""log4r"

A log4j-Comparable Logging Framework for ooRexx Applications"

2007 International Rexx Symposium

Tampa, Florida, U.S.A. (April 2007)

Rony G. Flatscher (Rony.Flatscher@wu-wien.ac.at)
Wirtschaftsuniversität Wien, Austria (http://www.wu-wien.ac.at)

Agenda

- "log4r" – What can it do ?
- Architecture
  - LogManager, et.al.
  - Loggers, appenders, layouts, filters
- Concluding example
  - Using built-in log4r logging from class libraries
- Roundup and Outlook

```
say ' The beginning: '~center(50, "-") /* draw a line    */  /* "t0.rex" */

p=.person~new("Peedin", "Lee", 250000) /* create a person*/
say "p="p~string           /* show the person's state    */
p~increaseSalary(12345.67) /* increase salary            */
p~increaseSalary("abc")    /* provoke an error           */
p~increaseSalary(-1000)    /* decrease salary            */

say ' The end. '~center(50, "-") /* draw a line          */

/* ============================================================== */
::class person          /* class "PERSON"                */

/* ----------------------------------------------------- */
::method init           /* method "INIT" (constructor)   */
  expose familyName firstName salary
  use arg familyName, firstName, salary

::method familyName     attribute
::method firstName      attribute
::method salary         attribute

/* ----------------------------------------------------- */
::method string         /* create a string rendering of a person  */
  expose familyName firstName salary
  return familyName"," firstName":" salary

/* ----------------------------------------------------- */
::method increaseSalary /* method to increase the salary */
  expose salary
  parse arg raise

  signal on syntax      /* in case arithmetic creates a condition */
  salary=salary+raise
  return
syntax:                 /* just there to let the program continue */

/* ----------------------------------------------------- */
::method uninit         /* optional destructor method    */
```

"rexx t0.rex"

```
----------------- The beginning: -----------------
p=Peedin, Lee: 250000
-------------------- The end. --------------------
```

```
say ' The beginning: '~center(50, "-") /* draw a line    */  /* "t1.rex" */

call load_log4r         /* load the 'log4r' framework    */
l=.logManager~getLogger("rgf.test") /* get/create a logger named 'rgf.test'   */

parse arg logLevel      /* retrieve logLevel, if supplied */
if logLevel="" then l~logLevel="OFF"      /* do not show any log messages     */
               else l~logLevel=logLevel   /* set logLevel to argument's value */
l~debug("just created a logger named 'rgf.test':" pp(l~string))

parse source s          /* get source information        */
l~trace("source:" pp(s))

p=.person~new("Peedin", "Lee", 250000) /* create a person*/
say "p="pp(p~string)    /* pp() defined in 'log4r' framework   */
p~increaseSalary(12345.67) /* increase salary            */
p~increaseSalary("abc")    /* provoke an error           */
p~increaseSalary(-1000)    /* decrease salary            */

say ' The end. '~center(50, "-") /* draw a line          */

l~trace("end of program.")

/* ================================================================= */
::class person          /* class "PERSON"                */

/* ----------------------------------------------------- */
::method init           /* method "INIT" (constructor)   */
  expose familyName firstName salary

  l=.logManager~getLogger("rgf.test")     /* get logger  */
  l~trace("method 'init'")

  use arg familyName, firstName, salary

  l~debug("method 'init' - created the following person:" pp(self~string))

  if salary>10000 then  -- warn about something
     l~warn("method 'init' - salary quite high:" salary)
```

```
::method familyName     attribute
::method firstName      attribute
::method salary         attribute

/* ----------------------------------------------------- */
::method string
  expose familyName firstName salary

  .logManager~getLogger("rgf.test")~trace("method 'string'")
  return familyName"," firstName":" salary

/* ----------------------------------------------------- */
::method increaseSalary /* method to increase the salary */
  expose salary

  l=.logManager~getLogger("rgf.test")
  l~trace("method 'increaseSalary'")

  parse arg raise
  l~debug("method 'increaseSalary', received="pp(raise))
  signal on syntax      /* in case arithmetic creates a condition    */
  salary=salary+raise
  l~debug("method 'increaseSalary', new salary="pp(salary))
  return
syntax:
  l~error("method 'increaseSalary', exception has occurred!", condition("O"))

/* ----------------------------------------------------- */
::method uninit
  .logManager~getLogger("rgf.test")~trace("method 'uninit'")
  .logManager~getLogger("rgf.test")~debug("method 'uninit' running for person:" pp(self))
```

