

**Skeena Sockeye In-river Run Reconstruction Analysis Model
and
Analysis Results for 1982-2017**

Prepared for:

Pacific Salmon Foundation

Prepared by:

Karl K. English, Cameron Noble and Charmaine Carr-Harris

**LGL Limited
environmental research associates
9768 Second Street
Sidney, BC, V8L 3Y8**

and

**Fisheries and Oceans, Canada
Prince Rupert, B.C.**



TABLE OF CONTENTS

LIST OF TABLES i

LIST OF FIGURES i

LIST OF APPENDICES i

INTRODUCTION 2

 Analysis Objectives 2

DATA SOURCES AND PREPARATION 2

 Sockeye Stocks and Stock Aggregates 2

 Fishery Definitions..... 3

 Escapement and Entering Run Size 3

 Run Timing 4

 Fishery Residence Time..... 4

 Catch 4

 Fishing Patterns..... 6

RUN RECONSTRUCTION MODEL 6

 Model Assumptions 7

 Model Structure 7

RESULTS 9

DISCUSSION 10

LITERATURE CITED 11

LIST OF TABLES

Table 1. Fisheries, stocks and estimated residence time in days for the Skeena sockeye in-river run reconstruction analyses.	13
Table 2. Model input spawning escapement estimates for each sub-stock of Skeena sockeye salmon 1982-2017, where the estimates for Babine sub-stocks are prior to removals in fisheries at or above the Babine fence.....	14
Table 3. Run-timing parameters used for each Skeena sockeye sub-stock group.	15
Table 4. Initial annual estimates of the harvest of Skeena River sockeye by First Nations in FSC fisheries, 1982-2017.....	16
Table 5. Revised annual estimates of the harvest of Skeena River sockeye by First Nations in FSC fisheries, 1982-2017.....	17
Table 6. Annual estimates of the harvest of Skeena River sockeye by First Nations in ESSR and in-land “Demonstration” fisheries, 1993-2014.	18
Table 7. Harvest rate estimates for each Skeena sockeye sub-stock group in First Nation fisheries conducted in and adjacent to the Skeena watershed, 1982-2014.	19

LIST OF FIGURES

Figure 1. Estimated Tye daily abundance, stock composition and run-timing curves for Skeena sockeye stocks in 2006 (late run-timing year).	20
Figure 2. Estimated Tye daily abundance, stock composition and run-timing curves for Skeena sockeye stocks in 2007 (average run-timing year).....	21
Figure 3. Estimated Tye daily abundance, stock composition and run-timing curves for Skeena sockeye stocks in 2008 (early run-timing year).	22

LIST OF APPENDICES

Appendix A - Summary of catch and escapement results for 1982-2017
Appendix B - Skeena Sockeye In-River Model Visual Basic Code

INTRODUCTION

The Fraser, Skeena and Nass watersheds are the three largest sockeye producing watersheds in British Columbia. Exploitation rate estimates for the Nass and Skeena Sockeye stock aggregates are estimated annually using the Northern Boundary Sockeye Run Reconstruction (NBSRR) Model (English et al. 2004; 2005; Alexander et al. 2010). English et al. (2012) provided estimates of marine exploitation rates for each Nass and Skeena sockeye Conservation Unit (CU) using estimates of the migration timing for each CU. These analyses have not included the details on the location and timing of in-river fisheries needed to estimate harvests for the various sockeye CUs or sub-stocks within each watershed. In some years, in-river harvest account for a large portion of the Canadian harvest of sockeye returning to these rivers. Run reconstruction analyses have been used to estimate CU-specific harvest rates for in-river fisheries targeting Fraser Chinook and sockeye salmon (English et al. 2007; Noble 2011). This report provides a brief outline of a Skeena Sockeye In-River (SSIR) run reconstruction model built to combine information on run timing and escapements for Skeena sockeye sub-stocks with catch estimates for each sockeye fishery within the Skeena watershed. The model is similar to those developed for the Fraser River sockeye fisheries within the Fraser watershed except that the Skeena model moves fish forward (upstream) through the fisheries and subtracts sockeye catches from estimates of the number of sockeye entering the Skeena River each day. The Fraser Chinook and sockeye models reconstruct the runs entering the Fraser River by adding catches to daily estimates of escapement for each sub-stock (English et al. 2007; Noble 2011).

Analysis Objectives

Estimates of in-river harvest by sub-stock need to be combined with those for marine fisheries to estimate total exploitation rates for Skeena sockeye. The SSIR model provides a systematic process for combining information on catch, fishery timing, stock-specific migration rates through fisheries, escapement and river entry run timing by sub-stock. The results from these run reconstruction analyses will be combined with the marine harvest rates from NBSRR model to provide estimates of the Canadian and total exploitation rate for each Skeena sockeye sub-stock which are used to derive catch estimates and total annual returns for each of the Skeena sockeye sub-stocks.

DATA SOURCES AND PREPARATION

Sockeye Stocks and Stock Aggregates

The model has the capacity to accommodate details for as many sockeye sub-stocks or run-timing groups as can be defined using the available data and combine these stocks into any number of management groups. Initial analyses of available run-timing data resulted in the definition of 20 sub-stocks for the Skeena watershed (English et al. 2012; 2013). Each of these sub-stock groups contains one or more of the 30 (28 Lake + 2 River type) Skeena River sockeye salmon Conservation Units (CUs) defined in Holtby and Ciruna (2007). Where the run-timing and geographic location of the spawning area is not different for two or more CUs, these CUs were combined into a single sub-stock group. The “+” at the end of a sub-stock name is used to indicate that more than one CU is associated with that sub-stock group (**Table 1**). The only exception in the Zymoetz sub-stock which does not have a “+” but includes three CUs (Mcdonell, Aldrich, Dennis). It should also be noted that the Babine Lake CU has been sub-divided into 5 sub-stock groups based on run-timing and level of enhancement. The two enhanced streams with major spawning channel (Pinkut and Fulton) have slightly different run-timing than the three wild Babine run-timing groups (**Table 1**).

Fishery Definitions

The SSIR model includes two types of fisheries and 12 fishing areas: 3 Tsimshian fishery strata in the lower Skeena and 9 fishing areas used by other Skeena First Nations above Fiddler Creek (**Table 1**). The two fishery types are: 1) food, social and ceremonial (FSC) fisheries and 2) in-river commercial fisheries. For the purpose of this model, in-river commercial fisheries include all sockeye harvested for sale in Escapement Surplus to Spawning Requirement (ESSR) fisheries and the Inland Demonstration fisheries within the Skeena watershed. ESSR fisheries have been conducted in many but not all years from 1993 to present while the first Inland Demonstration fisheries were permitted in 2009. All of the fisheries on the Skeena River (mainstem) harvest multiple stocks and stock composition estimates are not available for these fisheries. Consequently, run reconstruction analysis is required to distribute the estimated weekly catches between the stocks vulnerable to each fishery. Some FSC and most ESSR fisheries occur in locations where only a single stock is affected. The harvest rate estimates for fisheries within the Bulkley and Sustut rivers were computed by dividing the total annual catch by the sum of the estimated abundance entering these fisheries. The daily catch estimates derived from test fishery timing were not used for these fisheries.

