**Purpose:** A function used to map scallop fishing areas in R. Note this is DK revised version. Strata, bathymetry and plot regions can all be added to ScallopMap either collectively or individually, there are a number of options within each of these outlined below.

* Note that you do not want to have all of the plot options = T as the plot gets very busy. This is especially so with the Strata and Management boundary labels turned on.
* Note that the behaviour of ScallopMap and images added to ScallopMap differ depending on the plotting device. For example if you use “addPolys” to add polygons to ScallopMap after the initial plot has been made the polygons “may” show up as either transparent or opaque depending on the plotting device used. Using the R default plot window results in an opaque polygon, while running windows() will result in the polygons being transparent. In general I suggest not adding Polygons where possible but using the AddLines command.

**Version Control:** I see there are at least 2 versions of ScallopMap, 1 for inshore 1 for offshore. Differences in area names between inshore and offshore. This function should supersede them both.

**Function Arguments Summary**

1. **area**: What area to you want to print, default is 'custom' where xlim & ylim are specified or select from area list below
2. **plot.strata**: Plot the strata? (T/F) default is False
3. **plot.bathy**: Plot the Bathymetry? (T/F) default is False
4. **plot.boundries**: Plot the management area boundaries? (T/F) default is False
5. **direct**: the directory to choose. Default is "Y:/Offshore scallop/Assessment/Assessment\_fns/"

If running locally copy the "Maps" folder, and the "data/maps" folder to a common location then change direct to that location. For example, if you copy these 2 folders to d:/maps, set direct = d:/maps and this should work (I hope). The folder structure in this case would be d:/maps/maps which contains our functions & d:/maps/data/… which contains subfolders with the map data.

These options are used when **plot.strata** = T

1. **strata**: What Strata to plot. Options are "inshore", "offshore", and "all", default is all
2. **un**: To obtain strata there is a SQL call, default is “pwd.ID” please set this up in your .rProfile so it does not need added directly into the function
3. **pw**: Your SQL password. default is "un.ID" please set this up in your .rProfile so it does not need added directly into the function
4. **db.con**: Database to connect to. Default is "ptran"
5. **strata.colors**: The color palette for the strata. default = pastel.colors(64,seed=2)
6. **strata.cex**: Size of the strata labels, strata labels only printed if "strata = inshore". Default is 0.5

These options are used when **plot.bathy** = T

1. **isobath**: The isobaths to plot. Default is 50,100,150,200. NB "bathy.source=quick" will only work with combinations of these 4 isobaths
2. **bathy.source**: Bathymetry sources, options are "CHS", "usgs", "quick", "topex". Default is "quick". NB: To use CHS you need to have access to SQL database (see documenation for get.bathy.r function)
3. **bathcol**: Color of the bathymetry lines. Default is "blue".

These options are used when **plot.boundries** = T

1. **boundries**: Which management boundaries are to be plotted? Options are 'inshore', "offshore", "all". Default is "all". NB: selecting "inshore" will plot high detailed inshore boundaries.
2. **label.boundries**: Plot the Management Boundary names? (T/F) default is F
3. **offshore.names**: Plot the common names for the offshore management areas? (T/F) default is F
4. **bound.color**: Plot the management boundaries in color? (T/F) default is F
5. **manage.colors**: Colors for management boundaries if "bound.color=T". Default is pastel.colors(n=64,seed=2)
6. **manage.cex**: Size of the text if "label.boundries = T". Default is 0.7

These options are for adding additional layers

1. **points.lst** = points to overlay on map in PBSmapping format - list with 2 elements: 1st element is eventSet (EID, POS, X, Y), 2nd element is eventData (EID, pch, col, etc.)
2. **lines.lst** = lines to overlay on map in PBSmapping format - list with 2 elements: 1st element is polySet (PID, SID, POS, X, Y), 2nd element is polyData (PID, SID, lty, col, etc.)
3. **poly.lst** = polygons to overlay on map in PBSmapping format - list with 2 elements: 1st element is polySet (PID, SID, POS, X, Y), 2nd element is polyData (PID, SID, border, col, etc.)
4. **contours** = plots overlaping polygons as contours (same format as poly.lst)
5. **image.lst** = image to overlay on map - list with 3 elements (x, y, z), 'bathymetry' produces image from bathymetry data
6. **color.fun** = color function for image
7. **zlim**: zlim for image