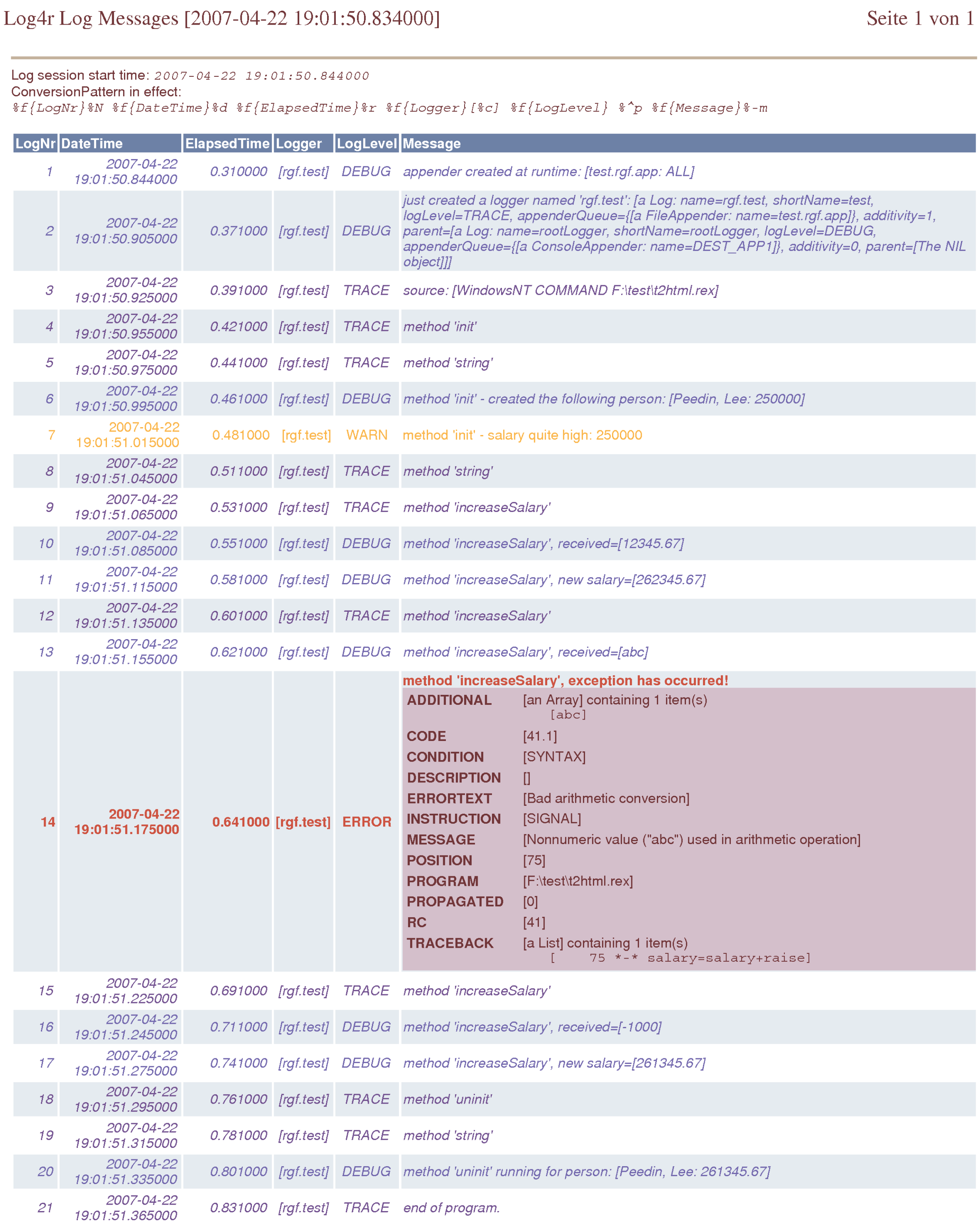
"rexx t1.rex error"

```
----------------- The beginning: -----------------
p=[Peedin, Lee: 250000]
    1: 0.301000 [rgf.test] ERROR - method 'increaseSalary', exception has occurred!
                        ADDITIONAL..[an Array] containing 1 item(s)
                                         --> [abc]
                        CODE........[41.1]
                        CONDITION...[SYNTAX]
                        DESCRIPTION.[]
                        ERRORTEXT...[Bad arithmetic conversion]
                        INSTRUCTION.[SIGNAL]
                        MESSAGE.....[Nonnumeric value ("abc") used in arithmetic operation]
                        POSITION....[63]
                        PROGRAM.....[F:\test\t1.rex]
                        PROPAGATED..[0]
                        RC..........[41]
                        TRACEBACK...[a List] containing 1 item(s)
                                         --> [    63 *-* salary=salary+raise]

-------------------- The end. --------------------
```

```
say ' The beginning: '~center(50, "-") /* draw a line    */  /* "t2.rex" */

call load_log4r         /* load the 'log4r' framework    */

l=.logManager~getLogger("rgf.test") /* get (create?) a logger named 'rgf.test'   */

   /* configure the logger a little bit:  */
app=.FileAppender~new("test.rgf.app")        /* create an appender         */
app~layout=.HTMLLayout~new("test.rgf.layout")/* create a layout and assign it to appender */
app~fileName="test_"date("S")".html"         /* set filename               */
app~append=.false                            /* replace an existing file   */
l~addAppender(app)                           /* add appender to logger     */

.local~test.rgf.logger=l   /* save logger in .local, even easier to refer to  */

l~debug("appender created at runtime:" pp(app~name":" getLogLevelAsString(app~threshold)))

parse arg logLevel
if logLevel="" then l~logLevel="OFF"      /* do not show any log messages     */
               else l~logLevel=logLevel   /* set logLevel to argument's value */




... cut (remaining code the same as in "t1.rex" above ...
```

"rexx t2.rex **trace**"

```
----------------- The beginning: -----------------
    1: 0.310000 [rgf.test] DEBUG - appender created at runtime: [test.rgf.app: ALL]
    2: 0.371000 [rgf.test] DEBUG - just created a logger named 'rgf.test': [a Log: name=rgf.test,
shortName=test, logLevel=TRACE, appenderQueue={[a FileAppender: name=test.rgf.app]}, additivity=1,
parent=[a Log: name=rootLogger, shortName=rootLogger, logLevel=DEBUG, appenderQueue={[a
ConsoleAppender: name=DEST_APP1]}, additivity=0, parent=[The NIL object]]]
    3: 0.391000 [rgf.test] TRACE - source: [WindowsNT COMMAND F:\test\t2html.rex]
    4: 0.421000 [rgf.test] TRACE - method 'init'
    5: 0.441000 [rgf.test] TRACE - method 'string'
    6: 0.461000 [rgf.test] DEBUG - method 'init' - created the following person: [Peedin, Lee: 250000]
    7: 0.481000 [rgf.test] WARN  - method 'init' - salary quite high: 250000
    8: 0.511000 [rgf.test] TRACE - method 'string'
p=[Peedin, Lee: 250000]
    9: 0.531000 [rgf.test] TRACE - method 'increaseSalary'
   10: 0.551000 [rgf.test] DEBUG - method 'increaseSalary', received=[12345.67]
   11: 0.581000 [rgf.test] DEBUG - method 'increaseSalary', new salary=[262345.67]
   12: 0.601000 [rgf.test] TRACE - method 'increaseSalary'
   13: 0.621000 [rgf.test] DEBUG - method 'increaseSalary', received=[abc]
   14: 0.641000 [rgf.test] ERROR - method 'increaseSalary', exception has occurred!
                      ADDITIONAL..[an Array] containing 1 item(s)
                                       --> [abc]
                      CODE........[41.1]
                      CONDITION...[SYNTAX]
                      DESCRIPTION.[]
                      ERRORTEXT...[Bad arithmetic conversion]
                      INSTRUCTION.[SIGNAL]
                      MESSAGE.....[Nonnumeric value ("abc") used in arithmetic operation]
                      POSITION....[75]
                      PROGRAM.....[F:\test\t2html.rex]
                      PROPAGATED..[0]
                      RC..........[41]
                      TRACEBACK...[a List] containing 1 item(s)
                                       --> [    75 *-* salary=salary+raise]

   15: 0.691000 [rgf.test] TRACE - method 'increaseSalary'
   16: 0.711000 [rgf.test] DEBUG - method 'increaseSalary', received=[-1000]
   17: 0.741000 [rgf.test] DEBUG - method 'increaseSalary', new salary=[261345.67]
   18: 0.761000 [rgf.test] TRACE - method 'uninit'
   19: 0.781000 [rgf.test] TRACE - method 'string'
   20: 0.801000 [rgf.test] DEBUG - method 'uninit' running for person: [Peedin, Lee: 261345.67]
-------------------- The end. --------------------
    21: 0.831000 [rgf.test] TRACE - end of program.
```