Escapement and Entering Run Size

Annual escapement estimates for each Skeena sockeye CU were combined with daily Tyee test fishery data and stock-specific run timing parameters to produce estimates of daily escapement past the Tyee test fishery for each CU.

Annual estimates of spawning escapement for the 20 Skeena CU-run timing groups were derived from four sources: 1) nuSEDS data for 11 non-Babine sockeye stock groups; 2) DFO Prince Rupert historical databases for the 5 Babine sockeye stock groups (Cox-Rogers, DFO, pers. comm.); BC government records for sockeye counted at the Sustut fence (Kris Maier, kris.maier@gov.bc.ca); and 3) assumed fixed values for the remaining 3 sockeye stock groups without escapement monitoring programs (Kluatan+, Sicintine and Bulkley+). Escapement data from nuSEDS is provided first via DFO to the custodian of the NCC database who then provides 'adjusted' escapement data that accounts for the fact that most escapement estimates derived from visual surveys tend to underestimate true escapement, thus they are expanded by a factor of 2.0 prior to being incorporated into the SSIR model. The only exceptions were the Nanika sockeye escapement estimates from 1989 to present which were not expanded because many of these estimates were derived from mark-recapture studies or adjusted using information from these mark-recapture studies. The other escapement estimates that were not expanded were those derived from fence counts (all Babine, Kitwanga 2000-present, Slamgeesh 2000-present and Sustut 2003-present). Most of the non-Babine sockeye stocks had one or more years of missing escapement estimates and these were filled in by using: the average of the estimates for adjacent years, interpolating between the available estimates or a fixed annual value for years prior to the first available estimate. The filled in values are highlighted in yellow in **Table 2**. Escapement estimates for the enhanced Pinkut and Fulton sub-stocks included escapements surplus to the spawning channel capacities. The annual escapement estimates for sub-stocks with terminal fisheries (i.e. Pinkut, Fulton, Sustut and Bulkley-Morice) were increased to account for catches in these terminal fisheries prior to determining the portions that each sub-stock proportions represents of the total return of Skeena sockeye in a given year. These sub-stock proportions were combined with annual estimates of the total sockeye abundance passing Tyee and run-timing parameters derived from analysis of 2000-10 Tyee DNA samples (Cox-Rogers, DFO Rupert, pers. comm.) to compute the daily abundance passing Tyee for each sub-stock.

Babine sockeye represent 85-95% of the sockeye escapement at Tyee and thus the in-river harvest rates for mainstem fisheries that intercept Babine sockeye are not sensitive to the escapement values used for the smaller sockeye stocks. However, the harvest rate estimates for the Morice+ and Sustut+ sub-stocks

are sensitive to the escapement values used for these stocks because a large portion of the in-river catch of these stocks is taken in fisheries within the Bulkley and Sustut rivers that don't include Babine sockeye.

Entering run size and timing at Tyee is provided upon request from Richard Alexander (ralexander@lgl.com), the custodian for the Northern Boundary Sockeye Run Reconstruction Model (NBSRRM). Ensuring that the Tyee data is identical for both models provides for a seamless transition between the two models, as escapement at Tyee is the point of initiation for both models, where the SSIR model works forward (upstream) from Tyee while the NBSRRM works backwards (out) from here. This also allows CU (or stock specific) harvests generated in the SSIR model to be included in the CU harvest estimates generated in the NBSRRM so that estimates of total exploitation rates can be generated.

Run Timing

Initial estimates of river entry timing for 20 sub-stocks of Skeena sockeye were obtained from information reported in a memorandum entitled "SKEENA SOCKEYE SUB-STOCK RUN-TIMING AND ABUNDANCE EVALUATED USING TYEE TEST FISHERY DNA: 2000-2010" prepared by Steve Cox-Rogers dated 23 February 2012 (Cox-Rogers, 2012). The relative timing for each sub-stock was used to determine the offset difference between the average annual timing for all Skeena sockeye and that for a specific sub-stock. For example: Lakelse sockeye were estimated to have a timing 3 weeks earlier than the median timing for Skeena sockeye, therefore, the offset parameter was set at -21 days for Lakelse sockeye. The duration of the run for each sub-stock was also derived from the 2000-2010 DNA stock composition estimates. The parameter used to define the duration was the standard deviation (SD) measured in number of days. The run durations defined in the February 2012 Cox-Rogers memo reflected the combined durations for multiple years. After further discussion and some sensitivity analysis it was agreed that the run duration for individual years was probably less than these initial values and the durations should be reduced by 20%. **Table 3** provides the current offset and SD parameters for each sub-stock. The run timing curve for each sub-stock was defined by a normal curve where the mid-point was defined by combining the stock-specific offset with the median date of sockeye migration past the Tyee test fishery and the start and end points for each timing curve were 3 SD units each side of the mid-point. Therefore, the SD of 13.3 d for Lakelse sockeye results in a total duration of 80 days for this stock. For our initial analysis we used the same offset and SD parameters for each year except 2006, when the duration of the run (SD) for the two large enhanced stocks (Pinkut and Fulton) was increased from 11.2 to 14 days (total duration increased from 67 to 84 days) in order to reflect the notably longer duration of the Skeena sockeye run observed in 2006.

Fishery Residence Time

Residency time was defined as the number of days (to the nearest day) a stock resides within the boundaries of a single fishery. These residence times were derived from historical tagging studies, the differences between peak abundances estimated at Tyee and the Babine fence, and information on the size (river kms) and location of each fishery (English et al. 1985, Steve Cox-Rogers, pers. comm.). The fishery residence times in the current version of the model are the same as those in previous versions, except for the Pinkut and Fulton Terminal fisheries. The residence times for these fisheries was set at 7 days to be consistent with the changes in the input data and methods used to calculate the weekly harvest rates for these fisheries.