These options are used for adding land and other offshore elements

1. **shore** = shoreline detail ('marHR' = very fine martimes only, 'nwatlHR' = fine NW Atlantic, 'nwatlMR' = medium NW Atlantic, 'worldLR' = coarse world)
2. **plot.land**: Plot the land. (T/F) default is T
3. **land.col**: Color of the land. default is 'wheat',
4. **lol**: Land outline (T/F) F = default (outline is in same color as the bathymetry)
5. **stippling**: Plot stippling (dots) on the land. (T/F) default is F
6. **nafo**: If plotting NAFO regions which ones do you plot. Default is NULL, there are 42 regions to choose from and any combination of the 42 is acceptable.
7. **nafo.bord**: Plot the NAFO borders. (T/F) default is F
8. **nafo.lab**: Plot the NAFO area names. (T/F) default is F
9. **banks**: Plot the outline of the offshore banks. (T/F) default is F
10. **plot.EEZ**: Plot the Exclusive Economic Zone. (T/F) default is T
11. **grid**: Add a grid to the plot. Default is NULL (no grid), grid size is in degrees.

These Options are used for changing the appearance of the plot and for Saving the plot automatically.

1. **title**: Plot title. Default is empty
2. **cex**: Magnification for plot text/symbols. Default is 1
3. **xlab**: Label for x axis. Default is empty and uses PlotMap's default
4. **ylab**: Label for y axis. Default is empty and uses PlotMap's default
5. **output.pdf**: Save a copy of the plot as a pdf file. (T/F) default is F
6. **output.png**: Save a copy of the plot as a png image. (T/F) default is F NB: if this and "output.pdf" are both TRUE only a pdf file is produced.
7. **plot.directory**: Directory of saved plot. Default is current working directory.
8. **filename**: Filename of saved plot. Default is "testplot"
9. **width**: Width (inches) of the plot image to be saved. Default is 8
10. **height**: Height (inches) of the plot image to be saved. Default is 8

I have divided the ScallopMap function into discrete “Sections” it might be over the top but something like this will really help with clarity of function/troubleshooting/edited/ revisions…

**Section 1**

Within this section the offshore and onshore areas are delineated for the plot. At the end the initial plot call is made.

***Note(s)***

* **Area**: Area is used to define the region we are interested in plotting, it defines ylim and xlim for the plot. The units appear to be latitude and longitude using decimal degree notation.
  1. Custom: define the plot area of interest. When custom you have to specify ylim and xlim in the call to the function.
  2. Predefined co-ordinates used to bound the plot area, do not use ylim or xlim in your function call if using these pre-defined regions. Various areas are also included here, at time of writing these included:

*Offshore*

* + NL, offshore, SS, ESS, WSS, BBn, BBs, BB, GB, GBb, Ger, Sab, SPB

*Inshore*

* + sfa29, gm, inshore, bof, upper, mid, spa3, spa4, spa1,spa6
* **Shore:** This is used to define the shore line boundaries for the plot. There are 4 possibilities currently
  + - 'marHR' = very fine martimes only (…/Maps/approved/Coastline/martimesHIGH.ll)
    - 'nwatlHR' = fine NW Atlantic (…/Maps/ approved/Coastline/nwatlHR.ll)
    - 'nwatlMR' = medium NW Atlantic (…/Maps/ approved/Coastline/nwatlMR.ll)
    - 'worldLR' = coarse world (…/Maps/ approved/Coastline/worldLR.ll)

**Section 2**

Here the various different optional layers are added to the plot. The first two calls in this section are to add NAFO and banks respectively. Three of the optional image overlays are plotted at the end, **poly.lst, contours**, and **image.lst**. Also a special line to plot St. Pierre Bank region has been added here.

Between these calls all of the outside function calls are made in this section. A function call is made to **managePlot.r**, this plots the management boundaries and prepares the management labels if they are to be plotted. A second function call is to **strataPlot.r** if we want to add the strata to plot. Finally a call to **the bathyPlot.r** function allows for the bathymetry to be added (if using “CHS” this calls the function **get.bathy.r**). At the end of this section the land is plotted if requested. This overlays the land nicely on all the previous layers

***Note(s)***

1. **nafo**: Data from (…/Maps/approved/Other\_Borders/nafo.csv). Plots 'all' or specified NAFO areas “Null” is default which plots nothing. There are currently 42 possible NAFO areas to choose from (any combination of the 42 is acceptable):
   1. 4VN 4VS 4VSB 4VSC 4VSE 4VSV 4W 4WD 4WE 4WF 4WG 4WH 4WJ 4WK 4WL 4WM 4W 4X 4XL 4XM 4XN 4XO 4XP 4XQ 4XR 4XS 4XX 5Y 5YB 5YC 5YD 5YE 5YF 5ZC 5ZE 5ZEG 5ZEH 5ZEJ 5ZEM 5ZEN 5ZEO 5ZWQ
2. **bathy.source:** Specifies bathymetry currently 4 options

