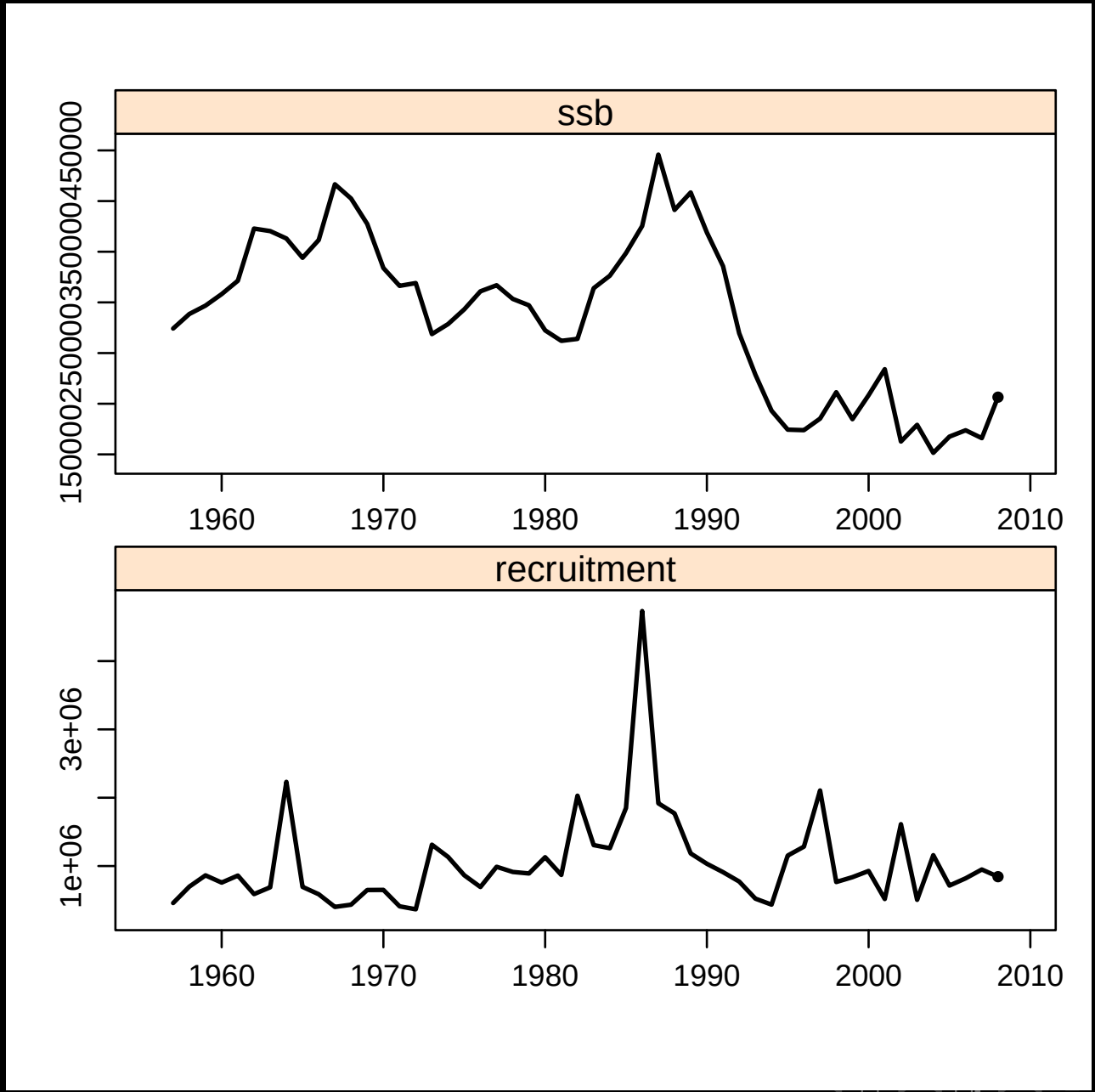
FLBiol example

```
> summary(flbiol)

An object of class "FLBiol"

Name: Plaice in IV
Description: Imported from a VPA file. ( N:\Projecten\ICES WG\Demersale werkgroep WGNSSK\2009\st
Range:         min        max        pgroup        minyear        maxyear        minfbar
        1         10         10        1957           2008           2              6
Quant: age

n             : [ 10 52 1 1 1 1 ], units =  thousands
m             : [ 10 52 1 1 1 1 ], units =  NA
wt            : [ 10 52 1 1 1 1 ], units =  kg
fec           : [ 10 52 1 1 1 1 ], units =  NA
spwn          : [ 10 52 1 1 1 1 ], units =  NA
```