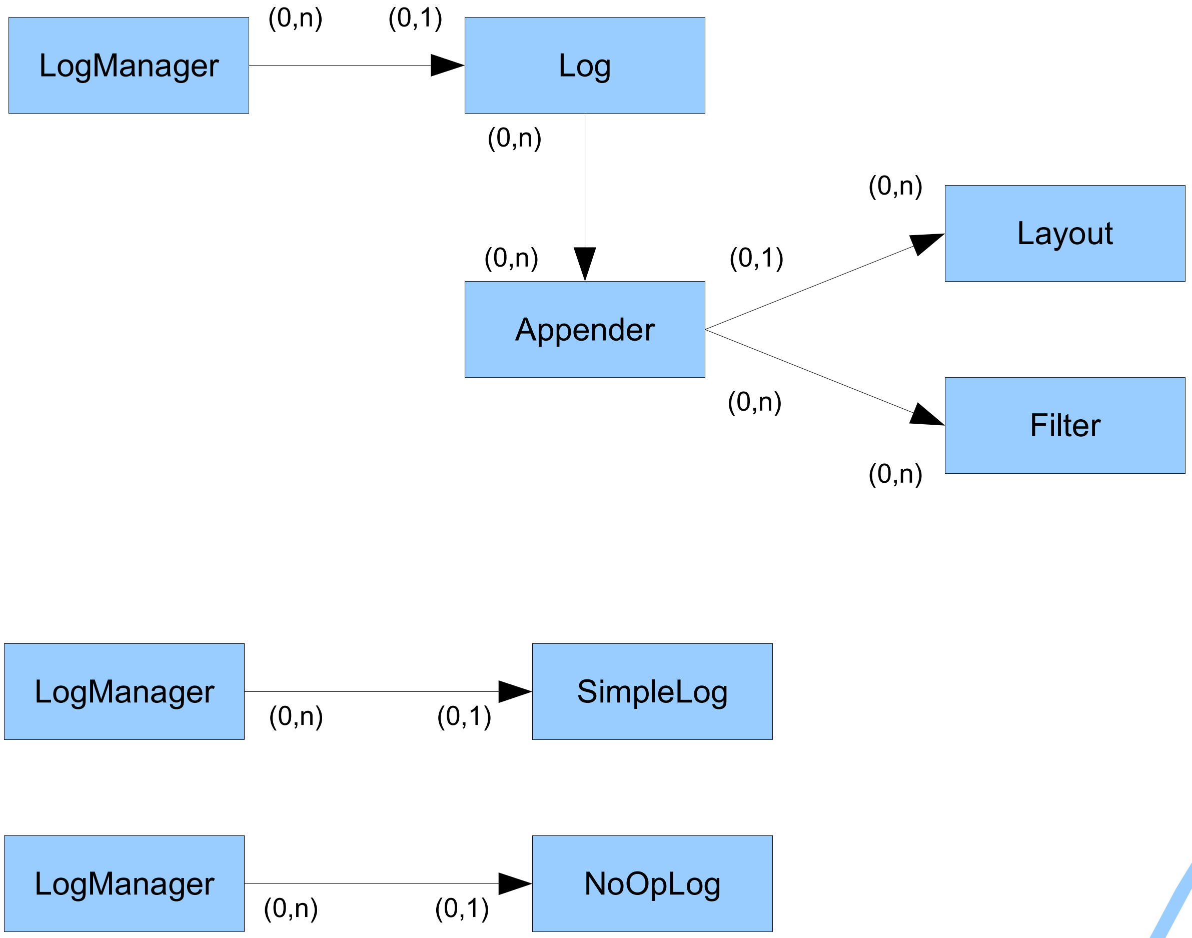


Log4r Log Messages [2007-04-22 19:01:50.834000] Seite 1 von 1

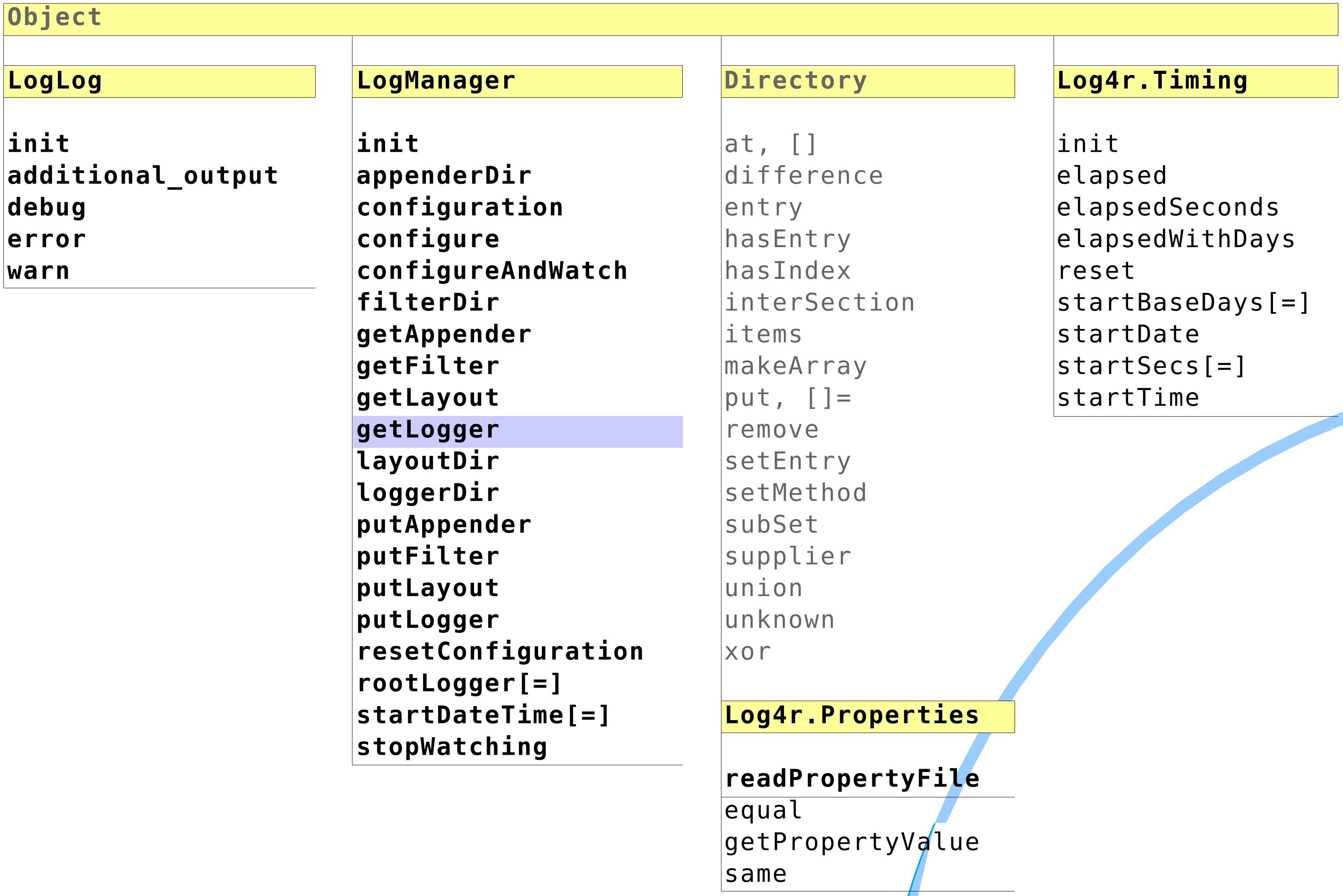
Log session start time: `2007-04-22 19:01:50.844000`
ConversionPattern in effect:
`%f{LogNr}%N %f{DateTime}%d %f{ElapsedTime}%r %f{Logger}[%c] %f{LogLevel} %^p %f{Message}%-m`