Catch

Estimates of annual harvest by Skeena River First Nations for each fishery location and type from 1982-2017 were obtained from DFO (**Table 4**, Steve Cox-Rogers, pers. comm.). Estimates of the annual harvests were available for most FSC fisheries from 1996-2017. FSC catch estimates were not available

for the Kitsegass and Sustut fisheries prior to 1993 and catch estimates for Tsimshian fisheries in the Coastal to Kasiks stratum were substantial underestimates from 2003-09. There were also notable missing FSC estimates for a large portion of the Tsimshian fisheries in 1992 and 2001, Lake Babine FN fisheries in 1987 and the Bulkley River (Wet'suwet'en) fisheries in 1984 and 1990. There were concerns regarding the accuracy of the FSC estimates for the Kitsumkalum fishing area (Kasik-Terrace), the distribution of the GWWA catch between the mainstem Skeena fisheries above and below Hazelton, and FSC catch estimates for fisheries above the Babine fence. All of these issues, except the missing catch estimates for Kitsegass and Sustut fisheries, were addressed using the best available data from a variety of sources. The shaded cells in **Table 5** identify all the FSC catch estimates that were adjusted. The footnotes for **Table 5** (expanded below) describe how these adjustments were made:

1. Coastal to Kasiks catches for 2001 and 2003-09 were estimated by multiplying the adjusted total FSC catch by the average portion (24.7%) that Coastal to Kasiks catches represented of the reported catch for 1996-09.
2. Kasiks to Terrace (Kalum fishery) catches were expanded by 1.96 for all years except 1993 and 2009. This expansion factor was derived by comparing the Kalum catch numbers derived from Kalum fishing permits (the usual catch reporting method) with those derived from the more rigorous catch monitoring efforts conducted in 2007 and 2008 as part of a Treaty Related Measures (TRM) project.
3. Catch estimates for Fiddler to Hazelton and Hazelton to Lower Babine for 2000-09 were derived from tables provided by the Skeena Fisheries Commission (SFC). The average portion that the Fiddler to Hazelton catch represented of the total catch for these two strata (48%) was applied to the annual totals for these fisheries to derive the catch estimates for these two fisheries for each year from 1982-1999.
4. Previous catch estimates for the 1984-92 FSC fisheries above the Babine fence were replaced with values derived from Babine stock assessment tables provided by Steve Cox-Rogers.
5. Missing catch estimates for the Bulkley-Morice fishery in 1984 and 1990 were estimated using the average of the catch in adjacent years.
6. The total SFC catch for 2001 and 2003-09 needed to calculate the catch for the Coastal to Kaskis stratum was estimated by expanding the total reported catch for fisheries above Kaskis by 1.328 (i.e. $1/0.753$) because, on average, fisheries above Kasiks represented 75.3% of the total FSC harvest of Skeena sockeye.

Catch estimates from ESSR commercial fisheries were first conducted on the Skeena River in 1993. These fisheries are only permitted at specific locations within the Skeena watershed when managers determine that Escapements Surplus to Spawning Requirements (ESSR) can be harvested for some Skeena sockeye stocks. In recent years, additional commercial fishing opportunities have been provided to Skeena River First Nations through the transfer of sockeye allocations for Area 4 seine and gillnet licences to in-land "demonstration" fisheries. From 1993-2017, there have been 16 years when ESSR and/or demonstration fisheries have been conducted within the Skeena watershed and total harvests in these fisheries have ranged from 3,000 to over 780,000 sockeye (**Table 6**).

Demonstration and ESSR harvest data were provided to DFO as a total. The timing of these harvests was only available for some Demo and ESSR fisheries. The breakdown of these annual harvest estimates to weekly harvests for other fisheries were prorated to weekly harvest estimates using the year-specific estimates of the number of sockeye passing Tyee each week adjusted for the time required for sockeye to migrate from Tyee to each fishery. Ideally, these initial estimates of weekly catch will be replaced with the best available estimates from Skeena First Nations. If weekly estimates are not available for some in-river fisheries, additional information on the timing and duration of these fisheries could be used to improve the estimate of weekly catches. Adjustments to the timing of FSC harvests are not expected to have a significant impact on the harvest rate estimates because of the small relative magnitude and

protracted nature of these fisheries. The timing of the more substantial ESSR fisheries along the Skeena mainstem from Fiddler Creek to the Babine fence could have a significant impact on harvest rates for sockeye stocks migrating through these fisheries.

Fishing Patterns

Detailed information on fishing patterns (number of fishing days per week) for Skeena First Nation fisheries have not been obtained. Thus, the SSIR model currently runs on the assumption that weekly First Nation catches are distributed equally across all days in a week.

RUN RECONSTRUCTION MODEL

The theoretical basis of run reconstruction analysis for salmon stocks and fisheries are described in Starr and Hilborn (1988), Cave and Gazey (1994), Gazey and English (2000) and English et al. (2007). The SSIR model uses similar algorithms as those described in the 2007 PSARC approved run reconstruction model for Fraser Chinook (English et al. 2007). The sequential steps in the run reconstruction are described below:

1. read all catch, escapement, run-timing parameters and total daily abundance estimates derived from Tyee test fishery data;
2. estimate the daily escapement past the Tyee test fishery for each sub-stock;
3. starting with the first fishery in the lower Skeena, assign portions of the weekly catch to each sub-stock present in the fishery using the estimated constant daily harvest rate for all days in a week based on the assumption of equal vulnerability for all sub-stocks present during the week;
4. subtract the catch for each stock from the abundance of that stock that entered the fishery;
5. repeat steps 3 and 4 for each fishery moving upstream along the Skeena mainstem and into the tributaries;
6. total the catch and escapement estimates for each sub-stock and calculate annual estimates for the in-river harvest rates for each sockeye sub-stock and management group.

The model control worksheet has locations where the user can define the start and end years for the analysis and input files to be used for the run reconstruction analyses.

A more mathematically rigorous description of the above methods can be found in English et al. (2007).

Model Assumptions

The assumptions associated with the SSIR run reconstruction analyses model include:

- a. The sockeye sub-stocks included in the models adequately represent the run timing and total escapement for Skeena sockeye;
- b. The daily sockeye CPUE estimates from Tyee test fishery provides a reliable indication of the relative abundance sockeye entering the Skeena River;
- c. The escapement estimates and run-timing parameters available for Skeena sub-stocks can be used with the assumption of normal distributions for each stock to derive daily stock composition estimates for the run at Tyee;
- d. The fisheries and catch data included in the model adequately represent the timing and location of fisheries that harvest sockeye within the Skeena watershed; and
- e. All stocks are equally vulnerable to harvesting when present in a fishery, such that harvests of a stock are proportional to the relative abundance of that stock in that fishery during the fishing period.

Model Structure

In order to expedite these run reconstruction analyses, we have used a model structure that is very similar to that used for the Fraser Chinook run reconstruction model (i.e. a MS Excel model prepared using the Visual Basic programming language). The model contains a series of sub-routines and function calls to read input data from MS Excel worksheets, conduct the analyses and output results to MS Excel files.

