Agenda Item E.2
Attachment 4
November 2019

Updating the Fishery Regulation Assessment Model (FRAM) User Manual

D. Auerbach, A. Hagen-Breaux, D. Dapp, Washington Department of Fish and Wildlife

M. Bellman, O. Miler, Northwest Indian Fisheries Commission

October 2019

Summary

Since completion of the most recent “User Manual” in 2007[1], the Fishery Regulation Assessment Model (FRAM) software has incorporated functional and design changes in response to management needs and software advancements. Since the last user manual was published, FRAM has undergone significant changes and transitioned from a Visual Basic 6 platform that works with text and binary files to a Visual Studio.Net platform that works with MS Access databases. This has enhanced the organization and sharing of model run data while facilitating output creation by reducing reliance on the creation of “report drivers” and supporting flexible queries across multiple runs.

These changes prompted the development of a new User Manual with descriptions of the model and its operation. Staff from the Washington Department of Fish and Wildlife (WDFW) and the Northwest Indian Fisheries Commission (NWIFC) undertook this task during 2019, and have completed an initial work product[2].

The revised and updated FRAM documentation is implemented as an online living document; a code repository of scripts that form the basis for a simple website organized as a “book” of several chapters. This approach provides many attractive features including ease of access and navigation. An accessible user manual can increase the mutual technical understanding among users of FRAM and its outputs, and thereby increase confidence in and support for decisions based on those outputs.

Methodology

During the spring of 2019, WDFW and NWIFC staff decided on an online “living document” approach. After reviewing existing materials, staff converged on an initial chapter structure, then added and revised content to reach the current draft.

This project updates the FRAM User Manual, while simultaneously introducing a means to support the consistent maintenance of help resources as the application itself continues to

[1] Currently available from the PFMC website: https://www.pcouncil.org/salmon/background/document-library/fishery-regulation-assessment-model-fram-documentation/

[2] Currently viewable at: https://wdfw-fp.github.io/framvs_doc/index.html

change[3]. The revised User Manual consists of a collection of Rmarkdown files, organized with the *bookdown*[4] package and hosted on the WDFW Fish Program GitHub account (Appendix Figure 1). These scripts are "knit" into various products, including a simple but interactive website based on Gitbook bootstrap styling as well as static output as Microsoft Word docx files. At this time, the underlying repository is only accessible to designated collaborators, but the "published" website is publicly and freely available for viewing.

Descriptions of Rmarkdown and Git/GitHub are beyond the scope of this report, but the basic work cycle involves editing Rmd source files and re-generating html and/or docx output files after a "commit" of one or several changes within a local copy of the repository. These changes are then shared by "pushing" the local commit(s) up to the hosted repository. Changes from multiple staff can be readily merged, and the record of committed changes allows rapid recovery of prior states. The collection of html files that form the website are generated within a single directory, such that an offline copy of the interactive content could be easily created (or re-hosted elsewhere).

Document Organization

After a landing page with links to the organizations responsible for FRAM (Appendix Figure 2), the current content begins with a FRAM overview in the introductory chapter (Appendix Figure 3), followed by brief descriptions of the various file types associated with FRAM runs (Appendix Figure 4), and then chapters with an overview of the main menu and a "start-to-finish" sequence of steps for the common use case of modifying fishery controls in a single pre-season run (Appendix Figure 5). This "how-to" chapter includes text and dynamic images (gifs) addressing the phases of file preparation, parameter manipulation and output acquisition. Having established an overview of the modeling process, the remaining content is structured thematically, mostly following the FRAM main menu options. Later chapters provide additional detail on editing and running iterations, on understanding the available outputs, on "Backwards FRAM" runs that reconstruct pre-fishing cohorts from terminal run sizes or escapements, on the

[3] As of this writing: `FramVS19bMarch11.exe` at https://github.com/Angelikahagen/MainFRAM_VS-Repo
[4] https://bookdown.org/

closely associated Terminal Area Mangement Module (TAMM) files, and on various utilities and advanced uses (Appendix Figure 6). A troubleshooting section with common error messages, a glossary of key terms, and tables of current stocks and fisheries wrap up the main document.

Highlights of Hosted Living User Manual

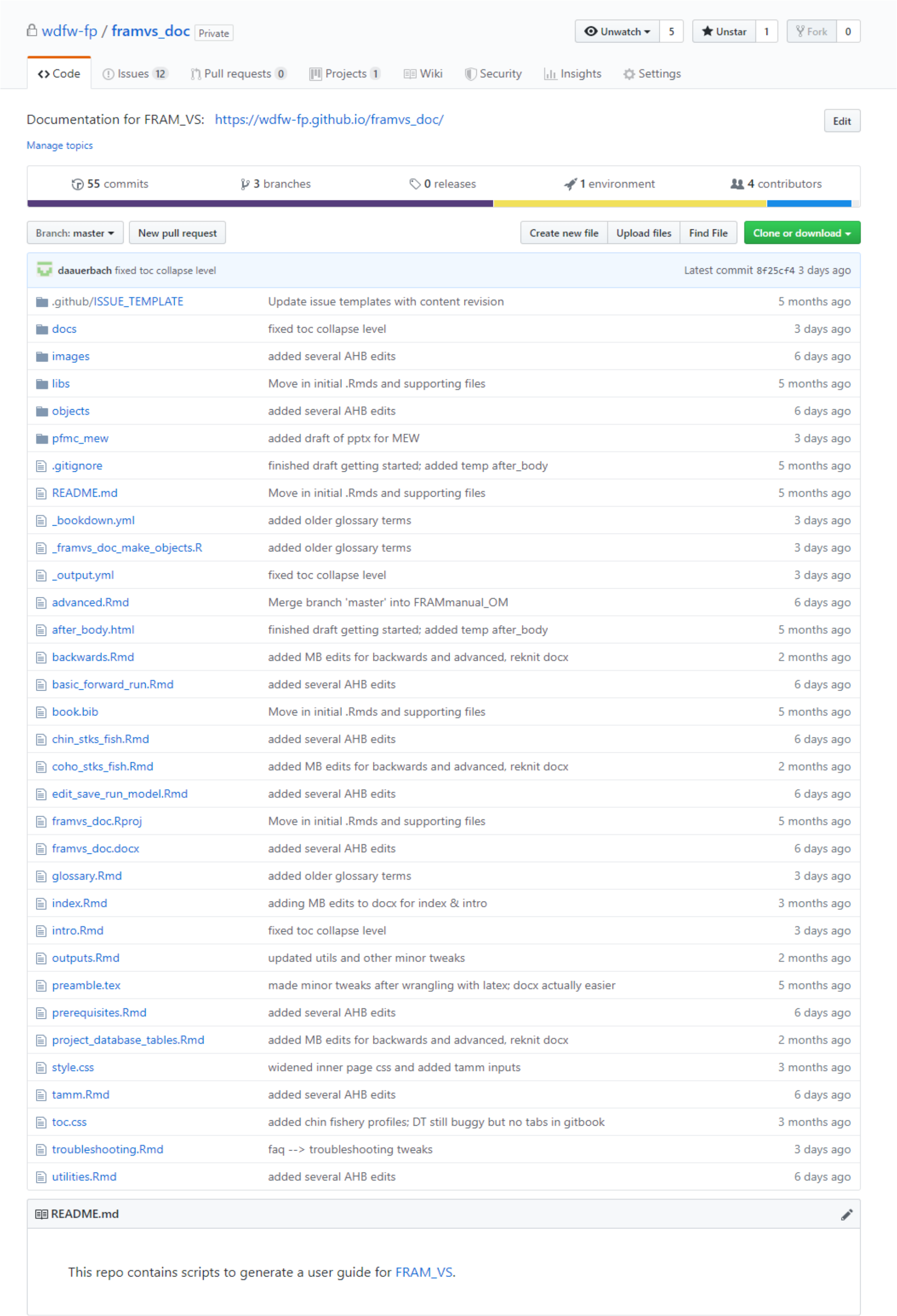
The ease of ongoing development positions the documentation to stay synchronized with the underlying FRAM application, and the webhosted, living user manual provides fast, intuitive navigation through refreshed content.