LogNr	DateTime	ElapsedTime	Logger	LogLevel	Message
1	2007-04-22 19:01:50.844000	0.310000	[rgf.test]	DEBUG	appender created at runtime: [test.rgf.app: ALL]
2	2007-04-22 19:01:50.905000	0.371000	[rgf.test]	DEBUG	just created a logger named 'rgf.test': [a Log: name=rgf.test, shortName=test, logLevel=TRACE, appenderQueue={[a FileAppender: name=test.rgf.app]}, additivity=1, parent=[a Log: name=rootLogger, shortName=rootLogger, logLevel=DEBUG, appenderQueue={[a ConsoleAppender: name=DEST_APP1]}, additivity=0, parent=[The NIL object]]]
3	2007-04-22 19:01:50.925000	0.391000	[rgf.test]	TRACE	source: [WindowsNT COMMAND F:\test\t2html.rex]
4	2007-04-22 19:01:50.955000	0.421000	[rgf.test]	TRACE	method 'init'
5	2007-04-22 19:01:50.975000	0.441000	[rgf.test]	TRACE	method 'string'
6	2007-04-22 19:01:50.995000	0.461000	[rgf.test]	DEBUG	method 'init' - created the following person: [Peedin, Lee: 250000]
7	2007-04-22 19:01:51.015000	0.481000	[rgf.test]	WARN	method 'init' - salary quite high: 250000
8	2007-04-22 19:01:51.045000	0.511000	[rgf.test]	TRACE	method 'string'
9	2007-04-22 19:01:51.065000	0.531000	[rgf.test]	TRACE	method 'increaseSalary'
10	2007-04-22 19:01:51.085000	0.551000	[rgf.test]	DEBUG	method 'increaseSalary', received=[12345.67]
11	2007-04-22 19:01:51.115000	0.581000	[rgf.test]	DEBUG	method 'increaseSalary', new salary=[262345.67]
12	2007-04-22 19:01:51.135000	0.601000	[rgf.test]	TRACE	method 'increaseSalary'
13	2007-04-22 19:01:51.155000	0.621000	[rgf.test]	DEBUG	method 'increaseSalary', received=[abc]
14	**2007-04-22 19:01:51.175000**	**0.641000**	**[rgf.test]**	**ERROR**	**method 'increaseSalary', exception has occurred!** **ADDITIONAL** [an Array] containing 1 item(s) `[abc]` **CODE** [41.1] **CONDITION** [SYNTAX] **DESCRIPTION** [] **ERRORTEXT** [Bad arithmetic conversion] **INSTRUCTION** [SIGNAL] **MESSAGE** [Nonnumeric value ("abc") used in arithmetic operation] **POSITION** [75] **PROGRAM** [F:\test\t2html.rex] **PROPAGATED** [0] **RC** [41] **TRACEBACK** [a List] containing 1 item(s) `[     75 *-* salary=salary+raise]`
15	2007-04-22 19:01:51.225000	0.691000	[rgf.test]	TRACE	method 'increaseSalary'
16	2007-04-22 19:01:51.245000	0.711000	[rgf.test]	DEBUG	method 'increaseSalary', received=[-1000]
17	2007-04-22 19:01:51.275000	0.741000	[rgf.test]	DEBUG	method 'increaseSalary', new salary=[261345.67]
18	2007-04-22 19:01:51.295000	0.761000	[rgf.test]	TRACE	method 'uninit'
19	2007-04-22 19:01:51.315000	0.781000	[rgf.test]	TRACE	method 'string'
20	2007-04-22 19:01:51.335000	0.801000	[rgf.test]	DEBUG	method 'uninit' running for person: [Peedin, Lee: 261345.67]
21	2007-04-22 19:01:51.365000	0.831000	[rgf.test]	TRACE	end of program.

Architectural Overview of "log4r"


LogManager
(0,n)
(0,1)
Log
(0,n)
(0,n)
Appender
(0,1)
(0,n)
Layout
(0,n)
Filter
(0,n)
LogManager
(0,n)
(0,1)
SimpleLog
LogManager
(0,n)
(0,1)
NoOpLog

The Infrastructural "log4r" Classes



Object

LogLog

- init
- additional_output
- debug
- error
- warn

LogManager

- init
- appenderDir
- configuration
- configure
- configureAndWatch
- filterDir
- getAppender
- getFilter
- getLayout
- getLogger
- layoutDir
- loggerDir
- putAppender
- putFilter
- putLayout
- putLogger
- resetConfiguration
- rootLogger[=]
- startDateTime[=]
- stopWatching

Directory

- at, []
- difference
- entry
- hasEntry
- hasIndex
- interSection
- items
- makeArray
- put, []=
- remove
- setEntry
- setMethod
- subSet
- supplier
- union
- unknown
- xor