The model includes the following sub-routines and functions:

Sub Reconstruction() - main program where all other sub-routines are called.

Sub Init() – reads the year range and input file names from the “Control” worksheet, opens the input files, creates the output files and writes the initial column headings into each output file.

Sub Read_Catch() – reads the weekly catch data for each FSC and ESSR fishery for the range of years included in the analysis.

Sub FishResSpawn() – reads the fishery residence times for each stock and determines the cumulative number of days between each fishery and the escapement area for each stock.

Sub Calc_Escape() – calculates the daily escapement for each stock for a specific year.

Sub Reconstruct() - conducts the run reconstruction analysis working backward through the fisheries building on the daily escapement estimates.

Function CalcHarvestRate() - calculates the weekly harvest rate for a given fishery based on the size of the reported catch, number of fishing days per week and the number of sockeye that escaped from that fishery.

Function gfs() – calculates the weekly catch for a given harvest rate. This function is used in the bisection algorithm to determine the weekly harvest rate that would result in the reported catch.

Sub OutputData() – writes the run reconstruction results to the various output files defined below.

Model input file name and description:

SkeenaReconV3_20180702.xlsm	<ul style="list-style-type: none"> Contains the macro while will run the model. User must define the start and end years. To run the macro select 'VIEW' → Macros → View Macros → Run.
SxCatch_FSC1982-17_2July2018.xlsx	<ul style="list-style-type: none"> Annual FSC catch in each area. Each year is a sheet. Also includes Tyeescapement.
SxCatch_ESSR1982-17_21March2018.xlsx	<ul style="list-style-type: none"> Annual harvests from ESSR and Domestic fisheries. Each year is a sheet.
SxTyeescapement1982-17_21March2018.xlsx	<ul style="list-style-type: none"> Residence times by stock in each fishery, and year specific standard deviations of run timing at Tyeescapement.
SkeenaSockeyeEscapement1982-17_20July2018.xlsx	<ul style="list-style-type: none"> Total escapement at Tyeescapement by stock.

Model output file name and description:

Catch.xls	<ul style="list-style-type: none"> Annual total catch by stock group (Babine and non-Babine), annual FSC catch by stock group and by stock, annual ESSR catch by stock group and stock.
FisheryHR.xls	<ul style="list-style-type: none"> Daily HR by fishery, week and year (separate worksheet for each year) computed to derive the weekly catch estimates for each stock.
WeeklyHR.xls	<ul style="list-style-type: none"> Mean estimated daily HR by fishery, week and year. These estimates are identical to those in FisheryHR.xls when fisheries are open 7 days a week.
ExploitRate.xls	<ul style="list-style-type: none"> Annual exploitation rates by stock group and by stock for each year
Reconstruction.xls	<ul style="list-style-type: none"> Annual summary of catch, for each stock and fishery (worksheet for each year).
Escape.xls	<ul style="list-style-type: none"> Annual escapement at spawning grounds by stock group and stock.
Unexplained.xls	<ul style="list-style-type: none"> Unexplained catch (not reconstructed) by fishery, week and year.

The internally documented source code for the current version of the SSIR model is provided in Appendix B.

Model Instructions

The model is written in Visual Basic within the Microsoft Excel program and is extremely simple to run and relatively easy to update. When distributed all necessary input files as previously described should be included and the names of all input files must match exactly to the names of the files as they appear within the "Control" worksheet in the "SkeenaReconV3_20180720.xlsm" file.

REVIEW AND VALIDATION PROCESS

Initial model validation and review has been the responsibility of the LGL analytical team. The LGL team ensures that escapement estimates for Skeena sockeye stocks are calculated in a consistent manner across the years and catch estimates are complete for each First Nation fishing area from inside Area 4 to the headwaters of the Skeena River. Model inputs are compared with model outputs to ensure that there are no major unexplained discrepancies between input and outputs for total catch and escapement (see Appendix A).

In past years, the review process for the SSIR model inputs and results have included discussion and presentations to stock assessment biologists from DFO's Prince Rupert office and Skeena River First Nations. For this version of the model we have circulated the model input files to members of the North Coast Skeena First Nations Stewardship Society (NCSFNSS) and participates in the Skeena First Nations Technical Committee process. The NCSFNSS consists of First Nations communities from the Lower Skeena and the BC North Coast and provides a forum for member nations to collaborate and remain informed of activities within the region. The current NCSFNSS membership is comprised of the Gitxaala, Gitga'at, Kitsumkalum, and Kitselas First Nations. Prior to 2017, Metlakatla and Haisla were also members of the NCSFNSS. After each model update the NCSFNSS' fisheries program director, Angela Addison (angela.addison@ncsfness.ca), coordinates with members of the NCSFNSS and SFNTC to meet and review the input data and results prior to final release. During this review period, assessment of the FSC and ESSR catch data is critical, as is ensuring all available escapement data is incorporated and accurate. Upon agreement of the input and output data the model is considered approved until the next model update.

RESULTS

Figure 1 - Figure 3 provide a sample of the run-timing and abundance of sockeye passing the Tye test fishery in the lower Skeena River for 2006-08. These figures also show the normally distributed run-timing curves for each of the major sub-stock groups and the resulting breakdown of the total Tye abundance for each of the major sub-stocks. These years were selected because 2006 is an example of one of the latest run-timing years, 2007 run-timing is close to the multi-year mean and 2008 is one of the earliest run-timing years. As indicated above, 2006 was the only year where we increased the duration of two sub-stocks (Pinkut and Fulton) to reflect the protracted nature of the sockeye return in that year and ensure that the stock composition estimates for Tye were consistent with the best available escapement estimates.

These figures clearly show the substantial overlap in the run-timing and long durations estimated for most Skeena sockeye stocks. These long durations are likely the result of having to use multiple years of DNA samples to obtain an adequate sample size for the relatively small non-Babine sockeye stocks. Steve Cox-Rogers 23 February 2012 memorandum included the following conclusion:

“The estimated peak dates of run entry for most Skeena sockeye sub-stocks, based on updated 2000-2010 DNA analysis, are not substantially different from past tagging assessments and the peak dates currently being used to assess stock impacts. The DNA data does suggest slightly wider “spreads” about the peaks for most stocks than currently assumed, and some apparent skewness/bi-modal variability to the timings may not be appropriately captured with the current practice of fitting normal curve approximations to the data. However, it is not clear how much of the shape variation is real or simply an artefact of sample size issues given the small number of DNA samples actually analyzed for some stocks in certain weeks (e.g. the tails of the test fishery). This, coupled with the fact that many non-Babine stocks are present in small proportions at Tye

in the first place, means the derived timings for the larger stocks are probably ok, but will always be uncertain for the smaller ones.”