Users of this electronic "book" can navigate via various mouse-click and keyboard options, including the collapsing chapter sidebar, the "last/next page" arrows, and embedded links to other related content. The static Word files that also can be produced from the source Rmarkdown scripts preserve some of this functionality within that application's Navigation sidebar (Appendix Figure 7).

In addition to screen shots of various menus, the web book includes several looped "mini-tutorials" illustrating a sequence of steps for a particular model task. Built-in search and font-scaling functionality may also make it easier to find information on a particular topic and then read it comfortably. Finally, although modeling and driving is not recommended, the ability to quickly read the User Manual from any device with a web-browser greatly increases the portability of this reference resource

As noted, content creation and revision for this project is fully tracked in the commit log that is generated by the Git version control system (Appendix Figure 8). Project management boards and issue threads facilitate the integration of user feedback and ensure attribution. In combination, these features build collaborative confidence, foster transparency, and encourage testing ideas.

Appendix



Documentation for FRAM_VS: https://wdfw-fp.github.io/framvs_doc/

55 commits | 3 branches | 0 releases | 1 environment | 4 contributors

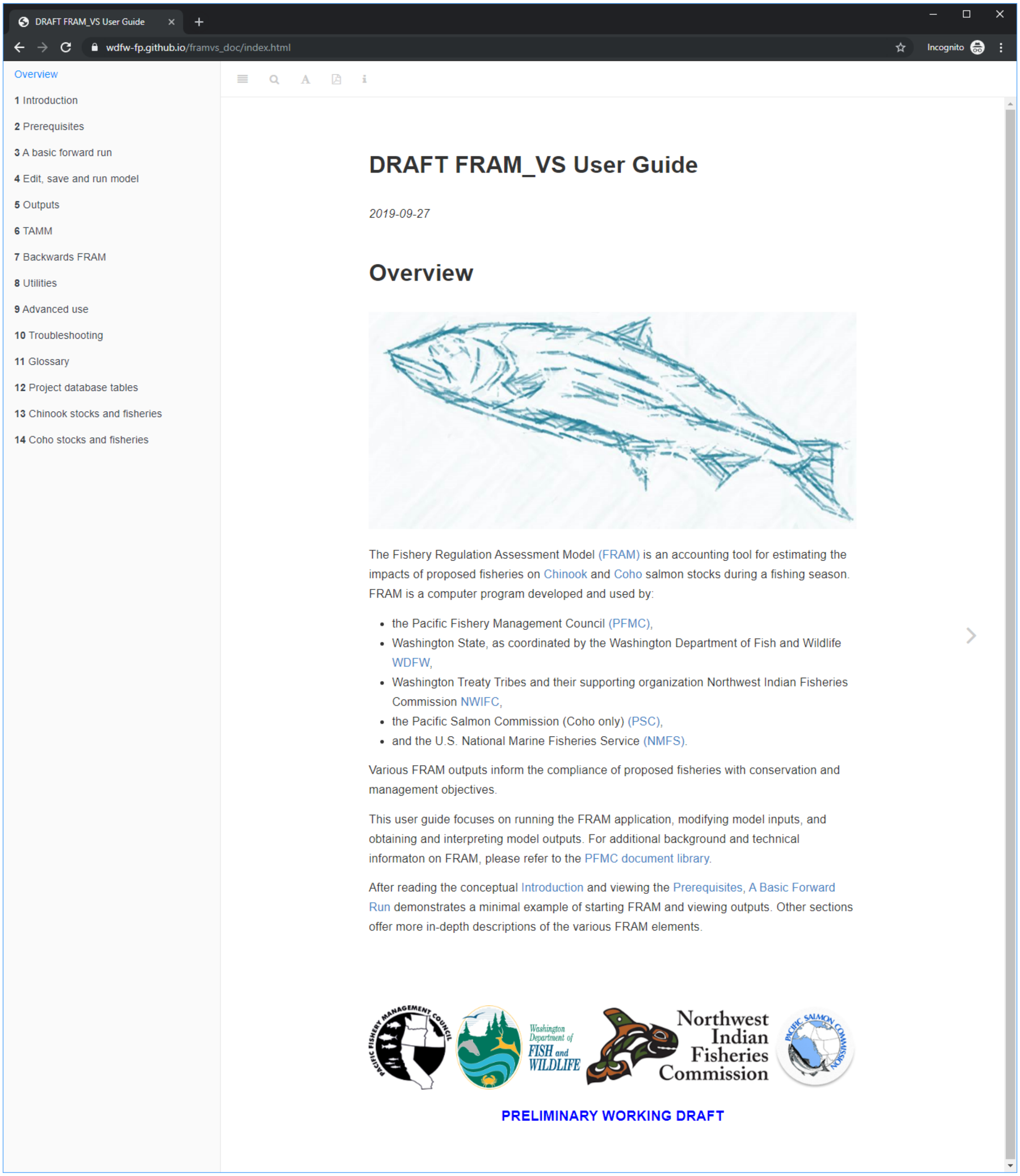
daauerbach fixed toc collapse level — Latest commit 8f25cf4 3 days ago