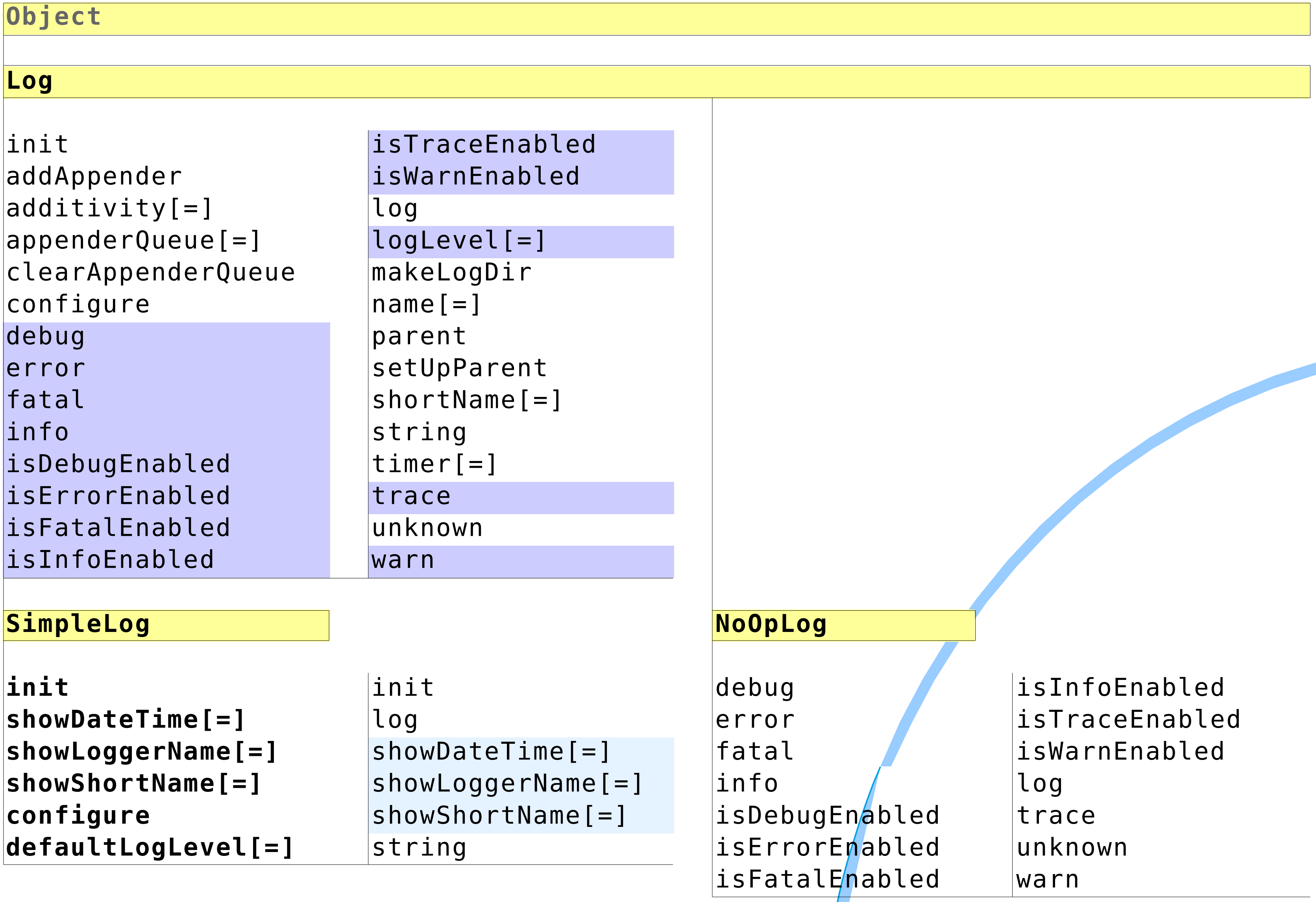
Log4r.Properties

- readPropertyFile
- equal
- getPropertyValue
- same

Log4r.Timing

- init
- elapsed
- elapsedSeconds
- elapsedWithDays
- reset
- startBaseDays[=]
- startDate
- startSecs[=]
- startTime

The Logger Classes



Object

Log

init
addAppender
additivity[=]
appenderQueue[=]
clearAppenderQueue
configure
debug
error
fatal
info
isDebugEnabled
isErrorEnabled
isFatalEnabled
isInfoEnabled

isTraceEnabled
isWarnEnabled
log
logLevel[=]
makeLogDir
name[=]
parent
setUpParent
shortName[=]
string
timer[=]
trace
unknown
warn

SimpleLog

init
showDateTime[=]
showLoggerName[=]
showShortName[=]
configure
defaultLogLevel[=]

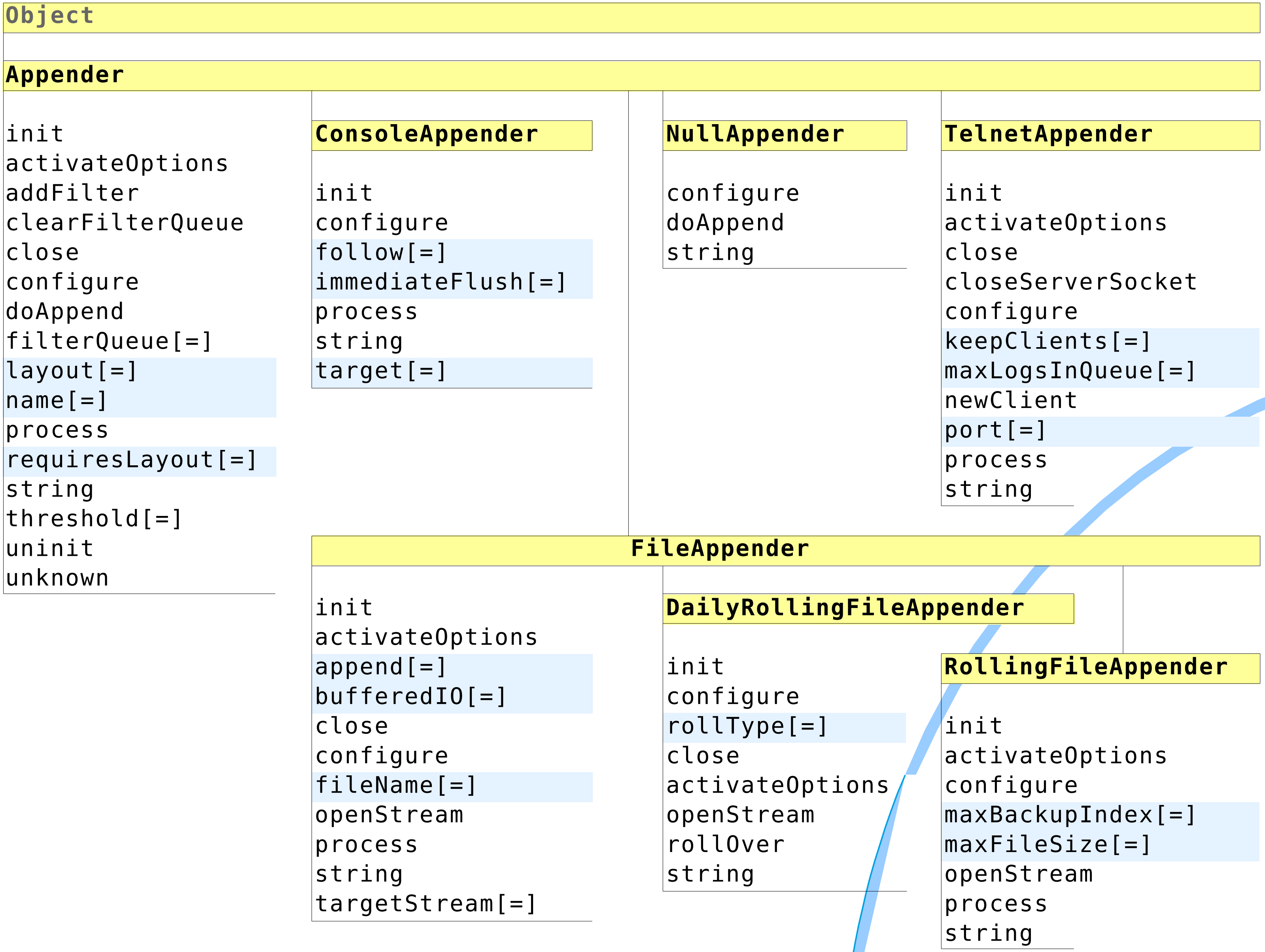
init
log
showDateTime[=]
showLoggerName[=]
showShortName[=]
string

