

# Transparent Assessment Framework

## Project proposal seminar

Arni Magnusson  
Colin Millar

ICES, 25 August 2016



# Aim

*To implement a system that will give stock assessors, reviewers and the secretariat, the tools to routinely document data, methods and disseminated results used in an ICES assessment, so that they can be easily replicated at a future date and also be re-run the following years with the addition of corresponding additional data.*

# Aim

*To implement a framework to organize **data**, **methods**, and **results** used in an ICES assessment, so they are easy to **find** and **rerun** later with new data.*

# Current workflow

# Steps

## 1. **Work** within institutes

*prepare data, prepare model, run model(s)*

# Steps

1. **Work** within institutes

*prepare data, prepare model, run model(s)*

2. **EG** meeting

*finalize model run, write report + draft advice sheet*

3. **ADG** meeting

*finalize advice sheet*

4. **ACOM** meeting

*finalize advice*

# Components

- **Chapter** in report  
*data, results*
- **Stock annex** in report  
*data-methods, assmt-methods*
- **Data folder** on Sharepoint  
*data, assmt-methods, results*
- **Advice sheet** via Stock Assessment Graphs  
*results*
- **GitHub** EG repository  
*data, assmt-methods, results*

# Input data

For surveys, catch, weights, maturity, etc.:

1. Data **preparation**

*filter, calculate, aggregate*

2. Data in **model-specific** format

*model.dat — or R objects*

3. Generate **crosstab** and **annual** tables

*report tables*



# Model run

A model run requires:

1. Compiled **model**

*model.exe — or model.dll*

2. **Data** in model-specific format

*model.dat — or R objects*

3. **Configurations**

*model.cfg — or R objects*

# Output

After running the model:

1. **Model-specific** output  
*results.dat — or R objects*
2. **Postprocess** results  
*extract, calculate, aggregate*
3. Generate **crosstab** and **annual** tables  
*report tables, SAG database*

# Problems

Current workflow is not fully:

- **Quality controlled:** input data can bypass ICES DBs  
*e.g. WoS Cod, Haddock and Whiting*
- **Repeatable:** sharepoint folder may contain an old model run  
*all components of model may not be present*
- **Encapsulated:** generally only input files are stored so  
*links with databases are broken*

## Problems (cont.)

Difficult / time-consuming to:

- Find data and results from a given assessment
- Rerun model with different data or assumptions
- Prepare and run an update assessment

## Problems (cont.)

The current workflow does not help people to:

- Construct **model input** from various ICES data services
- Streamline the flow from **model output** to the ICES SAG database
- See **changes** in model setup and data between years
- Use reproducible research to strengthen **institutional memory**
- Make every step of the analysis **transparent on the web**
- Access data and results from all stocks for **big-picture research**

# New framework

# Objectives

The new framework should be:

- **Quality controlled:** all data must pass through DATSU (or equiv.)
- **Repeatable:** data and assessment workflow is recorded and is functional
- **Encapsulated:** input DBs are linked to SAG with no user dependent steps

## Objectives (cont.)

Make it easier to:

- **Find** data and results from a given assessment
- **Rerun** model with different data or assumptions
- Prepare and run an **update** assessment



## Objectives (cont.)

Help people to:

- Construct **model input** from various ICES data services
- Streamline the flow from **model output** to the ICES SAG database
- See **changes** in model setup and data between years
- Use reproducible research to strengthen **institutional memory**
- Make every step of the analysis **transparent on the web**
- Access data and results from all stocks for **big-picture research**

## Desired properties

It would be great if we can:

- Use **GitHub** to track changes in user scripts and undo mistakes
- Use **web services** to access commercial catch-at-age data
- Access **CSV files** from web services (20–50 times faster than XML)
- Streamline the flow between TAF and **stockassessment.org**
- Rerun models **many years later**

# System workflow (core)

# Output



**input.R**  
+ data  
=> model input files



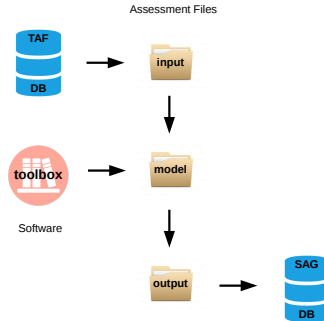
**model.R**  
+ executable  
+ model input files  
+ model config files  
=> model output files