.github/ISSUE_TEMPLATE	Update issue templates with content revision	5 months ago
docs	fixed toc collapse level	3 days ago
images	added several AHB edits	6 days ago
libs	Move in initial .Rmds and supporting files	5 months ago
objects	added several AHB edits	6 days ago
pfmc_mew	added draft of pptx for MEW	3 days ago
.gitignore	finished draft getting started; added temp after_body	5 months ago
README.md	Move in initial .Rmds and supporting files	5 months ago
_bookdown.yml	added older glossary terms	3 days ago
_framvs_doc_make_objects.R	added older glossary terms	3 days ago
_output.yml	fixed toc collapse level	3 days ago
advanced.Rmd	Merge branch 'master' into FRAMmanual_OM	6 days ago
after_body.html	finished draft getting started; added temp after_body	5 months ago
backwards.Rmd	added MB edits for backwards and advanced, reknit docx	2 months ago
basic_forward_run.Rmd	added several AHB edits	6 days ago
book.bib	Move in initial .Rmds and supporting files	5 months ago
chin_stks_fish.Rmd	added several AHB edits	6 days ago
coho_stks_fish.Rmd	added MB edits for backwards and advanced, reknit docx	2 months ago
edit_save_run_model.Rmd	added several AHB edits	6 days ago
framvs_doc.Rproj	Move in initial .Rmds and supporting files	5 months ago
framvs_doc.docx	added several AHB edits	6 days ago
glossary.Rmd	added older glossary terms	3 days ago
index.Rmd	adding MB edits to docx for index & intro	3 months ago
intro.Rmd	fixed toc collapse level	3 days ago
outputs.Rmd	updated utils and other minor tweaks	2 months ago
preamble.tex	made minor tweaks after wrangling with latex; docx actually easier	5 months ago
prerequisites.Rmd	added several AHB edits	6 days ago
project_database_tables.Rmd	added MB edits for backwards and advanced, reknit docx	2 months ago
style.css	widened inner page css and added tamm inputs	3 months ago
tamm.Rmd	added several AHB edits	6 days ago
toc.css	added chin fishery profiles; DT still buggy but no tabs in gitbook	3 months ago
troubleshooting.Rmd	faq --> troubleshooting tweaks	3 days ago
utilities.Rmd	added several AHB edits	6 days ago

README.md

This repo contains scripts to generate a user guide for FRAM_VS.

Figure 1 - GitHub repo code overview



DRAFT FRAM_VS User Guide

2019-09-27

Overview

The Fishery Regulation Assessment Model (FRAM) is an accounting tool for estimating the impacts of proposed fisheries on Chinook and Coho salmon stocks during a fishing season. FRAM is a computer program developed and used by:

- the Pacific Fishery Management Council (PFMC),
- Washington State, as coordinated by the Washington Department of Fish and Wildlife WDFW,
- Washington Treaty Tribes and their supporting organization Northwest Indian Fisheries Commission NWIFC,
- the Pacific Salmon Commission (Coho only) (PSC),
- and the U.S. National Marine Fisheries Service (NMFS).

Various FRAM outputs inform the compliance of proposed fisheries with conservation and management objectives.

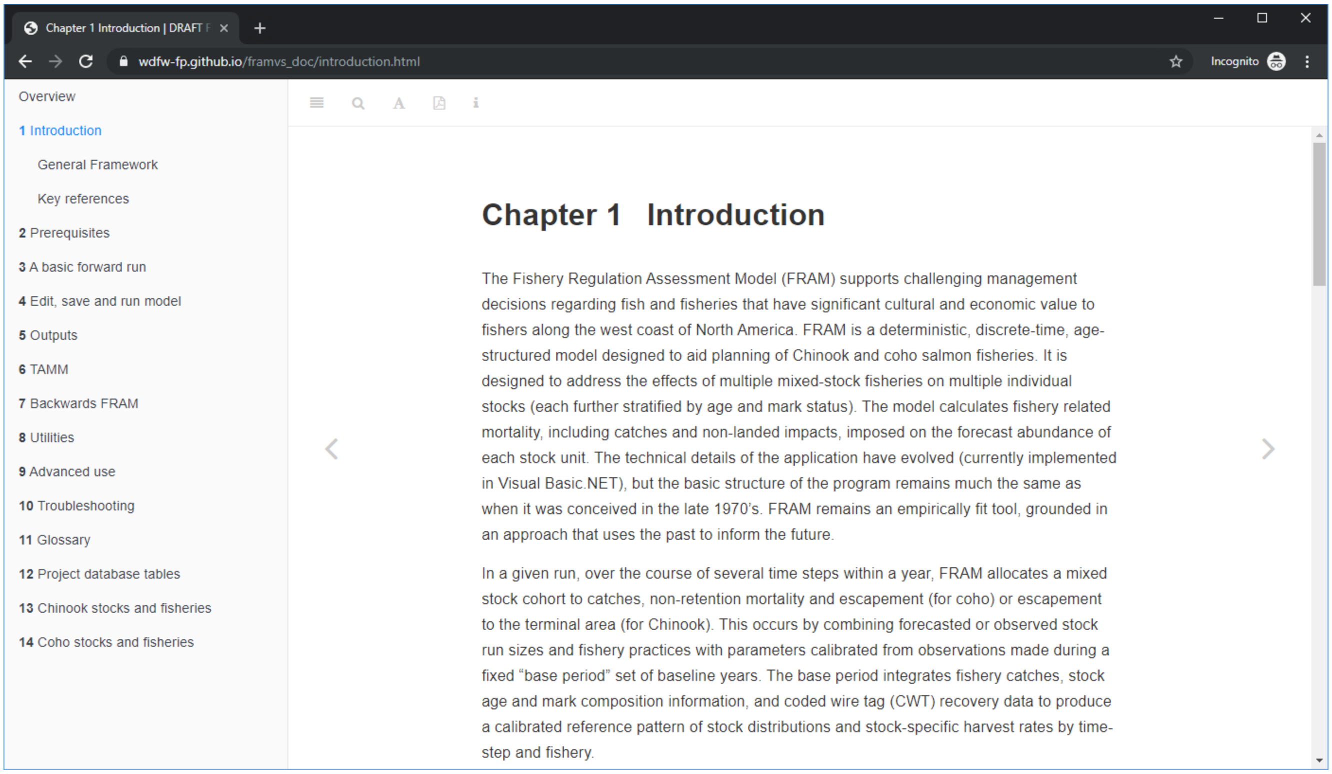
This user guide focuses on running the FRAM application, modifying model inputs, and obtaining and interpreting model outputs. For additional background and technical informaton on FRAM, please refer to the PFMC document library.

After reading the conceptual Introduction and viewing the Prerequisites, A Basic Forward Run demonstrates a minimal example of starting FRAM and viewing outputs. Other sections offer more in-depth descriptions of the various FRAM elements.