NoOpLog

debug
error
fatal
info
isDebugEnabled
isErrorEnabled
isFatalEnabled

isInfoEnabled
isTraceEnabled
isWarnEnabled
log
trace
unknown
warn

The Appender Classes



Object

Appender

init
activateOptions
addFilter
clearFilterQueue
close
configure
doAppend
filterQueue[=]
layout[=]
name[=]
process
requiresLayout[=]
string
threshold[=]
uninit
unknown

ConsoleAppender

init
configure
follow[=]
immediateFlush[=]
process
string
target[=]

NullAppender

configure
doAppend
string

TelnetAppender

init
activateOptions
close
closeServerSocket
configure
keepClients[=]
maxLogsInQueue[=]
newClient
port[=]
process
string

FileAppender

init
activateOptions
append[=]
bufferedIO[=]
close
configure
fileName[=]
openStream
process
string
targetStream[=]

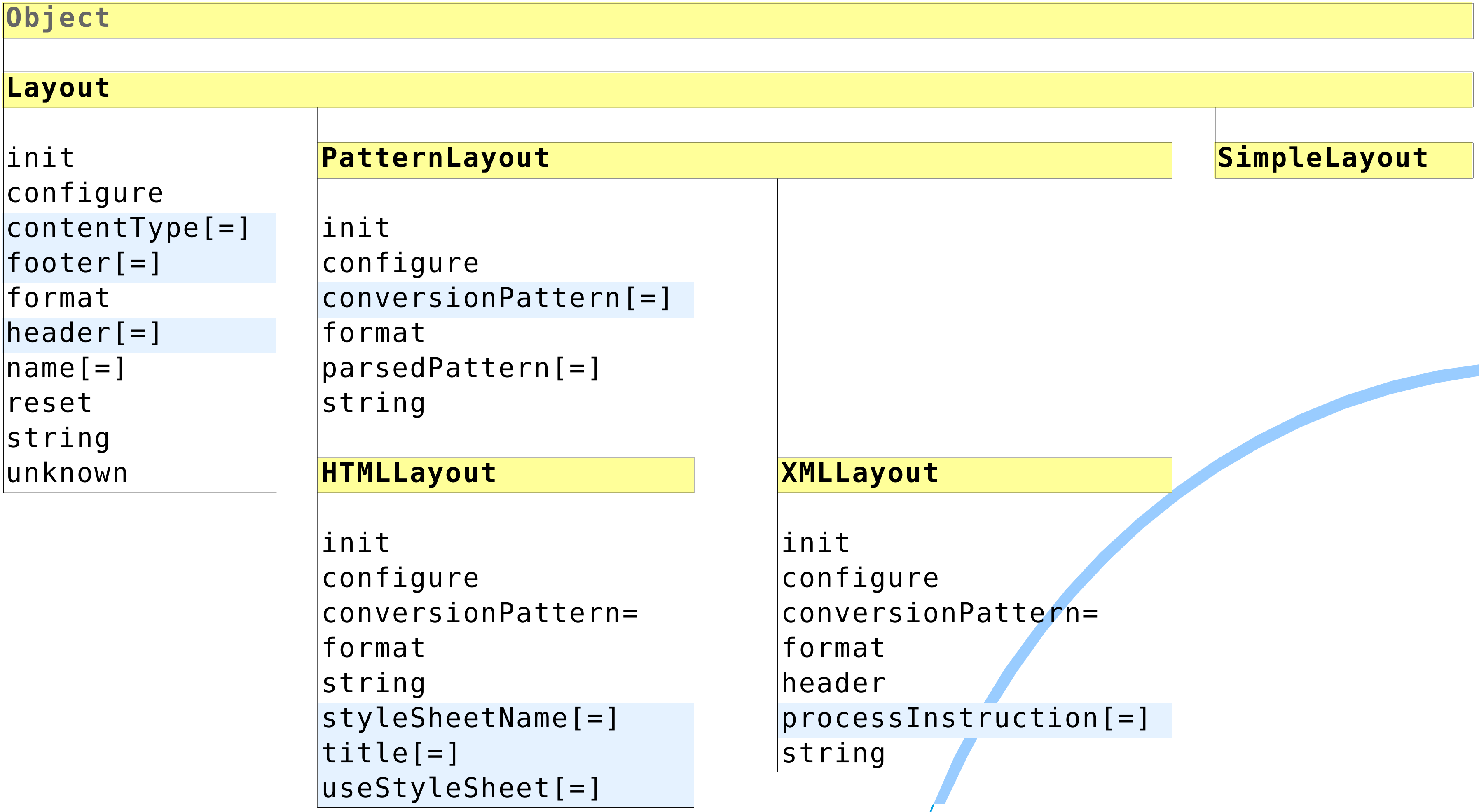
DailyRollingFileAppender

init
configure
rollType[=]
close
activateOptions
openStream
rollOver
string

RollingFileAppender

init
activateOptions
configure
maxBackupIndex[=]
maxFileSize[=]
openStream
process
string

The Layout Classes



Object

Layout

init
configure
contentType[=]
footer[=]
format
header[=]
name[=]
reset
string
unknown