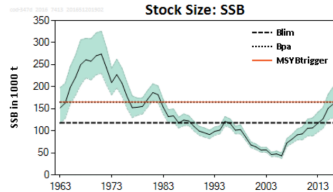
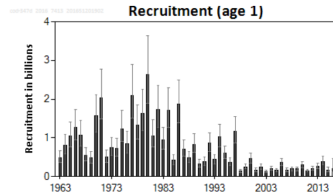
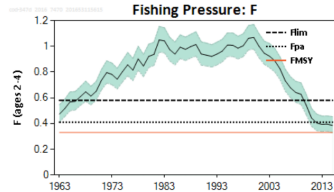
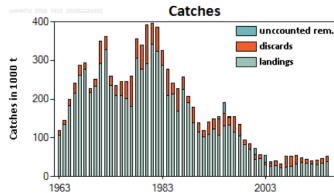
**output.R**  
+ model output files  
=> standard output files

# System workflow (databases)

# Databases for input and output



# Stock Assessment Graphs



# TAF database

## Purpose

store final input data used in ICES stock assessments

## Characteristics

stored in few tables, not big data

## Table structure (first ideas)

data by year and age	AssmtYear, Stock, Year, Age, Quantity, Value
----------------------	--

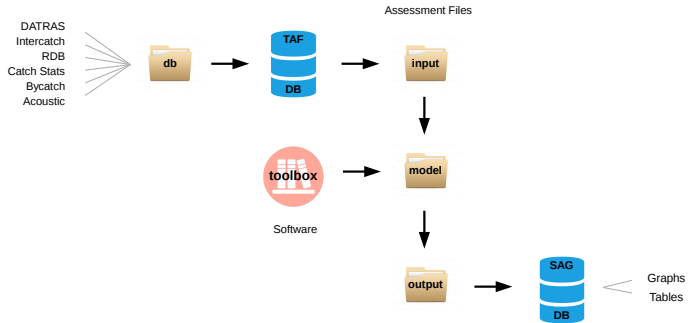
data by year	AssmtYear, Stock, Year, Quantity, Value
--------------	---

data by age	AssmtYear, Stock, Age, Quantity, Value
-------------	--

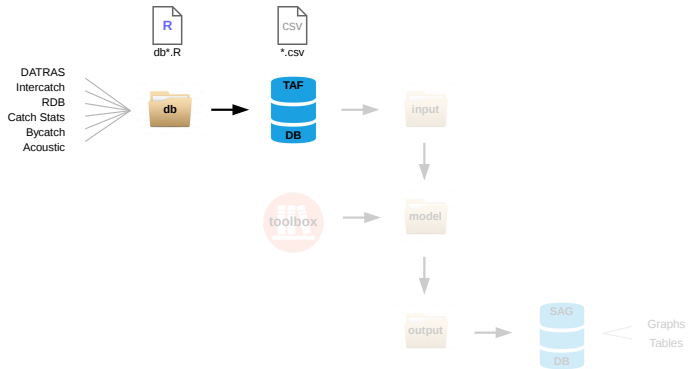
and then there is length data...