While it is likely that the run duration for a single year would be shorter than the duration derived from samples collected over multiple years, the harvest rate estimates derived from these longer run durations will be less sensitive to uncertainties in the run and harvest timing for a given year. In the absence of more reliable year specific data on run-timing and duration, a conservative approach for estimating harvest rates for in-river fisheries is to use these longer durations.

Two model inputs that are critical for deriving reliable estimates of in-river harvest rates are catch estimates for all major fisheries and relative escapement estimates for each sub-stock. Our initial analyses of the available catch (**Table 4**) and escapement data revealed several deficiencies that have been specifically addressed for this report. Additional information on sockeye harvests from Tsimshian fisheries managers, SFC and DFO has been used to fill gaps and errors in the time-series of catch estimates for each of the major FSC sockeye fisheries (**Table 5**). The likely underestimation of escapements to the Bulkley-Morice watershed prior to 1989 was flagged as an issue that still needs to be addressed. In previous analysis, the escapement estimates for these CUs resulted in substantial overestimates of the harvest rates for the Moricetown fishery for 1982-1988 (see Table 9 in English et al. 2015). For all other sockeye stocks, the SSIR model has produced time-series of in-river harvest rates that are consistent with all the available data on the magnitude and location of fisheries and run-timing and geographic distribution of the sockeye sub-stocks. For example: in 2000 when ESSR fisheries were permitted to target surplus escapements for enhanced Babine stocks, in-river harvest rates were 43-45% for Pinkut and Fulton, 20-31% for the three run-timing groups of wild Babine sockeye, 23-27% for upper Skeena sockeye stocks and 1-5% for the early run lower Skeena sockeye stocks (**Table 7**). In years without ESSR fisheries (e.g. 2002 and 2005), in-river harvest rates are similar (16-23%) for all Babine sockeye sub-stocks, 14-17% for upper Skeena stocks and 2-7% for lower Skeena stocks. In 2017 there were no ESSR fisheries and for many stock groups, harvest rates are estimated to be the lowest ever (**Table 7**). The two lower Skeena stocks with notably shorter durations and least overlap with the Babine enhanced stocks are Lakelse and Zymoetz. The in-river harvest rates for these stocks were estimated to be 2-5% for most years.

DISCUSSION

Increasing levels of harvest in fisheries within the Skeena watershed have made these fisheries a significant component of the annual exploitation rates for many Skeena sockeye CUs. The location of stocks and fisheries within the watershed can result in harvest rates that differ substantially between the various sockeye CUs. In general, the harvest rates tend to be larger for the Babine sub-stocks because of the additional fisheries within the Babine watershed that target the surplus returns to the enhanced stocks. However, the highest harvest rates estimated for Skeena sockeye CUs were for the Bulkley/Morice CUs prior to 1989 and the Sustut CU after 2004. The bulk of the in-river harvest for these CUs occurred in terminal fisheries within the Bulkley or Sustut watersheds. The unusually high harvest rates estimated for sockeye in the 1982-88 Bulkley River fisheries has been cause for concern that the catch estimates, escapement estimates or both may be unreliable for this period. Sustut was the only other sockeye CU with unusually high harvest rates. The Sustut escapement estimates were derived from fence counts and are believed to be very reliable. The catch numbers are also believed to be reliable. Given the location of the fishery and relatively small annual returns since 2005, harvest rates estimates in the 50-75% range for in-river fisheries are not unrealistic. The Babine enhanced stocks are the only other Skeena sockeye stocks with in-river harvest rates above 35% and these high harvest rates only occur in years when major ESSR fisheries are conducted in Babine Lake near the Pinkut and Fulton spawning channels.

LITERATURE CITED

- Alexander, R., K.K. English, D. Peacock, and G. Oliver. 2010. Assessment of the Canadian and Alaskan Sockeye Stocks harvested in the northern boundary fisheries using run reconstruction techniques, 2004-08. Draft report for Pacific Salmon Comm. Northern Boundary Technical Committee.
- Cave J. and W.J. Gazey. 1994. A simulation model for fisheries on Fraser River sockeye salmon. *J. Fish. and Aquat. Sci.* 51:1535-1549.
- Cox-Rogers, S. 2012. Skeena sockeye sub-stock run-timing and abundance evaluated using Tye test fishery DNA: 2000-2010. Fisheries and Oceans Canada Memorandum dated 23 February 2012. 23 p. Available at: http://skeenawild.org/images/uploads/docs/Cox-Rogers_2012_Skeena_sockeye_run-timing.pdf.
- English, K.K., T. Mochizuki and D. Robichaud. 2012. Review of North and Central Coast Salmon Indicator Streams and Estimating Escapement, Catch and Run Size for each Salmon Conservation Unit. Report for Pacific Salmon Foundation and Fisheries and Oceans, Canada. 68 p.
- English, K.K., R. E. Bailey, and D. Robichaud. 2007. Assessment of Chinook returns to the Fraser River watershed using run reconstruction techniques, 1982-04. Canadian Science Advisory Secretariat, Research Document 2007/020. 76 p.
- English, K.K., R. Alexander, D. Peacock, and G. Oliver. 2005. Assessment of the Canadian and Alaskan Sockeye Stocks harvested in the northern boundary fisheries using run reconstruction techniques, 2002-03. Prepared for Pacific Salmon Comm. Northern Boundary Technical Committee. 59 p.
- English, K.K., W. J. Gazey, D. Peacock, and G. Oliver. 2004. Assessment of the Canadian and Alaskan Sockeye Stocks harvested in the northern boundary fisheries using run reconstruction techniques, 1982-2001. Pacific Salmon Comm. Tech. Rep. No. 13:93 p.
- English, K.K., D. Hall, and J.A. Taylor. 1985. The North Coast Salmon Tagging Project. Management information introductory volume, guide to figures and tables. 38 p.
- Volume A, 1982 Sockeye Salmon, 105 p.
- Volume B, 1982 Pink Salmon, 111 p.
- Volume C, 1983 Sockeye Salmon, 129 p.
- Volume D, 1984 Pink Salmon, 131 p.
- Unpublished report by LGL Limited for Fisheries and Oceans, Canada.
- Gazey, W.J. 2009. Interception of Skeena River Sockeye salmon stocks in northern boundary marine fisheries. Report for Skeena Wild Conservation Trust, Terrace, BC. 43 p.
- Gazey, W.J., and K.K. English. 2000. Assessment of sockeye and pink salmon stocks in the northern boundary area using run reconstruction techniques, 1982-95. *Can. Tech. Report Fish. Aquat. Sci.* No. 2320. 132 p.