FLBiol methods

```
> getClassMethods("FLBiol", "package:FLCore")

 [1] "as.FLBiol"     "as.FLSR"       "catch.n"       "coerce"
 [5] "computeStock"  "fbar"          "fec<-"         "fec"
 [9] "harvest"       "leslie"        "mean.lifespan" "m<-"
[13] "m"             "n<-"           "n"             "plot"
[17] "rec"           "r"             "setPlusGroup"  "spwn<-"
[21] "spwn"          "ssb"           "ssn"           "summary"
[25] "survprob"      "tsb"           "wt<-"          "wt"
```

FLBiol plot


ssb
150000
250000
350000
450000
1960
1970
1980
1990
2000
2010
recruitment
1e+06
3e+06
1960
1970
1980
1990
2000
2010

FLIndex

Represents a index (e.g. index of abundance from a survey)

```
> showClass("FLIndex")

Class "FLIndex" [package "FLCore"]

Slots:

Name:          type distribution         index     index.var       catch.n
Class:    character    character       FLQuant       FLQuant       FLQuant

Name:      catch.wt       effort  sel.pattern       index.q          name
Class:      FLQuant      FLQuant      FLQuant       FLQuant     character

Name:          desc        range
Class:    character      numeric

Extends: "FLComp"
```

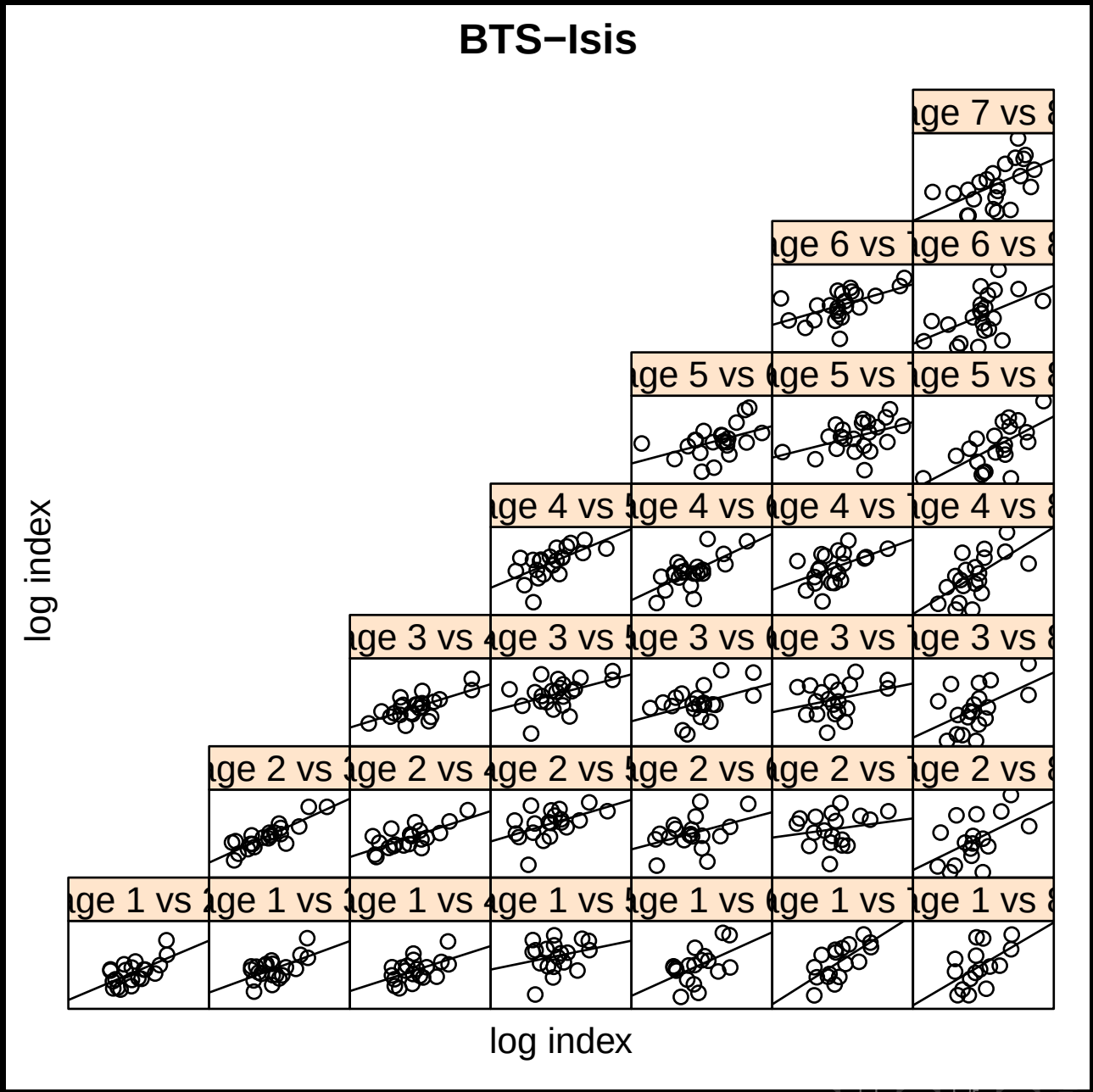
FLIndex example

```
> summary(ple4.index)

An object of class "FLIndex"

Name: BTS-Isis
Description: Plaice in IV . Imported from VPA file.
Range:         min        max        pgroup        minyear        maxyear        startf        e
        1          8         NA        1985          2008          0.66          0.75
Type :
Distribution :
Quant: age

index          : [ 8 24 1 1 1 1 ], units =  NA
index.var      : [ 8 24 1 1 1 1 ], units =  NA
catch.n        : [ 8 24 1 1 1 1 ], units =  NA
catch.wt       : [ 8 24 1 1 1 1 ], units =  NA
effort         : [ 1 24 1 1 1 1 ], units =  NA
sel.pattern    : [ 8 24 1 1 1 1 ], units =  NA
index.q        : [ 8 24 1 1 1 1 ], units =  NA
```