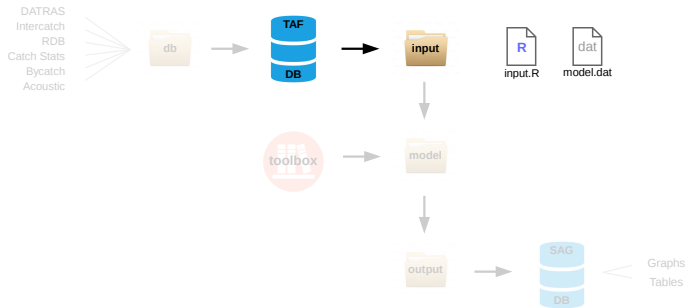
# System workflow



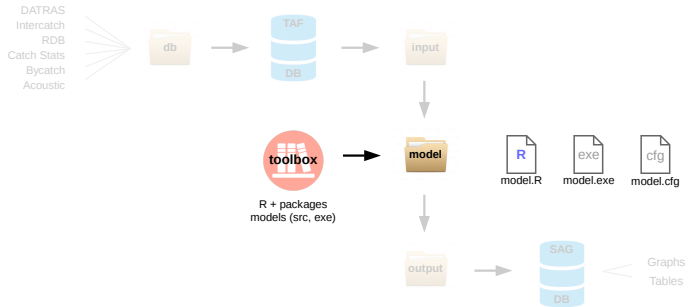
# TAF database



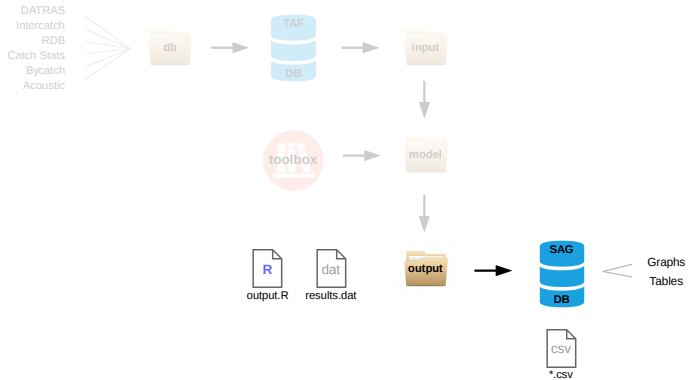
# Input data



# Model run

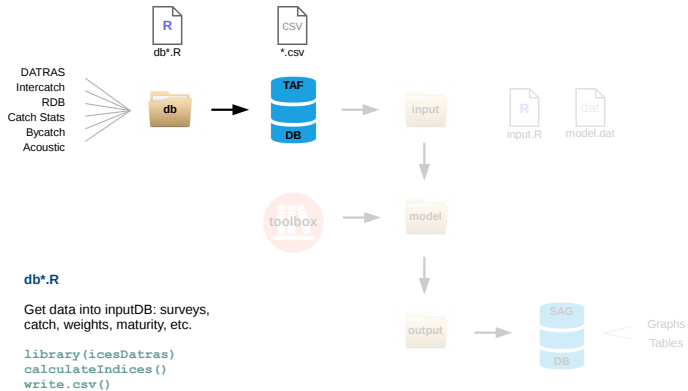


# Output

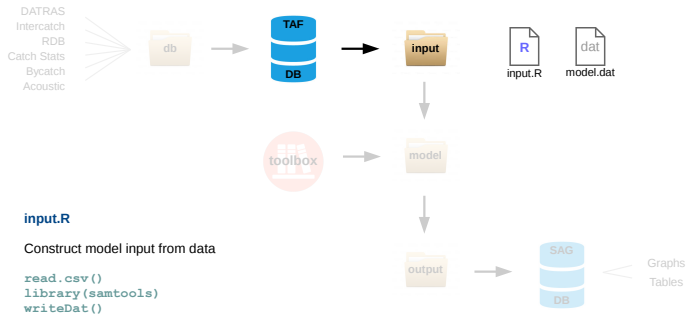


# System workflow (scripts)

# db\*.R

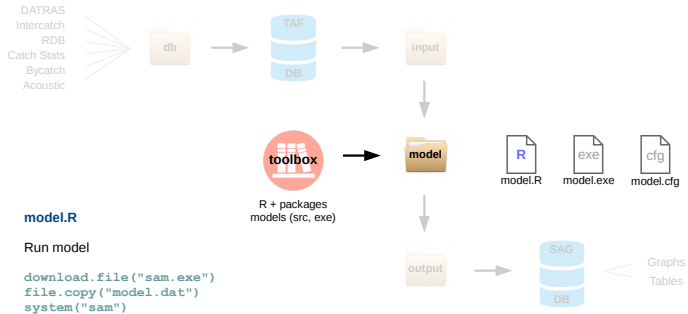


# input.R

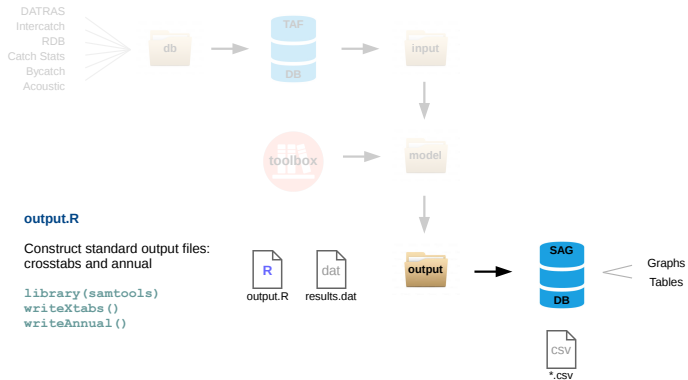




# model.R



# output.R



# Scripts

## db\*.R

Get data into inputDB: surveys, catch, weights, maturity, etc.

```
library(icesDatras); calculateIndices(); write.csv()
```