Pacific Fishery Management Council
Washington Department of Fish and Wildlife
Northwest Indian Fisheries Commission
Pacific Salmon Commission

PRELIMINARY WORKING DRAFT

Figure 2 - Landing page

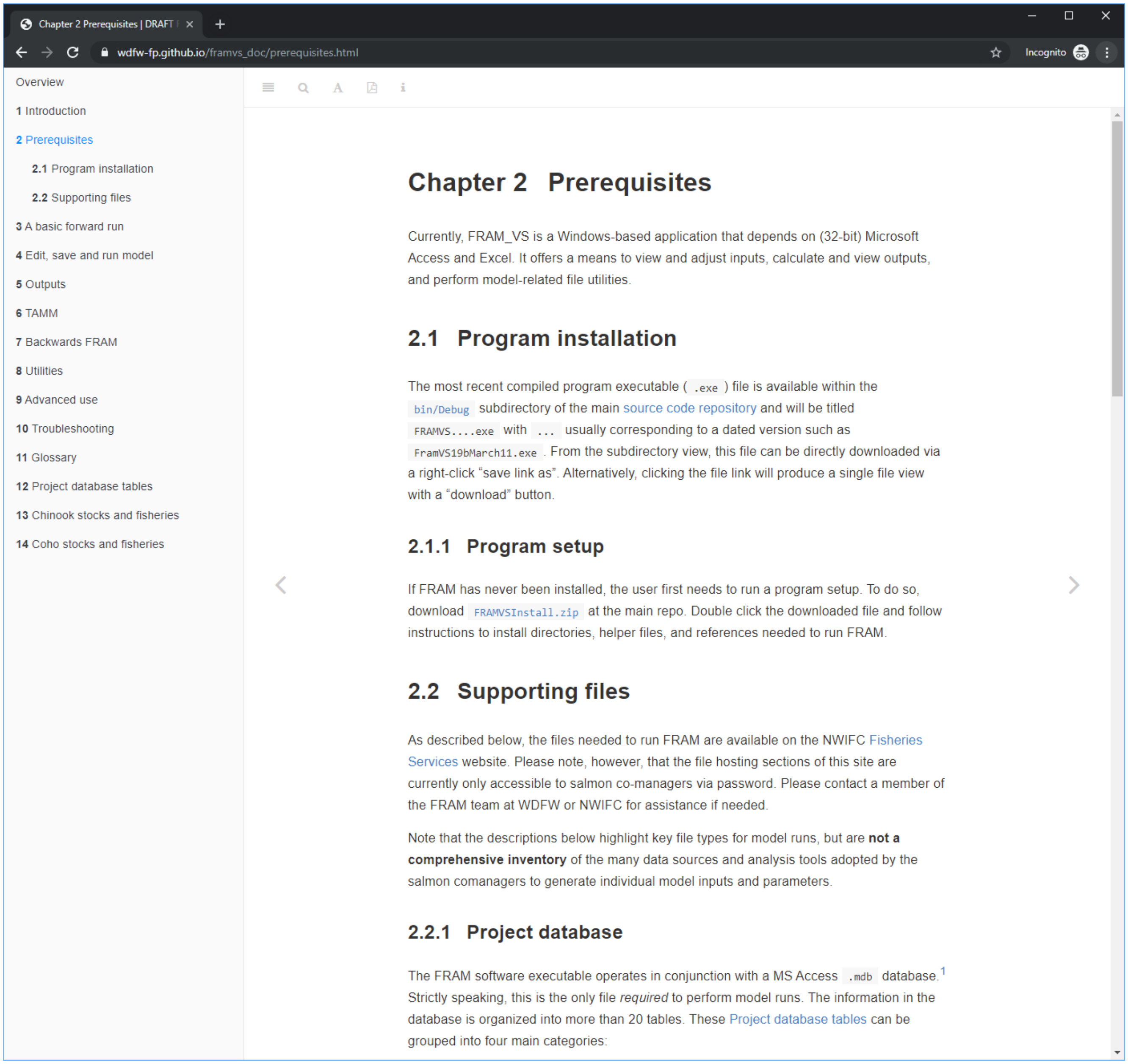

Chapter 1 Introduction | DRAFT F
wdfw-fp.github.io/framvs_doc/introduction.html
Incognito
Overview
1 Introduction
General Framework
Key references
2 Prerequisites
3 A basic forward run
4 Edit, save and run model
5 Outputs
6 TAMM
7 Backwards FRAM
8 Utilities
9 Advanced use
10 Troubleshooting
11 Glossary
12 Project database tables
13 Chinook stocks and fisheries
14 Coho stocks and fisheries

Chapter 1 Introduction

The Fishery Regulation Assessment Model (FRAM) supports challenging management decisions regarding fish and fisheries that have significant cultural and economic value to fishers along the west coast of North America. FRAM is a deterministic, discrete-time, age-structured model designed to aid planning of Chinook and coho salmon fisheries. It is designed to address the effects of multiple mixed-stock fisheries on multiple individual stocks (each further stratified by age and mark status). The model calculates fishery related mortality, including catches and non-landed impacts, imposed on the forecast abundance of each stock unit. The technical details of the application have evolved (currently implemented in Visual Basic.NET), but the basic structure of the program remains much the same as when it was conceived in the late 1970's. FRAM remains an empirically fit tool, grounded in an approach that uses the past to inform the future.

In a given run, over the course of several time steps within a year, FRAM allocates a mixed stock cohort to catches, non-retention mortality and escapement (for coho) or escapement to the terminal area (for Chinook). This occurs by combining forecasted or observed stock run sizes and fishery practices with parameters calibrated from observations made during a fixed "base period" set of baseline years. The base period integrates fishery catches, stock age and mark composition information, and coded wire tag (CWT) recovery data to produce a calibrated reference pattern of stock distributions and stock-specific harvest rates by time-step and fishery.

Figure 3 - Introduction


Chapter 2 Prerequisites | DRAFT
wdfw-fp.github.io/framvs_doc/prerequisites.html
Incognito

Chapter 2 Prerequisites

Currently, FRAM_VS is a Windows-based application that depends on (32-bit) Microsoft Access and Excel. It offers a means to view and adjust inputs, calculate and view outputs, and perform model-related file utilities.

2.1 Program installation

The most recent compiled program executable (`.exe`) file is available within the `bin/Debug` subdirectory of the main source code repository and will be titled `FRAMVS....exe` with `...` usually corresponding to a dated version such as `FramVS19bMarch11.exe` . From the subdirectory view, this file can be directly downloaded via a right-click "save link as". Alternatively, clicking the file link will produce a single file view with a "download" button.