FLIndex methods

```
> getClassMethods("FLIndex", "package:FLCore")

 [1] "["             "catch.n<-"     "catch.n"       "catch.wt<-"
 [5] "catch.wt"      "coerce"        "computeCatch"  "dims"
 [9] "effort<-"      "effort"        "index<-"       "index"
[13] "index.q<-"     "index.q"       "index.var<-"   "index.var"
[17] "plot"          "sel.pattern<-" "sel.pattern"   "summary"
[21] "trim"          "type<-"        "type"
```

FLIndex plot



FLCatch

Represents the catch of a fleet

```
> showClass("FLCatch")

Class "FLCatch" [package "FLCore"]

Slots:

Name:      landings   landings.n  landings.wt landings.sel     discards
Class:      FLQuant      FLQuant      FLQuant      FLQuant      FLQuant

Name:    discards.n  discards.wt discards.sel      catch.q        price
Class:      FLQuant      FLQuant      FLQuant      FLQuant      FLQuant

Name:          name         desc        range
Class:    character    character      numeric

Extends: "FLComp"
```

FLIndex example

```
> summary(bt4[["TBB"]][["ple"]])

An object of class "FLCatch"

Name: ple
Description: North Sea Plaice
Range:         min        max        pgroup        minyear        maxyear
        1        15         NA        1957        2001
Quant: age

landings      : [ 1 45 1 1 1 1 ], units =  tonnes
landings.n    : [ 15 45 1 1 1 1 ], units =  thousands
landings.wt   : [ 15 45 1 1 1 1 ], units =  kg
landings.sel  : [ 15 45 1 1 1 1 ], units =  NA
discards      : [ 1 45 1 1 1 1 ], units =  tonnes
discards.n    : [ 15 45 1 1 1 1 ], units =  thousands
discards.wt   : [ 15 45 1 1 1 1 ], units =  kg
discards.sel  : [ 15 45 1 1 1 1 ], units =  NA
catch.q       : [ 15 45 1 1 1 1 ], units =  thousands
price         : [ 15 45 1 1 1 1 ], units =  NA
```

FLCatch methods

```
> getClassMethods("FLCatch", "package:FLCore")

 [1] "[<-"              "["                "catches<-"        "catch"
 [5] "catchNames"       "catch.n"          "catch.q<-"        "catch.q"
 [9] "catch.sel"        "catch.wt"         "coerce"           "computeCatch"
[13] "computeDiscards"  "computeLandings"  "discards<-"       "discards"
[17] "discards.n<-"     "discards.n"       "discards.sel<-"   "discards.sel"
[21] "discards.wt<-"    "discards.wt"      "FLFleet"          "FLMetier"
[25] "landings<-"       "landings"         "landings.n<-"     "landings.n"
[29] "landings.sel<-"   "landings.sel"     "landings.wt<-"    "landings.wt"
[33] "price<-"          "price"            "revenue"          "setPlusGroup"
[37] "summary"          "trim"
```