## input.R

Construct model input from data

```
read.csv(); library(samtools); writeDat()
```

## model.R

Run model

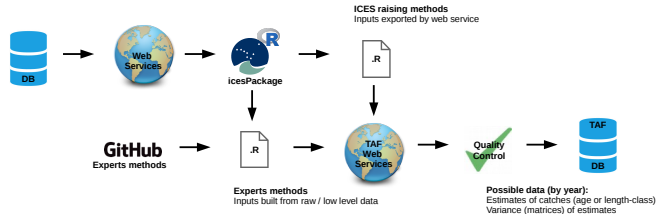
```
download.file("sam.exe"); file.copy("model.dat"); system("sam")
```

## output.R

Construct standard output files: crosstabs and annual

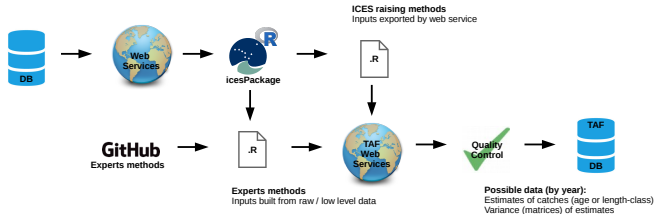
```
library(samtools); writeXtabs(); writeAnnual()
```

# General input model (db\*.R)



1. **Two options**    *a) using ICES methods; b) using experts methods*
2. **Encapsulated, repeatable and quality checked** data flow  
*Web services to read from and check and write to ICES DBs*
3. Gives **EG direct access** to data methods - benchmarks, reviews
4. **Allows for new development and sharing** of methods

## General input model (cont.)



5. **Transparent** open-source procedures and methods

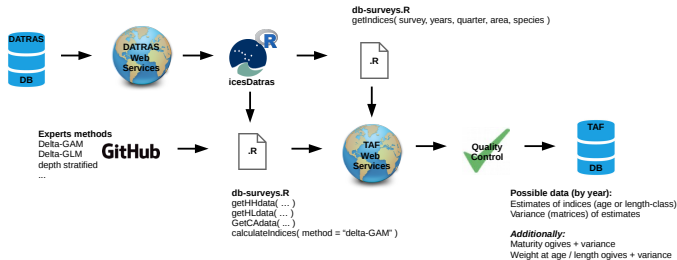
6. **Requirements** (broadly)

*Read access to ICES databases via web services*

*Automatic data submission to TAF via web services*

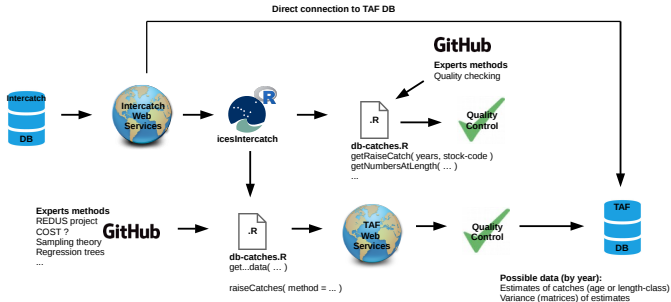
*Data and method versioning*

# DATRAS input model



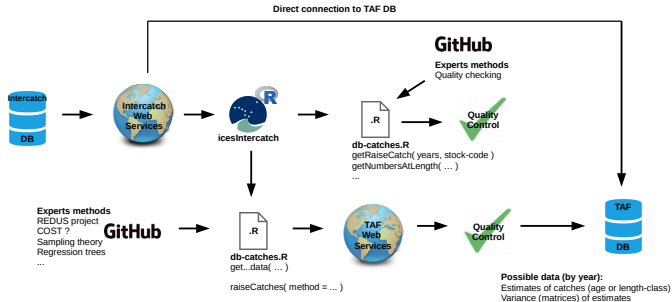
1. Some stocks (almost) using this workflow already:  
*e.g. NS Horse mackerel, NS Cod*
2. Many stocks easily fit into this workflow:  
*e.g. NS stocks, WoS demersal stocks*
3. **Links on GitHub:** Delta-GLMM, Delta-GAM