2.1.1 Program setup

If FRAM has never been installed, the user first needs to run a program setup. To do so, download `FRAMVSInstall.zip` at the main repo. Double click the downloaded file and follow instructions to install directories, helper files, and references needed to run FRAM.

2.2 Supporting files

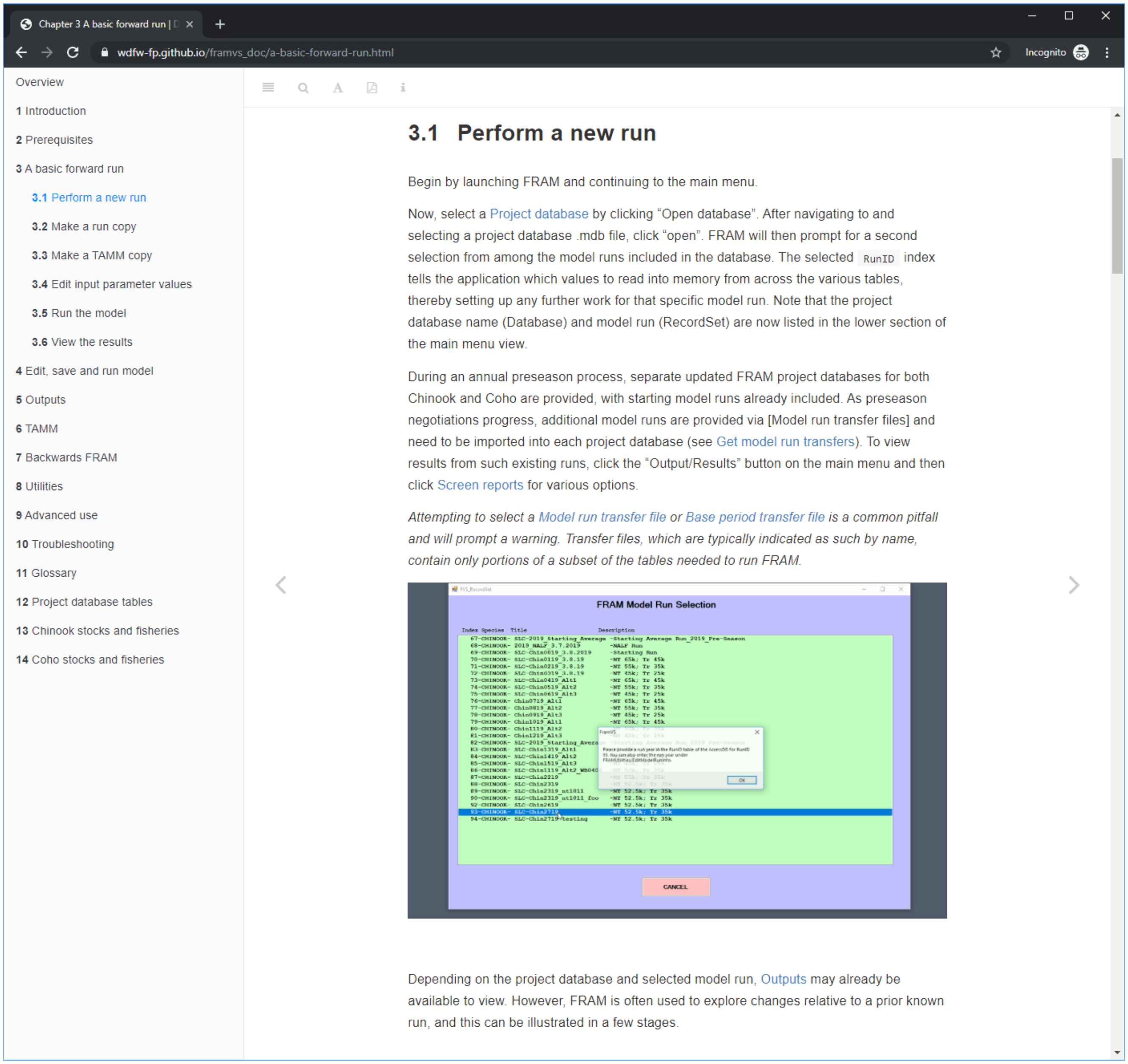
As described below, the files needed to run FRAM are available on the NWIFC Fisheries Services website. Please note, however, that the file hosting sections of this site are currently only accessible to salmon co-managers via password. Please contact a member of the FRAM team at WDFW or NWIFC for assistance if needed.

Note that the descriptions below highlight key file types for model runs, but are **not a comprehensive inventory** of the many data sources and analysis tools adopted by the salmon comanagers to generate individual model inputs and parameters.

2.2.1 Project database

The FRAM software executable operates in conjunction with a MS Access `.mdb` database.[1] Strictly speaking, this is the only file *required* to perform model runs. The information in the database is organized into more than 20 tables. These Project database tables can be grouped into four main categories:

Figure 4 - Prerequisite setup and supporting files


Chapter 3 A basic forward run | D
wdfw-fp.github.io/framvs_doc/a-basic-forward-run.html
Incognito

3.1 Perform a new run

Begin by launching FRAM and continuing to the main menu.

Now, select a Project database by clicking "Open database". After navigating to and selecting a project database .mdb file, click "open". FRAM will then prompt for a second selection from among the model runs included in the database. The selected `RunID` index tells the application which values to read into memory from across the various tables, thereby setting up any further work for that specific model run. Note that the project database name (Database) and model run (RecordSet) are now listed in the lower section of the main menu view.

During an annual preseason process, separate updated FRAM project databases for both Chinook and Coho are provided, with starting model runs already included. As preseason negotiations progress, additional model runs are provided via [Model run transfer files] and need to be imported into each project database (see Get model run transfers). To view results from such existing runs, click the "Output/Results" button on the main menu and then click Screen reports for various options.

Attempting to select a Model run transfer file or Base period transfer file is a common pitfall and will prompt a warning. Transfer files, which are typically indicated as such by name, contain only portions of a subset of the tables needed to run FRAM.