FLMetier

Represents a fleet's metier (classification of activity targeting a species or group of species, in a specific period and area, with a particular gear)

```
> showClass("FLMetier")

Class "FLMetier" [package "FLCore"]

Slots:

Name:       gear  effshare     vcost   catches      name      desc     range
Class: character   FLQuant   FLQuant FLCatches character character   numeric

Extends: "FLComp"
```

FLMetier example

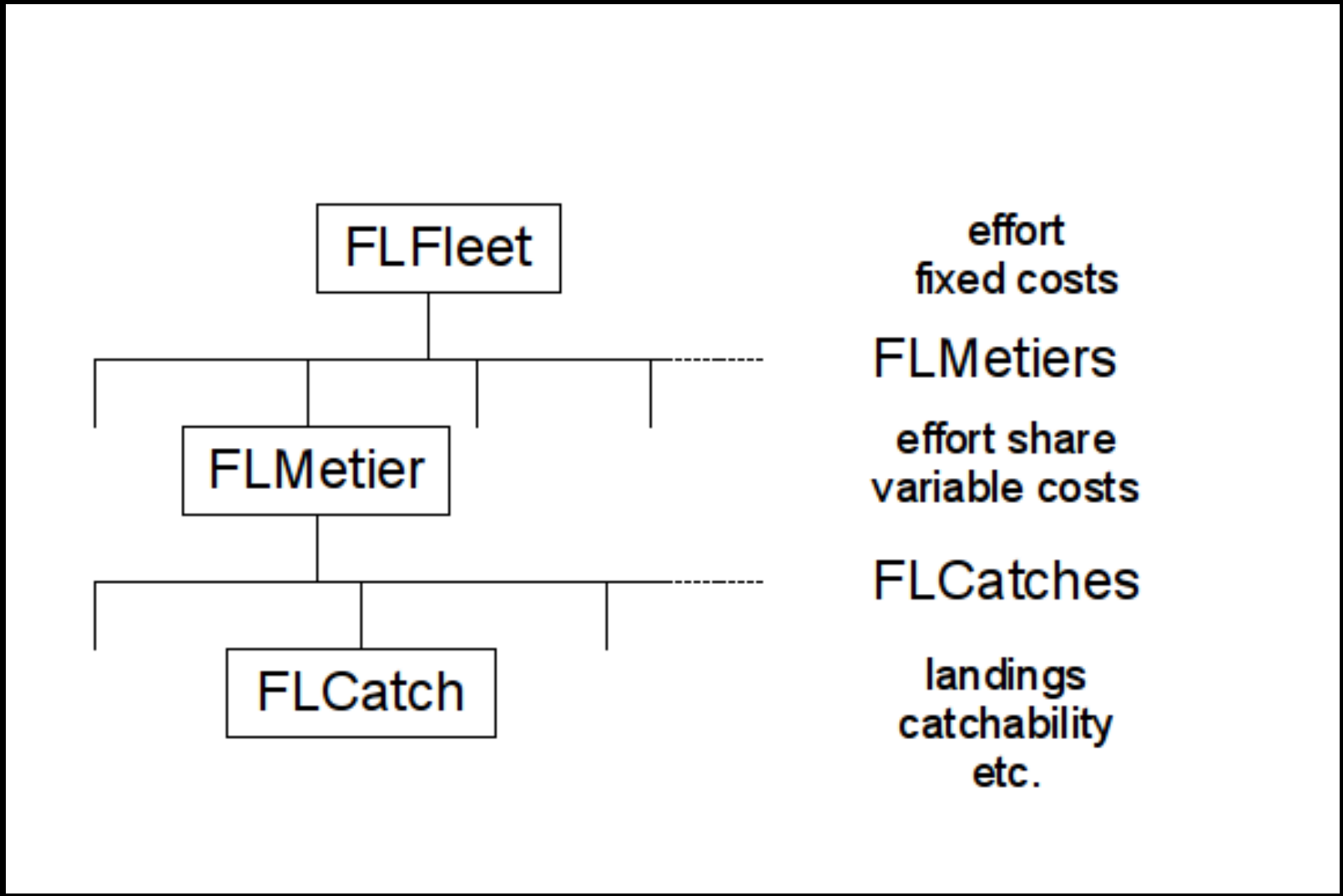
```
> summary(bt4[["TBB"]])

An object of class "FLMetier"

Name: TBB
Description:
Range:         min         max        pgroup        minyear        maxyear
        1          15          NA        1957        2001
Gear :  NA
Quant: age

effshare      : [ 1 45 1 1 1 1 ], units =  NA
vcost         : [ 1 45 1 1 1 1 ], units =  NA

Catches:
         ple : [ 15 45 1 1 1 1 ]
         sol : [ 10 45 1 1 1 1 ]
```