# Intercatch input model



1. **New methodology** incorporate specialised statistical methods  
*e.g. Redus, XSAM model (NSS herring assessment)*
2. **Requirements** Using expert methods  
*API access to low level data*

# Intercatch input model (cont.)



## 3. Requirements using ICES methods

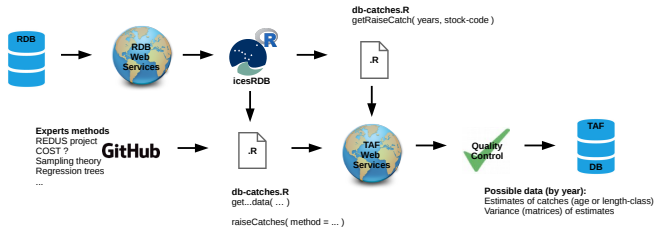
*Direct input from InterCatch to TAF DB*

## 4. Would be nice using ICES methods

*API access to current InterCatch output  
(improves current workflow)*



# RDB input model

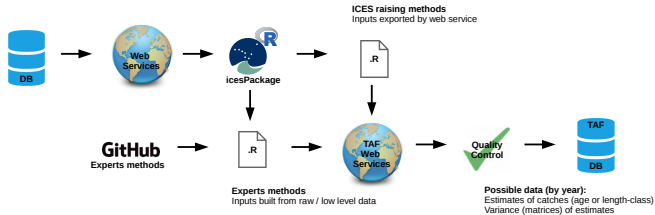


## 1. Notes

*Issues with RDB access via web services?*

*Scripts visible, but access to RDB only within a userspace?*

# Non-ICES DB input model



## 1. Data read from other sources (national DBs)

*Must still go through quality control*

*Where possible, R scripts with raw data files stored*

# System components

# GitHub – transparency of methods

1. **ices-tools-prod** ▶ ices-tools-prod

*tools maintained/supported by ICES, e.g. icesDatras, icesSAG*

2. **ices-tools-dev**

*tools maintained by EGs and/or institutions*

3. **ices-eg**

*workspaces for EGs to keep stock specific methods, QC code, etc.*

4. **ices-taf**

*where TAF R scripts will reside (input.R, model.R, output.R, etc.)*

## Existing examples

- TAF demo: [github.com/ices-taf/demo](https://github.com/ices-taf/demo)
- Computational server: [stockassessment.org](https://stockassessment.org)
- CSV data server: [data.hafro.is](https://data.hafro.is)
- icesDatras and icesSAG:  
[github.com/ices-tools-prod/icesDatras](https://github.com/ices-tools-prod/icesDatras)  
[github.com/ices-tools-prod/icesSAG](https://github.com/ices-tools-prod/icesSAG)
- GitHub EG repositories: [github.com/ices-eg](https://github.com/ices-eg)

# User interface

Open [taf.ices.dk](http://taf.ices.dk) in a web browser

**Browse** (everything is open) or **log in** to modify/run assessments

- **Stock mode**      upload, edit, save, run

- **Boss mode**

HAWG	NWWG
<input checked="" type="radio"/> her-3a22	<input checked="" type="radio"/> cod-iceg
<input type="radio"/> her-47d3	<input checked="" type="radio"/> sai-faro
<input checked="" type="radio"/> her-67bc	<input checked="" type="radio"/> sai-icel
<input checked="" type="radio"/> her-irls	<input type="radio"/> smr-5614
<input type="radio"/> ...	<input type="radio"/> ...

Download any dataset into R using **read.csv**

# Discussion topics I

- Data sources *databases, user upload*
- Quality checks
- Computational server *ices or cloud, windows or linux  
stockassessment.org*
- Toolbox *one toolset per year, virtual machine*
- Web user interface
- GitHub
- + other topics related to TAF

## Discussion topics II

- User workflow    *stock+assmt coordinators, secretariat, others*
- Data input    *“gold-silver-bronze”*
- User scripts
- Our interaction with EGs    *chairs, good stocks to start with*
- Types of output    *catch in tonnes, uncertainty, forecast, retro*
- + other topics related to TAF



# Collaborations required

## Existing groups

- DATRAS Team    *icesDatras devel, web service features, ...*
- Intercatch/RDB Team    *icesInterCatch, interface, ...*
- SAG Team    *icesSAG devel, data import, DB structure, ...*

## New groups

- Web server
- Computational server