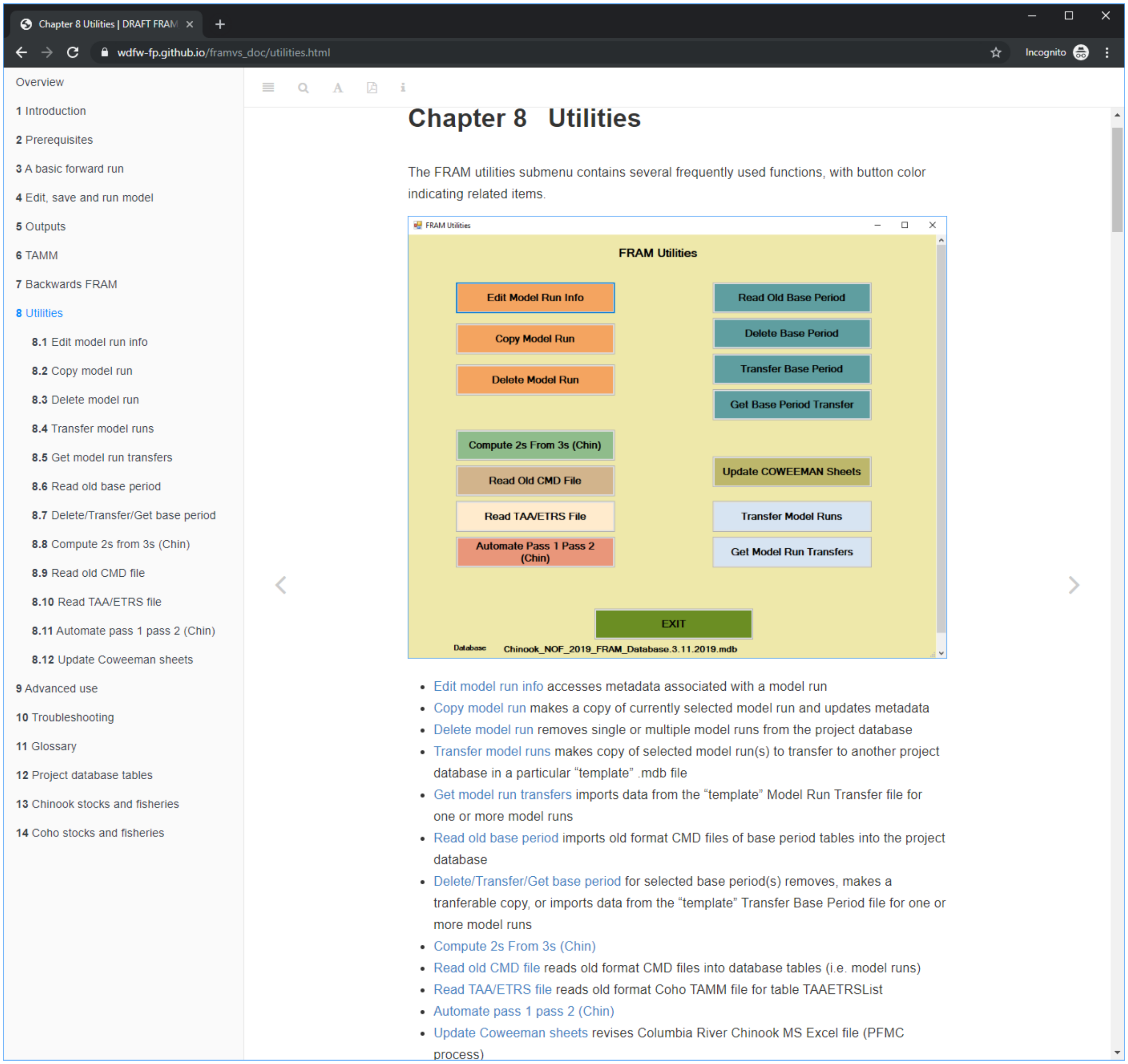
FRAM Model Run Selection

```
Index Species  Title                       Description
  67-CHINOOK- SLC-2019_Starting_Average  -Starting Average Run_2019_Pre-Season
  68-CHINOOK- 2019_NALF_3.7.2019         -NALF Run
  69-CHINOOK- SLC-Chin0019_3.8.2019      -Starting Run
  70-CHINOOK- SLC-Chin0119_3.8.19        -NT 65k; Tr 45k
  71-CHINOOK- SLC-Chin0219_3.8.19        -NT 55k; Tr 35k
  72-CHINOOK- SLC-Chin0319_3.8.19        -NT 45k; Tr 25k
  73-CHINOOK- SLC-Chin0419_Alt1          -NT 65k; Tr 45k
  74-CHINOOK- SLC-Chin0519_Alt2          -NT 55k; Tr 35k
  75-CHINOOK- SLC-Chin0619_Alt3          -NT 45k; Tr 25k
  76-CHINOOK- Chin0719_Alt1              -NT 65k; Tr 45k
  77-CHINOOK- Chin0819_Alt2              -NT 55k; Tr 35k
  78-CHINOOK- Chin0919_Alt3              -NT 45k; Tr 25k
  79-CHINOOK- Chin1019_Alt1              -NT 65k; Tr 45k
  80-CHINOOK- Chin1119_Alt2              -NT 55k; Tr 35k
  81-CHINOOK- Chin1219_Alt3              -NT 45k; Tr 25k
  82-CHINOOK- SLC-2019_Starting_Average  -Starting Average Run_2019_Pre-Season
  83-CHINOOK- SLC-Chin1319_Alt1
  84-CHINOOK- SLC-Chin1419_Alt2
  85-CHINOOK- SLC-Chin1519_Alt3
  86-CHINOOK- SLC-Chin1119_Alt2_WN0404   -NT 50k; Tr 35k
  87-CHINOOK- SLC-Chin2219               -NT 55k; Tr 35k
  88-CHINOOK- SLC-Chin2319               -NT 52.5k; Tr 35k
  89-CHINOOK- SLC-Chin2319_nt1011        -NT 52.5k; Tr 35k
  90-CHINOOK- SLC-Chin2319_nt1011_foo    -NT 52.5k; Tr 35k
  92-CHINOOK- SLC-Chin2619               -NT 52.5k; Tr 35k
  93-CHINOOK- SLC-Chin2719               -NT 52.5k; Tr 35k
  94-CHINOOK- SLC-Chin2719 testing       -NT 52.5k; Tr 35k
```

Please provide a run year in the RunID table of the AccessDB for RunID 93. You can also enter the run year under FRAMUtilities/EditModelRunInfo.

Depending on the project database and selected model run, Outputs may already be available to view. However, FRAM is often used to explore changes relative to a prior known run, and this can be illustrated in a few stages.