FLMetier methods

```
> getClassMethods("FLMetier", "package:FLCore")

 [1] "["              "[["             "catches<-"      "catches"
 [5] "catch"          "catchNames"     "catch.n"        "catch.q<-"
 [9] "catch.q"        "catch.sel"      "catch.wt"       "coerce"
[13] "computeCatch"   "computeDiscards" "computeLandings" "discards<-"
[17] "discards"       "discards.n<-"   "discards.n"     "discards.sel<-"
[21] "discards.sel"   "discards.wt<-"  "discards.wt"    "effshare<-"
[25] "effshare"       "FLFleet"        "gear<-"         "gear"
[29] "iter"           "landings<-"     "landings"       "landings.n<-"
[33] "landings.n"     "landings.sel<-" "landings.sel"   "landings.wt<-"
[37] "landings.wt"    "metier<-"       "price<-"        "price"
[41] "propagate"      "revenue"        "summary"        "trim"
[45] "vcost<-"        "vcost"
```

FLFleet

A more complicated class with three levels: Fleet, Metier and Catch



FLFleet

```
> showClass("FLFleet")

Class "FLFleet" [package "FLCore"]

Slots:

Name:     effort     fcost  capacity crewshare   metiers      name      desc
Class:   FLQuant   FLQuant   FLQuant   FLQuant FLMetiers character character

Name:      range
Class:   numeric

Extends: "FLComp"
```

FLFleet example

```
> summary(bt4)

An object of class "FLFleet"

Name: beam trawl fleet
Description: Example of an FLFleet
Range:         min        max        pgroup        minyear        maxyear
        0        0         NA        1957        2001
Quant: age

effort        : [ 1 45 1 1 1 1 ], units =  NA
fcost         : [ 1 45 1 1 1 1 ], units =  NA
capacity      : [ 1 45 1 1 1 1 ], units =  NA
crewshare     : [ 1 45 1 1 1 1 ], units =  NA

Metiers:
         TBB :
                ple : [ 15 45 1 1 1 1 ]
                sol : [ 10 45 1 1 1 1 ]
```

FLFleet methods

```
> getClassMethods("FLFleet", "package:FLCore")

 [1] "as.data.frame"   "as.FLIndex"      "["               "[["
 [5] "capacity<-"      "capacity"        "catches"         "catch"
 [9] "catchNames"      "catch.n"         "catch.q<-"       "catch.q"
[13] "catch.sel"       "catch.wt"        "coerce"          "computeCatch"
[17] "computeDiscards" "computeLandings" "crewshare<-"     "crewshare"
[21] "dims"            "discards<-"      "discards"        "discards.n<-"
[25] "discards.n"      "discards.sel<-"  "discards.sel"    "discards.wt<-"
[29] "discards.wt"     "effort<-"        "effort"          "effshare"
[33] "fcost<-"         "fcost"           "FLFleet"         "iter"
[37] "landings<-"      "landings"        "landings.n<-"    "landings.n"
[41] "landings.sel<-"  "landings.sel"    "landings.wt<-"   "landings.wt"
[45] "metier<-"        "metier"          "metiers<-"       "metiers"
[49] "price<-"         "price"           "propagate"       "revenue"
[53] "summary"         "trim"            "vcost"           "window"
```

List classes

	class	parent	nSlots	virtual	child	distance
1	FLBiols	FLlst	4	FALSE		
2	FLCatches	FLlst	4	FALSE		
3	FLCohorts	FLlst	4	FALSE		
4	FLFleets	FLlst	4	FALSE		
5	FLIndices	FLlst	4	FALSE		
6	FLMetiers	FLlst	4	FALSE		
7	FLQuants	FLlst	4	FALSE		
8	FLStocks	FLlst	4	FALSE		