- Noble, C. 2011. Assessing the performance of an in-river backward run reconstruction of Fraser River sockeye under biological uncertainty. MRM Thesis. Simon Fraser University. 53 p.
- Starr, P. and R. Hilborn. 1988. Reconstruction of harvest rates and stock contribution in gauntlet salmon fisheries: application to British Columbia and Washington sockeye (*Oncorhynchus nerka*). Can J. Fish. Aquat. Sci 45: 2216-2229.

Table 1. Fisheries, stocks and estimated residence time in days for the Skeena sockeye in-river run reconstruction analyses.

Stocks (Geographic CUs)	Short Name	Order	Aggregate	Data Quality ¹	CUs in Group	Fisheries														
						Coastal to Kasiks	Kasiks-Terrace	Terrace-Fiddler	Fiddler-Hazelton	Hazelton-L Babine	Babine below Fence	Babine Fence	Babine Lake	Pinkut Terminal	Fulton Terminal	Bulkley-Morice	Sustut			
Kluatantan/Kluayaz	Kluatan+	1	1	W	2	3	4	3	4	7										
Motase	Motase	2	1	W	1	3	4	3	4	7										
Sustut/Johanson/Spawning	Sustut+	3	1	G	3	3	4	3	4	7										1
Bear/Azuklotz/Asitka	Bear+	4	1	G	3	3	4	3	4	7										
Slamgeesh/Damshilgwit	Slamgeesh	5	1	G	2	3	4	3	4	7										
Sicintine	Sicintine	6	1	W	1	3	4	3	4	7										
Babine W Early	Babine-WE	7	2	G	1	3	4	3	4	7	4	1	3							
Babine W Middle	Babine-WM	8	2	G	1	3	4	3	4	7	4	1	3							
Babine W Late	Babine-WL	9	2	G	1	3	4	3	4	7	4	1	3							
Babine Pinkut	Babine-P	10	2	G	1	3	4	3	4	7	4	1	3	7						
Babine Fulton	Babine-F	11	2	G	1	3	4	3	4	7	4	1	3		7					
Swan/Stephans/Club	Swan+	12	1	G	3	3	4	3	4											
Bulkley/Maxan	Bulkley+	13	1	W	2	3	4	3	4											1
Morice/Atna	Morice+	14	1	G	2	3	4	3	4											1
Kitwanga	Kitwanga	15	1	G	1	3	4	3	2											
Zymoetz	Zymoetz	16	1	W	3	3	4	1												
Kalum	Kalum	17	1	W	1	3	4													
Lakelse	Lakelse	18	1	G	1	3	4													
Alastair	Alastair	19	1	G	1	3														
Johnston	Johnston	20	1	W	1	1														

¹ Babine W Early includes sockeye spawning in non-enhanced tributaries to Babine Lake.

² Babine W Middle includes the Tahlo/Morrison and Onerka CUs.

³ Babine W Late includes the Nikitkwa Lake CU.

⁴ Zymoetz includes three sockeye lake CUs in the Zymoetz watershed (Mcdonell, Aldrich and Dennis).

⁵ G= Good, W=Weak

Description of Fishery Locations:

- 1) Coastal = LaxKwalaams+Metlakatla (No Kitkatla+Hartley Bay) catch data
 - 2) Marine to Kasiks = Prince Rupert catch data
 - 3) Kasiks to Terrace = Kitsumkalum catch data
 - 4) Terrace to Fiddler = Kitselas catch data
 - 5) Fiddler to Hazelton = 0.95 * Skeena (e.g. Hazelton) catch data based on comments provided by J. Steward regarding location of catch + Gitanyow
 - 6) Hazelton to Lower Babine = 0.05 * Skeena (e.g. Hazelton) catch data based on comments provided by J. Steward regarding location of catch
 - 7) Kitsegass (L. Babine) = Babine catch data
 - 8) All Babine Lake = sum of Nat'oo'ten catch data
 - 9) Sustut = Takla catch data
 - 10) 1982-1992 GWWA Kitsegass catch data are included in the Hazelton to Lower Babine catch estimates
 - 11) 1982-2000 Moricetown data are the actual reported catch figures
 - 12) Note: Some missing 1992-1982 catch by fishery calculated as area-specific IFF total catch*Prop. IFF catch for 1993-2000
- These numbers are of unknown accuracy: all have been interpolated from Kerra Hoyseth's (2000) DFO review. USE WITH CAUTION

Table 2. Model input spawning escapement estimates for each sub-stock of Skeena sockeye salmon 1982-2017, where the estimates for Babine sub-stocks are prior to removals in fisheries at or above the Babine fence.