Figure 5 - A step-by-step forward run tutorial



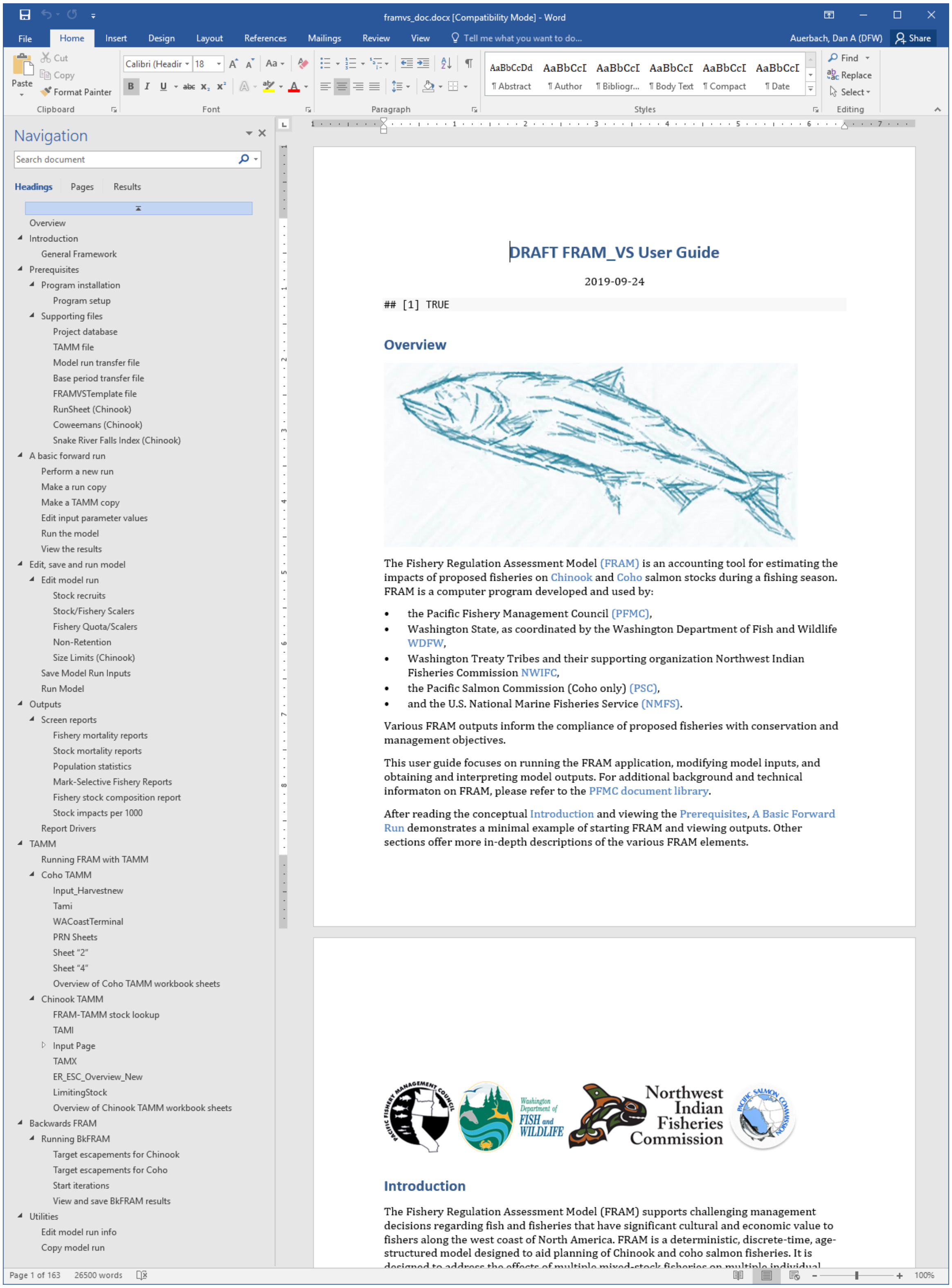
Chapter 8 Utilities

The FRAM utilities submenu contains several frequently used functions, with button color indicating related items.

Database Chinook_NOF_2019_FRAM_Database.3.11.2019.mdb

- Edit model run info accesses metadata associated with a model run
- Copy model run makes a copy of currently selected model run and updates metadata
- Delete model run removes single or multiple model runs from the project database
- Transfer model runs makes copy of selected model run(s) to transfer to another project database in a particular “template” .mdb file
- Get model run transfers imports data from the “template” Model Run Transfer file for one or more model runs
- Read old base period imports old format CMD files of base period tables into the project database
- Delete/Transfer/Get base period for selected base period(s) removes, makes a tranferable copy, or imports data from the “template” Transfer Base Period file for one or more model runs
- Compute 2s From 3s (Chin)
- Read old CMD file reads old format CMD files into database tables (i.e. model runs)
- Read TAA/ETRS file reads old format Coho TAMM file for table TAAETRSList
- Automate pass 1 pass 2 (Chin)
- Update Coweeman sheets revises Columbia River Chinook MS Excel file (PFMC process)

Figure 6 - Complete description of included file utilities



Overview
Introduction
General Framework
Prerequisites
Program installation
Program setup
Supporting files
Project database
TAMM file
Model run transfer file
Base period transfer file
FRAMVSTemplate file
RunSheet (Chinook)
Coweemans (Chinook)
Snake River Falls Index (Chinook)
A basic forward run
Perform a new run
Make a run copy
Make a TAMM copy
Edit input parameter values
Run the model
View the results
Edit, save and run model
Edit model run
Stock recruits
Stock/Fishery Scalers
Fishery Quota/Scalers
Non-Retention
Size Limits (Chinook)
Save Model Run Inputs
Run Model
Outputs
Screen reports
Fishery mortality reports
Stock mortality reports
Population statistics
Mark-Selective Fishery Reports
Fishery stock composition report
Stock impacts per 1000
Report Drivers
TAMM
Running FRAM with TAMM
Coho TAMM
Input_Harvestnew
Tami
WACoastTerminal
PRN Sheets
Sheet "2"
Sheet "4"
Overview of Coho TAMM workbook sheets
Chinook TAMM
FRAM-TAMM stock lookup
TAMI
Input Page
TAMX
ER_ESC_Overview_New
LimitingStock
Overview of Chinook TAMM workbook sheets
Backwards FRAM
Running BkFRAM
Target escapements for Chinook
Target escapements for Coho
Start iterations
View and save BkFRAM results
Utilities
Edit model run info
Copy model run

DRAFT FRAM_VS User Guide

2019-09-24

```
## [1] TRUE
```

Overview

The Fishery Regulation Assessment Model (FRAM) is an accounting tool for estimating the impacts of proposed fisheries on Chinook and Coho salmon stocks during a fishing season. FRAM is a computer program developed and used by:

- the Pacific Fishery Management Council (PFMC),
- Washington State, as coordinated by the Washington Department of Fish and Wildlife WDFW,
- Washington Treaty Tribes and their supporting organization Northwest Indian Fisheries Commission NWIFC,
- the Pacific Salmon Commission (Coho only) (PSC),
- and the U.S. National Marine Fisheries Service (NMFS).

Various FRAM outputs inform the compliance of proposed fisheries with conservation and management objectives.

