

## Useful tools developed or used by WGIPEM members, of potential interest for a wider community.

---

### **Atlantistools R package: to more easily visualize and explore output from Atlantis models (available on CRAN)**

Atlantis is an end-to-end marine ecosystem modelling framework. It was originally developed in Australia by E.A. Fulton, A.D.M. Smith and D.C. Smith (2007) and has since been adopted in many marine ecosystems around the world. The output of an Atlantis simulation is stored in various file formats like .netcdf and .txt and different output structures are used for the output variables like e.g. productivity or biomass. However, there was no unified approach to analyse the complex output of an Atlantis simulation, thus an R package was developed by a WGIPEM member to fill this gap. The "Atlantistools" package can be used to convert the different output types to a unified format according to the "tidy-data" approach by H. Wickham (2014). Additionally, ecological metrics like for example spatial overlap of predator and prey or consumption can be calculated and visualized to improve model calibration and benchmarking. Another key feature of "Atlantistools" is the automated generation and perturbation of model parameter files which is essential to apply a sensitivity analysis. The unified data structure offers the possibility to easily share model output with each other and perform inter- and intraspecific model comparisons. "Atlantistools" is freely available on CRAN (version 0.4.2, <https://CRAN.R-project.org/package=atlantistools>). The package is developed collaboratively using the Internet hosting service Github (<https://github.com/alketh/atlantistools>).

### **Shiny R package: a user-friendly interface to communicate model outputs (contact person: Sigrid Lehuta and Klaus Huebert)**

Some WGIPEM members use the user-friendly interface provided through the shiny R package to communicate model results and performance to stakeholders. Stakeholders often find model output graphs too complex and advise to present them sequentially so they progressively get familiar to their structure and meaning. They also often ask for new graphs at different scales that are difficult to provide in the course of a meeting without coming back to the code. Shiny is an R library that allows to interface R code and create web applications (<https://shiny.rstudio.com/>). It was positively tested by a WGIPEM member in order to create an interface that facilitates the exploration of a fleet dynamics model results in various dimensions and at different scales without a line of code. The interface is intended to be used within meetings with stakeholders and made available online for stakeholders to play with. Such kind of web applications could be used to make model validation results available online and increase the transparency of model performance.

Examples can be found here:

- [Shiny app to explore the impact of the landing obligation using the ISIS-Fish model:](http://sirs.agrocampus-ouest.fr/discardless_app/app10/) The app offers to explore and compare the results of 4 alternative scenarios of management of the French demersal mixed fishery in the Eastern English Channel. The scenarios aimed at comparing Discard As Usual situations (DAU) for two assumed discarding behavior: observed discard rates (DAU\_discardRates) and strict respect of the MLS (no highgrading)(DAU-MLS) and the Landing Obligation (LO) with (LO\_tacAv) and without TAC uplifts (LO-tacUplift). Drop-down menu offer to users to select one or several outputs between biomass, catch, revenues or discards for eight populations of target species at different time steps of the simulation or cumulated over time. They can also chose to investigate absolute values of choose a reference scenarios compared to which, relative values are displayed.

[http://sirs.agrocampus-ouest.fr/discardless\\_app/app10/](http://sirs.agrocampus-ouest.fr/discardless_app/app10/)

- **Shiny app for the Quirks larval fish model:** Estimated larval fish growth rates are useful for evaluating how well fish might thrive in different geographic ranges or climates, because fish are often particularly sensitive to their environment as larvae. This tool allows users to simulate growth rates of different types of fish larvae under a wide range of environmental conditions. It is a simple web-interface for Quirks, a peer reviewed, published model of larval fish foraging behaviour and growth physiology. Version 1.0 of the Shiny app ([https://klausbh.shinyapps.io/shiny\\_quirks/](https://klausbh.shinyapps.io/shiny_quirks/)) uses published Quirks open source code (<https://sourceforge.net/projects/larvalfishquirks/>) and larval types (Huebert and Peck, 2014, doi:10.1371/journal.pone.0098205). Newer development versions (<http://shiny.hpl.umces.edu:3838/>) allow users to define custom larval fish types as well. Quirks and the Shiny app were developed by a WGIPEM member to facilitate ecologically meaningful comparisons between different fish species and environments.

### **Zooplankton Model Library (ZoopLib) (contact person: Rubao Ji)**

The Zooplankton Model Library (ZoopLib) is an open repository of code, commentary, and references to primary literature for zooplankton models. The objectives are: (1) to provide a single portal or jumping off point for accessing the wide range of zooplankton models currently in use; (2) to provide a collaborative forum for discussion and improvement of models; (3) to facilitate exchange of models and standardization of modelling practices. The repository is hosted by and has been archived at Google:

<https://code.google.com/archive/p/zooplib/wikis/Introduction.wiki>

Plans for version 2.0 include the following components: (1) Migrate existing content to GitHub; (2) Include executable example modules, rendered using Jupyter note-book; (3) Outreach to build out repository; (4) Meta-analysis of zooplankton models; (5) Explore links with other modelling communities' repositories

### **Ecosystem Code Generation Tool (contact person: Hagen Radtke)**

The ecosystem Code Generation Tool (CGT - <https://ergom.net/index.php/code-generation-tool.html>) is a tool to create ecosystem model code from two ingredients: (1) a formal description of the ecosystem tracers and processes in a list of text files (2) a set of "code templates" for the host model. The tool then extracts the information from the text files and fills the code templates to create your model code.