	Klutan+	Motase	Sustut+	Bear+	Slangeesh	Sicintine	Babine-WE	Babine-WM	Babine-WL	Babine-P	Babine-F	Swan+	Bulkley+	Morice+	Kirwanga	Zymoetz	Kalum	Lakelse	Alastair	Johnston	Skeena Agg.
1982	1,000	500	4,000	1,114	1,000	1,000	93,630	5,195	159,595	242,851	635,565	24,739	1,000	7,207	1,880	2,000	1,145	30,117	10,641	1,007	1,225,186
1983	1,000	500	4,000	1,080	1,000	1,000	26,965	9,226	103,027	197,094	550,081	27,179	1,000	9,610	2,720	10,000	916	19,251	14,561	1,410	981,620
1984	1,000	500	4,000	1,046	1,000	1,000	26,503	8,335	204,447	366,909	446,191	29,619	1,000	7,207	3,560	1,000	1,186	9,516	8,961	1,410	1,124,390
1985	1,000	500	4,000	3,061	1,000	1,000	75,649	17,696	623,637	598,005	833,056	42,766	1,000	4,805	4,400	1,200	2,371	36,315	8,961	1,813	2,262,236
1986	1,000	500	4,000	3,486	1,000	1,000	26,865	4,115	167,437	224,106	278,984	21,796	1,000	7,207	3,720	6,000	2,371	8,969	23,522	705	787,784
1987	1,000	1,500	4,000	11,383	1,000	1,000	38,206	16,344	237,400	678,196	337,706	39,367	1,000	9,610	3,040	6,000	4,743	5,305	11,201	1,209	1,409,209
1988	1,000	100	4,000	8,048	1,000	1,000	42,435	24,382	241,974	384,898	715,191	46,729	1,000	2,402	2,360	4,000	2,846	10,446	14,561	1,410	1,509,782
1989	1,000	400	4,000	1,856	1,000	1,000	18,412	8,005	132,563	239,008	734,328	25,847	1,000	6,727	1,680	3,500	4,269	7,679	16,725	604	1,209,603
1990	1,000	60	4,000	2,457	1,000	1,000	21,328	7,677	198,864	208,061	542,716	23,752	1,000	7,207	1,000	3,000	7,114	2,691	11,201	915	1,046,043
1991	1,000	300	4,000	8,402	1,000	1,000	58,719	26,200	432,582	450,512	208,305	42,776	1,000	48,049	1,000	1,200	7,114	13,454	24,642	1,226	1,332,482
1992	1,000	500	4,000	5,657	1,000	1,000	52,358	9,455	582,914	226,003	363,054	43,181	1,000	32,433	1,000	10,000	18,972	10,391	17,921	3,984	1,385,824
1993	1,000	400	4,000	6,028	1,000	1,000	16,646	28,016	595,377	539,525	557,861	58,234	1,000	26,427	1,000	15,000	16,600	13,673	16,801	6,742	1,906,331
1994	1,000	250	4,000	9,136	1,000	1,000	25,124	8,070	132,299	214,508	672,904	38,083	1,000	34,235	1,000	10,295	26,086	3,117	14,561	9,500	1,207,168
1995	1,000	250	4,000	12,244	1,000	1,000	79,679	7,351	69,506	603,037	977,436	17,933	1,000	42,043	500	10,295	23,122	27,455	19,042	12,258	1,910,149
1996	1,000	100	4,000	7,455	1,000	1,000	60,909	11,800	143,305	644,740	1,139,837	16,942	1,000	49,250	250	10,295	9,012	23,681	28,002	10,215	2,163,792
1997	1,000	220	4,000	4,186	1,000	1,000	92,245	43,995	129,975	224,874	595,521	15,826	1,000	28,829	250	10,295	17,312	3,446	26,882	8,172	1,210,027
1998	1,000	500	4,000	1,764	1,000	1,000	43,130	17,150	97,880	100,030	252,055	23,848	1,000	7,207	250	10,295	18,497	5,377	12,321	12,258	610,565
1999	1,000	500	4,000	9,713	1,000	1,000	63,692	24,171	155,040	141,127	222,106	31,871	1,000	18,018	250	10,295	19,740	7,856	2,240	16,344	730,964
2000	1,000	400	4,000	5,774	949	1,000	84,558	30,056	185,020	278,370	1,253,611	39,557	1,000	3,604	231	10,295	20,983	7,856	6,945	3,000	1,938,207
2001	1,000	200	4,000	5,181	855	1,000	232,802	85,174	617,401	208,119	840,765	47,244	1,000	4,805	221	10,295	12,528	10,335	12,097	9,000	2,104,020
2002	1,000	100	4,000	1,046	398	1,000	29,324	26,094	130,726	101,045	308,038	24,531	1,000	8,409	978	7,072	16,728	8,122	4,480	4,000	678,091
2003	1,000	2,000	4,992	13,031	430	1,000	55,028	86,328	99,284	224,981	704,737	97,886	1,000	12,012	3,377	9,106	42,883	8,122	30,242	10,100	1,407,540
2004	1,000	600	1,604	3,434	293	1,000	39,546	54,315	249,231	107,514	468,643	27,895	1,000	9,309	1,317	6,332	21,104	5,909	21,980	2,395	1,024,423
2005	1,000	290	1,175	1,892	216	1,000	25,141	33,241	163,178	137,759	349,880	16,070	1,000	9,611	937	6,888	13,506	6,268	13,717	2,395	785,163
2006	1,000	120	808	2,157	331	1,000	40,874	16,502	137,660	328,315	868,328	50,479	1,000	9,912	5,139	6,116	14,384	3,548	5,376	2,395	1,495,446
2007	1,000	300	2,469	2,422	366	1,000	52,862	13,376	100,762	239,849	643,632	22,619	1,000	16,096	245	3,800	18,818	6,585	24,642	2,395	1,154,240
2008	1,000	100	212	8,812	150	1,000	28,667	17,536	93,158	264,543	679,415	23,689	1,000	10,811	1,200	280	23,190	5,480	1,167	2,395	1,163,806
2009	1,000	410	540	7,683	161	1,000	20,503	20,163	93,791	157,109	380,436	39,842	1,000	13,760	3,047	3,400	29,493	7,530	24,082	2,395	807,346
2010	1,000	592	426	12,066	740	1,000	20,455	6,520	74,126	145,310	392,644	12,087	1,000	4,427	20,804	2,980	46,076	15,985	37,747	2,395	798,379
2011	1,000	880	1,884	8,146	687	1,000	64,622	60,814	152,998	252,595	578,977	25,476	1,000	5,534	2,366	4,418	28,186	34,781	23,298	2,395	1,251,057
2012	1,000	496	1,309	16,236	940	1,000	60,983	125,217	112,218	261,416	801,566	53,771	1,000	8,048	5,476	3,940	45,689	27,601	36,739	2,395	1,567,039
2013	1,000	496	1,366	6,546	93	1,000	8,760	8,084	39,602	30,754	225,127	19,826	1,000	7,773	828	3,320	37,361	27,616	19,938	2,395	442,883
2014	1,000	496	1,062	23,664	401	1,000	20,322	27,474	133,779	291,731	1,579,539	12,121	1,000	17,298	13,699	3,628	37,361	24,294	6,362	2,395	2,198,626
2015	1,000	496	1,464	5,799	1,281	1,000	52,239	102,501	140,168	160,302	1,056,480	75,804	1,000	24,921	4,636	9,818	37,361	4,740	18,818	2,395	1,702,220
2016	1,000	332	1,264	7,015	267	1,000	41,607	48,990	164,716	87,177	581,641	53,154	1,000	16,985	1,100	10,004	21,415	16,118	27,196	2,395	1,084,377
2017	1,000	1,124	2,818	17,220	500	1,000	28,036	21,353	84,732	134,113	537,429	32,000	1,000	18,611	375	16,026	34,932	12,000	5,779	2,395	952,443
After Fill																					
Average	1,000	442	3,362	5,319	1,000	1,000	50,658	22,134	214,920	291,396	578,934	32,751	1,000	14,664	2,213	6,275	14,348	11,932	16,156	4,373	1,273,680
Portion	0.001	0.000	0.003	0.004	0.001	0.001	0.040	0.017	0.169	0.229	0.455	0.026	0.001	0.012	0.002	0.005	0.011	0.009	0.013	0.003	
Before Fill																					
Average	0	441	1,528	5,080	444	0	50,658	22,134	214,920	291,396	578,934	8,709	0	23,796	3,078	4,939	8,990	12,490	14,283	2,395	1,244,215
Portion	0.001	0.000	0.003	0.004	0.001	0.001	0.041	0.018	0.173	0.234	0.465	0.007	0.001	0.019	0.002	0.004	0.007	0.010	0.011	0.002	

Table 3. Run-timing parameters used for each Skeena sockeye sub-stock group.