FLlist

A list of other classes

```
> showClass("FLlst")
```

```
Class "FLlst" [package "FLCore"]

Slots:

Name:       .Data      names       desc       lock
Class:       list character character    logical

Extends:
Class "list", from data part
Class "vector", by class "list", distance 2

Known Subclasses: "FLStocks", "FLIndices", "FLBiols", "FLCatches", "FLMetiers", "FLFleets",
"FLQuants", "FLCohorts"
```

FLlst example #1

```
> summary(ple4.indices)

An object of class "FLIndices"

Elements: BTS-Isis BTS-Tridens SNS

Name: BTS-Isis
       Description: Plaice in IV . Imported from VPA file.
       Range:         min        max        pgroup        minyear        maxyear        startf
                1         8         NA       1985          2008           0.66           0.75
       Quant: age
       dim: 8 24 1 1 1 1
Name: BTS-Tridens
       Description: Plaice in IV . Imported from VPA file.
       Range:         min        max        pgroup        minyear        maxyear        startf
                1         9         NA       1996          2008           0.66           0.75
       Quant: age
       dim: 9 13 1 1 1 1
Name: SNS
       Description: Plaice in IV . Imported from VPA file.
       Range:         min        max        pgroup        minyear        maxyear        startf
                1         3         NA       1982          2008           0.66           0.75
       Quant: age
       dim: 3 27 1 1 1 1
```

FLlst example #2

```
> flqs <- FLQuants(f1 = catch(ple4), f2 = landings(ple4))
> summary(flqs)

An object of class "FLQuants"

Elements: f1 f2

Name: f1
        dim  :  1 52 1 1 1 1
        quant:  age
        units:  tonnes

        Min    :  78422.95
        1st Qu.:  126077.3
        Mean   :  165127
        Median :  151975.4
        3rd Qu.:  182754.8
        Max    :  342985.1
        NAs    :  0 %
Name: f2
        dim  :  1 52 1 1 1 1
        quant:  age
        units:  tonnes

        Min    :  48874
        1st Qu.:  81541.75
        Mean   :  108403.7
        Median :  110466
        3rd Qu.:  132758.5
        Max    :  169818
        NAs    :  0 %
```

FLlst methods

```
> getClassMethods("FLlst", "package:FLCore")

 [1] "[<-"         "["           "[[<-"        "$<-"         "coerce"
 [6] "lapply"      "model.frame" "names"       "range"       "summary"
[11] "window"
```

Model classes

	class	parent	nSlots	virtual	child	distance
1	FLGrowth	FLModel	15	FALSE		
2	FLSR	FLModel	18	FALSE		

FLSR

Represents a stock-recruitment relationship and allows the estimation of its parameters.

```
> showClass("FLSR")

Class "FLSR" [package "FLCore"]

Slots:

Name:        rec       ssb     covar  logerror     model      logl        gr
Class:   FLQuant   FLQuant  FLQuants   logical   formula  function  function

Name:    initial    params    logLik      vcov   hessian   details residuals
Class:  function     FLPar    logLik     array     array      list   FLArray

Name:     fitted      name      desc     range
Class:   FLArray character character   numeric

Extends:
Class "FLModel", directly
Class "FLComp", by class "FLModel", distance 2
```

FLSR example

```
> summary(nsher)

An object of class "FLSR"

Name: Autumn spawning herring in IV, V  3/4/2005 14:46
Description: 'rec' and 'ssb' slots obtained from a 'FLStock' object
Range:

Quant: age

rec           : [ 1 45 1 1 1 1 ], units =  NA
ssb           : [ 1 45 1 1 1 1 ], units =  NA
residuals     : [ 1 45 1 1 1 1 ], units =  NA NA
fitted        : [ 1 45 1 1 1 1 ], units =  NA

Model:         rec ~ a * ssb * exp(-b * ssb)
<environment: 0x3558dc0>
Parameters:
    params
iter     a        b
   1 119.4 0.009027

Log-likelihood:  16.352(0)
Variance-covariance:
               a           b
  a 258.66388793 1.838394e-02
  b   0.01838394 2.002586e-06
```

FLSR methods

```
> getClassMethods("FLSR", "package:FLCore")

 [1] "ab"      "covar<-" "covar"   "fmle"    "lowess"  "plot"    "rec<-"
 [8] "rec"     "spr0"    "ssb<-"   "ssb"     "summary" "sv"
```

FLSR plot

NULL