* CHS – Canadian Hydrological survey, this requires our in house **get.bathy.r** function to pull data from mflib.chsdem\_ind SQL database. You will get an error unless you have access to this database and you have setup the database connection correctly for get.bathy.r.
* topex – Data from flat file ( …/Maps/approved/bathymetry/topex/AtlCanbathy.xyz). I belive topex data originally comes from here. http://topex.ucsd.edu/cgi-bin/get\_data.cgi.
* USGS - Data from flat file (… /Maps/approve/usgs/bathy15m.csv).
* quick – Several flat files make up this option and it is divided into inshore and offshore, this uses ‘**boundries’** and ‘**isobath’.**
  + Offshore - …/Maps/approved/bathymetry/quick/offshore/d",isobath[i],".ll"
  + Inshore - … /Maps/ approved/bathymetry/quick/inshore/d",isobath[i],".ll"

1. **Isobath:** Specify the depths of the isobaths, defaults to (50,100,150,200). Note only combinations of these 4 values with a **bathy.source** = ‘quick’ will work, ***any other combination will look for a file that is non-existent*** **and will cause an error like this**

Error in file(file, "rt") : cannot open the connection

In addition: Warning message:

In file(file, "rt") :

cannot open file 'c:/Maps/gmtBase/bathy/dXX.ll': No such file or directory

**Section 3**

Here we add the final labels and plot elements to make the plot look pretty. The EEZ, other land element, grid, points, lines and title are all added here.

***Notes(s)***

* The label calls for strata and boundaries rely on objects returned from their respective functions. This way of plotting the labels is a bit clunky but ensures they are plotted nicely over the other layers irrespective of plotting device used.

**Function Index**

!is.null:

addLabels

addLines

addPoints

addPolys

addStipples

attr

calcCentroid

ceiling

contourLines

convCP

duplicated

for

function

If

Image

joinPolys

length

lines

makeGrid

makeTopography

merge

missing

names

na.omit

paste

plotMap

polygon

read.csv

read.table

rep

require

seq

stop

subset

unique

Major Additions and changes to *ScallopMap*

1. Option to not plot inshore/offshore management boundaries. If plotting there are several options you can choose (see above) to customize the figure.
   * **plot.boundries:** Plot the management boundaries. Options are T or F. If =T area boundaries are plotted, if = F they are not. Default is (T)
2. Option to plot bathymetry and strata in a similar fashion, customizations possible.
3. Strata for inshore and offshore now can be added directly as a function call and the data is pulled directly from SQL database.
4. 3 new sub functions added managePlot, strataPlot, and bathyPlot, each can produce a stand alone plot, see functions for specific details.
5. I have made the land plot optional, removing the land.twice argument

* **plot.land**: Plot the land. Options are T or F, if F no land is plotted. Default is T.

1. Do we want to plot the EEZ, might as well make it an option!

* **plot.EEZ**: Plot the EEZ. Option are T or F, if F no EEZ boundary is plot. Default is T

1. Let’s add some flexibility to NAFO as we may not want both borders and labels. Used only if **nafo** is set to something other than NULL.

* **nafo.lab**: Plot the names of the NAFO regions. If F no names are added. Default is T
* **nafo.bord**: Plot the border of the NAFO regions. IF F no borders are added. Default is T.

1. Flexibility to save as a pdf, png, or if blank just plot to a windows device.
2. Direct call to SQL db is an option for bathymetry and strata if using either need to input your SQL user credentials.
3. The addition of layers has been retained as per original function so this should work as per usual.

Arguments removed:

* plot.lines (becomes plot.boundries)
* land.twice (becomes plot.land)

**Notes:**

Banks not working with default banks.xy file. IT APPEARS THAT BANKS.xy has PID 1 repeated twice thus this is causing the problem. I have deleted 1 set of PID=1 and it seems to work (FILE in …/Data/Maps/approved/Other\_borders/)

I have removed the plot.lines argument, I replaced it with plot.boundries it is essentially the same thing but I used it to make somewhat cleaner code and I believe a more intuitive name

I have also rearranged the order of the functions to make more sense and to make it look better.