This user guide focuses on running the FRAM application, modifying model inputs, and obtaining and interpreting model outputs. For additional background and technical informaton on FRAM, please refer to the PFMC document library.

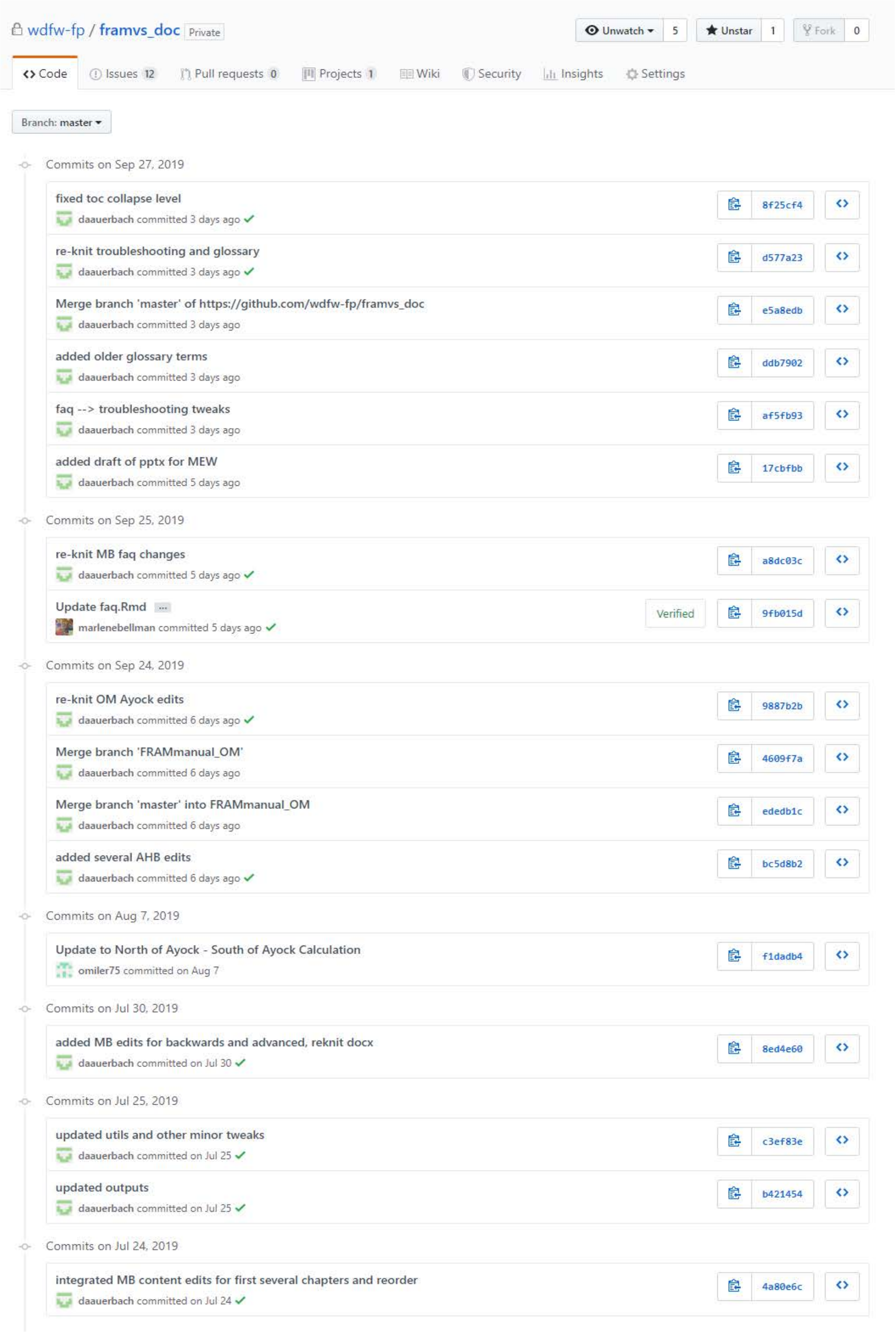
After reading the conceptual Introduction and viewing the Prerequisites, A Basic Forward Run demonstrates a minimal example of starting FRAM and viewing outputs. Other sections offer more in-depth descriptions of the various FRAM elements.

Pacific Fishery Management Council
Washington Department of Fish and Wildlife
Northwest Indian Fisheries Commission
Pacific Salmon Commission

Introduction

The Fishery Regulation Assessment Model (FRAM) supports challenging management decisions regarding fish and fisheries that have significant cultural and economic value to fishers along the west coast of North America. FRAM is a deterministic, discrete-time, age-structured model designed to aid planning of Chinook and coho salmon fisheries. It is designed to address the effects of multiple mixed-stock fisheries on multiple individual

Figure 7 - In addition to html output, the source scripts can produce docx for a static snapshot



Commits on Sep 27, 2019

fixed toc collapse level
daauerbach committed 3 days ago ✓ — 8f25cf4

re-knit troubleshooting and glossary
daauerbach committed 3 days ago ✓ — d577a23

Merge branch 'master' of https://github.com/wdfw-fp/framvs_doc
daauerbach committed 3 days ago — e5a8edb

added older glossary terms
daauerbach committed 3 days ago — ddb7902

faq --> troubleshooting tweaks
daauerbach committed 3 days ago — af5fb93

added draft of pptx for MEW
daauerbach committed 5 days ago — 17cbfbb

Commits on Sep 25, 2019

re-knit MB faq changes
daauerbach committed 5 days ago ✓ — a8dc03c

Update faq.Rmd ...
marlenebellman committed 5 days ago ✓ — Verified — 9fb015d

Commits on Sep 24, 2019

re-knit OM Ayock edits
daauerbach committed 6 days ago ✓ — 9887b2b

Merge branch 'FRAMmanual_OM'
daauerbach committed 6 days ago — 4609f7a

Merge branch 'master' into FRAMmanual_OM
daauerbach committed 6 days ago — ededb1c

added several AHB edits
daauerbach committed 6 days ago ✓ — bc5d8b2

Commits on Aug 7, 2019

Update to North of Ayock - South of Ayock Calculation
omiler75 committed on Aug 7 — f1dadb4

Commits on Jul 30, 2019

added MB edits for backwards and advanced, reknit docx
daauerbach committed on Jul 30 ✓ — 8ed4e60

Commits on Jul 25, 2019

updated utils and other minor tweaks
daauerbach committed on Jul 25 ✓ — c3ef83e

updated outputs
daauerbach committed on Jul 25 ✓ — b421454

Commits on Jul 24, 2019

integrated MB content edits for first several chapters and reorder
daauerbach committed on Jul 24 ✓ — 4a80e6c

Figure 8 - The Git log of repository commits creates a full record of changes