PatternLayout

init
configure
conversionPattern[=]
format
parsedPattern[=]
string

SimpleLayout

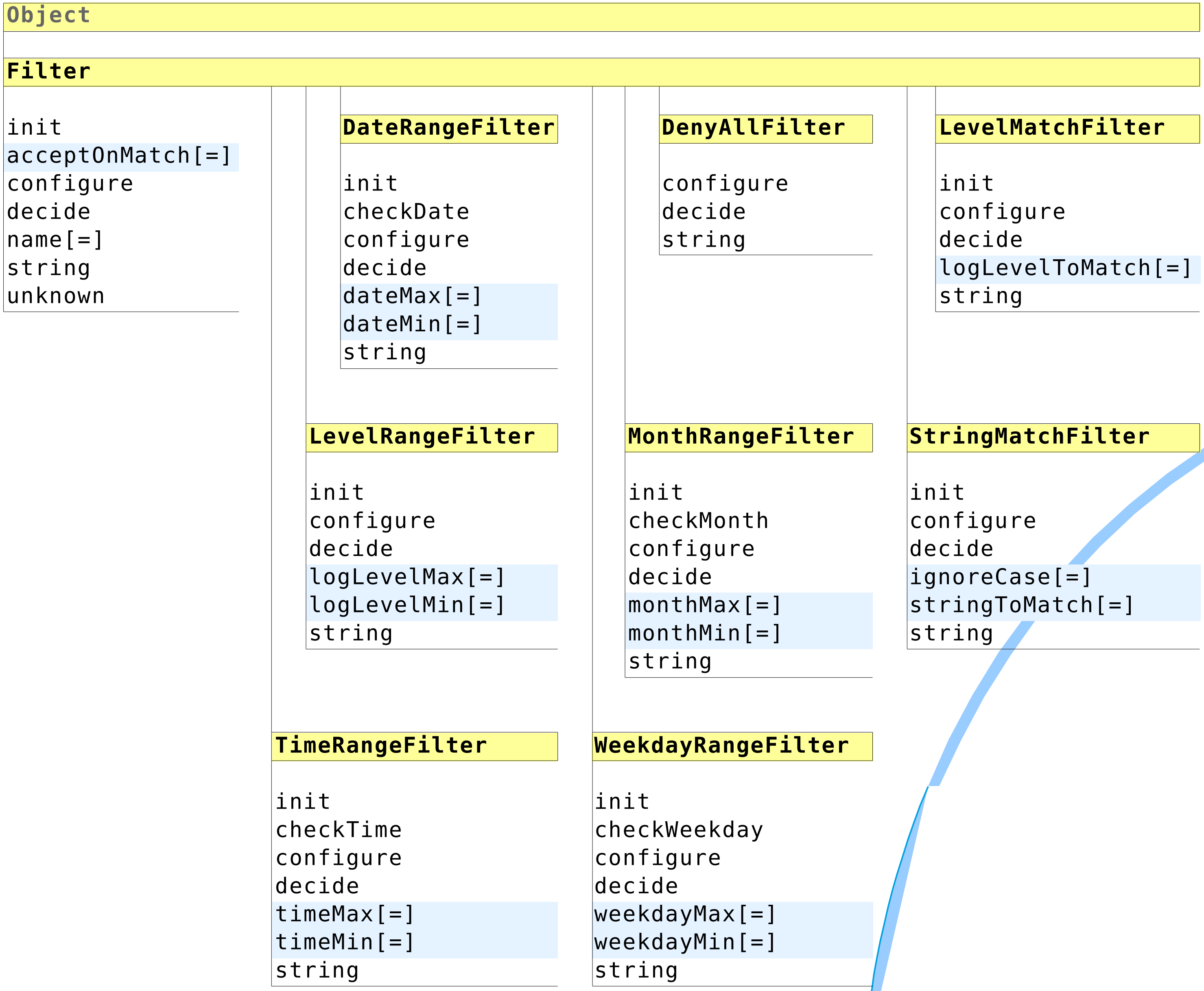
HTMLLayout

init
configure
conversionPattern=
format
string
styleSheetName[=]
title[=]
useStyleSheet[=]

XMLLayout

init
configure
conversionPattern=
format
header
processInstruction[=]
string

The Filter Classes



Object

Filter

init
acceptOnMatch[=]
configure
decide
name[=]
string
unknown

DateRangeFilter

init
checkDate
configure
decide
dateMax[=]
dateMin[=]
string

DenyAllFilter

configure
decide
string

LevelMatchFilter

init
configure
decide
logLevelToMatch[=]
string

LevelRangeFilter

init
configure
decide
logLevelMax[=]
logLevelMin[=]
string

MonthRangeFilter

init
checkMonth
configure
decide
monthMax[=]
monthMin[=]
string

StringMatchFilter

init
configure
decide
ignoreCase[=]
stringToMatch[=]
string

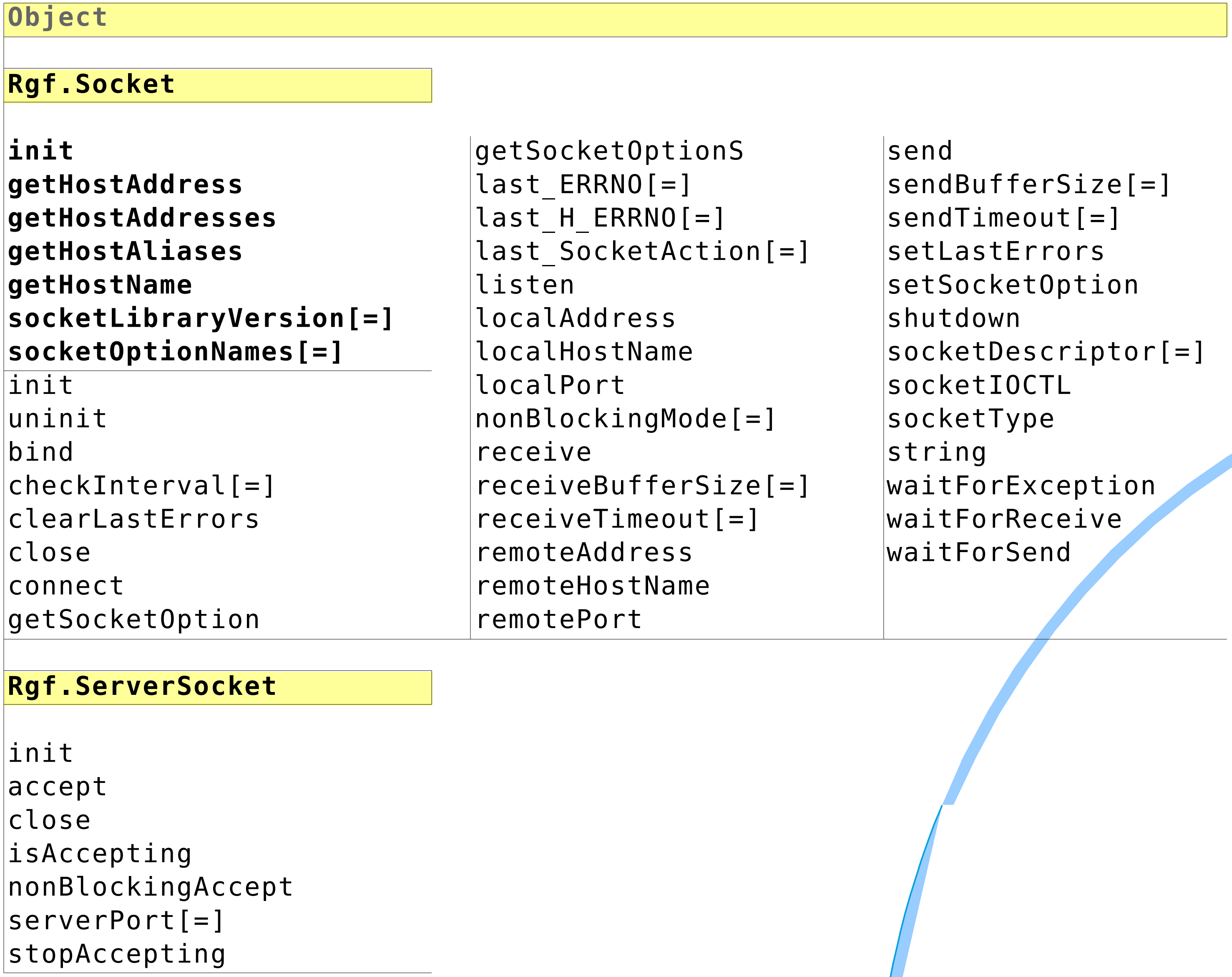
TimeRangeFilter

init
checkTime
configure
decide
timeMax[=]
timeMin[=]
string

WeekdayRangeFilter

init
checkWeekday
configure
decide
weekdayMax[=]
weekdayMin[=]
string

The "rgf.sockets.cls" Class Library (Developed for "TelnetAppender")



Object

Rgf.Socket

init
getHostAddress
getHostAddresses
getHostAliases
getHostName
socketLibraryVersion[=]
socketOptionNames[=]

init
uninit
bind
checkInterval[=]
clearLastErrors
close
connect
getSocketOption
getSocketOptionS
last_ERRNO[=]
last_H_ERRNO[=]
last_SocketAction[=]
listen
localAddress
localHostName
localPort
nonBlockingMode[=]
receive
receiveBufferSize[=]
receiveTimeout[=]
remoteAddress
remoteHostName
remotePort
send
sendBufferSize[=]
sendTimeout[=]
setLastErrors
setSocketOption
shutdown
socketDescriptor[=]
socketIOCTL
socketType
string
waitForException
waitForReceive
waitForSend

Rgf.ServerSocket

init
accept
close
isAccepting
nonBlockingAccept
serverPort[=]
stopAccepting

Configuration of "log4r", 1

- Global configuration settings via environment symbols
  - Entries in .local
    - Globally available to all Rexx programs
  - Configuration by properties file
    - Sound defaults, if no property file is found
  - Changing values at runtime will have immediate effect

Configuration of "log4r", 2

(Global Settings in .local)

Environment Symbol	*Brief Description*
.log4r.config.asyncMode	If set to .true the logger processes the log message asynchroneously. Default value: .false
.log4r.config.configFile	If set, denotes the name of a configuration file.
.log4r.config.configFileWatchInterval	If set to a value greater than 0, then this is the interval time in seconds to check whether the configuration file has been changed, and if so re-read it and apply the changes to the running system.
.log4r.config.disable	Determines the threshold log level in order for all the loggers to process log messages. One of LOWEST (ALL), TRACE, DEBUG, INFO, WARN, ERROR, FATAL, HIGHEST (OFF, NONE). Default value: LOWEST
.log4r.config.disableOverride	If set to .true all log messages will be processed by the loggers, independently of the setting log4r.config.disalbe. Default value: .false
.log4r.config.LoggerFactoryClassName	Set to the name of the logger class that should be used to create loggers, one of Log, SimpleLog, NoOpLog. Default value: NoOpLog
.log4r.config.LogLog.debug	If set to .true the debug log messages from the log4r framework classes are displayed. Default value: .false
.log4r.config.LogLog.quietMode	If set to .true the warn and error in addition to the debug log messages from the log4r framework classes are not displayed. Default value: .false

Configuration of "log4r", 3

- Usually, **all** configuration done via a text file
  - "Property" files
    - Standard filenames
      - "log4r.properties"
      - "simplelog4r.properties"
    - Possible, to name property files to anything else
  - Property file layout simple
    - key=value

Configuration of "log4r", 4

(Snippet from a Property File)

```
# ---------------------- logger named "rootLogger"
; default: send logs at level "DEBUG" or higher to to appender named "DEST_APP1"
log4r.logger.rootLogger = debug, dest_app1

# ---------------------- logger named "rgf.sockets"
; default: send logs at level "DEBUG" or higher to to appender named "RGF_APP2"
log4r.logger.RGF.SOCKETS = debug, rgf_app2

; ---------------------- appender named "DEST_APP1"
log4r.appender.DEST_APP1                = ConsoleAppender
log4r.appender.dest_app1.ImmediateFlush = true
log4r.appender.dest_app1.layout         = pat_layout1
log4r.appender.dest_app1.Target         = stderr

; ---------------------- appender named "RGF_APP2"
log4r.appender.RGF_APP2           = FileAppender
log4r.appender.rgf_app2.filter     = weekDays
log4r.appender.rgf_app2.layout     = html_layout1
log4r.appender.rgf_app2.fileName   = rgf_app2_appender.html
log4r.appender.rgf_app2.append     = false
log4r.appender.rgf_app2.bufferedIO = true

; ---------------------- layout named "PAT_LAYOUT1"
log4r.layout.PAT_LAYOUT1                   =PatternLayout
log4r.layout.pat_layout1.conversionPattern =%5N: %r [%c] %-5p - %m%n

; ---------------------- layout named "HTML_LAYOUT1"
log4r.layout.HTML_LAYOUT1                   =HTMLLayout
log4r.layout.html_layout2.title             =Standard Infos (HTML_LAYOUT1)
log4r.layout.html_layout1.conversionPattern =%5N: %r [%c] %-5p - %m%n

; ---------------------- filter named "WEEKDAYS"
log4r.filter.WEEKDAYS              = WeekdayRangeFilter
log4r.filter.weekdays.weekdayMin    = Monday
log4r.filter.weekdays.weekdayMax    = Friday
log4r.filter.weekdays.acceptOnMatch = true
```

Configuration of "log4r", 5

- Configuration also possible at runtime
  - ooRexx programmer needs to know the classes and available options

```
say ' The beginning: '~center(50, "-") /* draw a line    */  /* "t2.rex" */

call load_log4r         /* load the 'log4r' framework    */

l=.logManager~getLogger("rgf.test") /* get (create?) a logger named 'rgf.test'   */

   /* configure the logger a little bit:  */
app=.FileAppender~new("test.rgf.app")        /* create an appender         */
app~layout=.HTMLLayout~new("test.rgf.layout")/* create a layout and assign it to appender */
app~fileName="test_"date("S")".html"         /* set filename               */
app~append=.false                            /* replace an existing file   */
l~addAppender(app)                           /* add appender to logger     */

.local~test.rgf.logger=l   /* save logger in .local, even easier to refer to  */

l~debug("appender created at runtime:" pp(app~name":" getLogLevelAsString(app~threshold)))

parse arg logLevel
if logLevel="" then l~logLevel="OFF"      /* do not show any log messages    */
               else l~logLevel=logLevel   /* set logLevel to argument's value */

... cut (remaining code the same as in "t1.rex" above ...
```

Roundup and Outlook

- Modelled after Apache's "log4j"
- "log4r" thoroughly tested using
  - ooRexxUnit
  - Integration tests
- Adding new appenders, layouts, filters
  - E.g. e-mail or instant messaging appender, sending automatically e-mail for certain log messages (e.g. "WARN", "ERROR", "FATAL")
- If the ooRexx community adopts it, then even possible to turn it into an Apache project!