#	Stocks (Geographic CUs)	CUs in Group	Offset (days)	Duration (days)	Default SD (days)	2006 SD (days)	Source: Cox-Rogers (2012)				
							Group Name	Peak Week	Offset (days)	SD (weeks)	Revised SD (2015)
1	Kluatantan/Kluayaz	2	-10.5	84	14.0	14.0	Bulkley-Morice	72	-10.5	2.5	2.0
2	Motase	1	3.5	74	12.3	12.3	Motase	74	3.5	2.2	1.8
3	Sustut/Johanson/Spawning	3	-3.5	67	11.2	11.2	Sustut	73	-3.5	2.0	1.6
4	Bear/Azuklotz/Asitka	3	-3.5	67	11.2	11.2	Sustut	73	-3.5	2.0	1.6
5	Slamgeesh/Damshilgwit	2	-3.5	67	11.2	11.2	Sustut	73	-3.5	2.0	1.6
6	Sicintine	1	-3.5	67	11.2	11.2	Sustut	73	-3.5	2.0	1.6
7	Babine W Early	1	-10.5	67	11.2	11.2	Babine WE	72	-10.5	2.0	1.6
8	Babine W Middle	1	-3.5	67	11.2	11.2	Babine WM	73	-3.5	2.0	1.6
9	Babine W Late	1	10.5	67	11.2	11.2	Babine WL	75	10.5	2.0	1.6
10	Babine Pinkut	1	-3.5	67	11.2	14.0	Pinkut	73	-3.5	2.0	1.6
11	Babine Fulton	1	3.5	67	11.2	14.0	Fulton	73	3.5	2.0	1.6
12	Swan/Stephans/Club	3	-10.5	60	10.1	10.1	Swan+	72	-10.5	1.8	1.4
13	Bulkley/Maxan	2	-10.5	84	14.0	14.0	Bulkley-Morice	72	-10.5	2.5	2.0
14	Morice/Atna	2	-10.5	84	14.0	14.0	Bulkley-Morice	72	-10.5	2.5	2.0
15	Kitwanga	1	3.5	94	15.7	15.7	Kitwanga+	74	3.5	2.8	2.2
16	Zymoetz ¹	3	-17.5	47	7.8	7.8	Zymoetz	71	-17.5	1.4	1.1
17	Kalum	1	-3.5	84	14.0	14.0	Kalum-Bear	73	-3.5	2.5	2.0
18	Lakelse	1	-21.0	64	10.6	10.6	Lakelse+	64	-21	1.9	1.5
19	Alastair	1	-14.0	87	14.6	14.6	Alastair	71	-14	2.6	2.1
20	Johnston	1	-21.0	64	10.6	10.6	Lakelse+	64	-21	1.9	1.5

¹Zymoetz includes three sockeye lake CUs in the Zymoetz watershed (Mcdonell, Aldrich and Dennis)

Table 4. Initial annual estimates of the harvest of Skeena River sockeye by First Nations in FSC fisheries, 1982-2017.

Fishery	Group	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992	1994	1995	1996	1997	1998
Coastal	TTC	12,770	7,232	7,630	10,655	8,284	9,431	7,334	8,553	10,197	5,420		8,423	11,821	7,444	4,090	21,124
Marine to Kasiks	TTC	23,816	13,489	14,230	19,872	15,450	17,590	13,678	15,951	19,017	10,108		26,409	14,905	31,951	37,839	24,105
Coastal to Kasiks	TTC	36,586	20,721	21,860	30,526	23,734	27,021	21,012	24,504	29,214	15,528		34,832	26,726	39,395	41,929	45,229
Kasiks to Terrace	Kalum	4,582	2,595	2,738	3,823	2,972	3,384	2,631	3,069	3,658	1,945		3,665	5,177	5,927	4,656	1,951
Terrace to Fiddler	Kitselas	11,653	6,600	6,963	9,723	7,560	8,606	6,693	7,805	9,305	4,946		14,418	9,757	15,744	12,909	15,209
Fiddler to Hazelton	GWWA	102,600	79,420	128,250	114,000	85,500	76,000	71,250	85,500	83,600	83,600	66,500	35,307	42,668	21,058	32,880	50,369
Hazelton to L. Babine	GWWA	5,400	4,180	6,750	6,000	4,500	4,000	3,750	4,500	4,400	4,400	3,500	1,858	2,246	1,109	1,731	2,651
Babine below Fence	GWWA													6,439	2,802	1,637	195
Babine Fence	LBN	42,000	20,000	12,100	16,000	4,050		25,000	22,000	27,008	15,650	33,093	32,300	18,491	39,422	13,699	9,744
Babine Lake	LB/YECH																
Pinkut Terminal	LBN																
Fulton Terminal	LBN																
Bulkley-Morice	GWWA	4,500	6,450		4,000	22,450	20,296	4,250	1,450		13,000	15,138	12,629	23,912	14,453	15,512	3,674
Sustut	TAKLA												1,302		559	513	768
Total		207,320	139,966	178,660	184,072	150,766	139,307	134,586	148,828	157,185	139,069	118,231	136,311	135,416	140,469	125,466	129,790

Fishery	Group	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
Coastal	TTC	3,073	37,157		29,000	3,123	4,356	10,376	9,607	21,685	27,134	9,100	19,503	34,847	17,038	6,667	10,273	31,332	29,813	13,900
Marine to Kasiks	TTC	14,644	27,600		20,000		4,840	2,507		17,022	5,428									
Coastal to Kasiks	TTC	17,717	64,757		49,000		9,196	12,883		38,707	32,562									
Kasiks to Terrace	Kalum	2,294	1,544		4,905	6,075	7,056	4,360	5,803	4,168	5,966	10,763	5,500	7,196	4,937	935	5,559	6,744	7,540	2,751
Terrace to Fiddler	Kitselas	10,131	13,245	13,479	13,680	11,337	12,550	9,098	9,749	9,338	8,535	9,465			1,139	3,686			4,305	390
Fiddler to Hazelton	GWWA	51,854	39,178	21,594	31,536	22,618	25,931	22,040	36,251	14,825	27,616	14,882	39,421	32,161	24,064	9,385	26,344	32,509	25,387	10,579
Hazelton to L. Babine	GWWA	2,730	16,915	21,943	22,311	35,086	36,556	42,304	39,409	13,280	29,788	17,509	36,389	29,688	22,212	8,664	24,317	28,601	21,992	13,576
Babine below Fence	GWWA	3,366	2,658	5,000	1,091	533	333	1,273		16,643		107	794	1,279	168	233	321	330		
Babine Fence	LBN	23,220	23,300	24,080	24,785	32,000	31,441	33,117	38,600	36,070	48,901	43,957	36,252	37,707	36,723	4,500	32,391	40,137	36,893	7,533
Babine Lake	LB/YECH												2,005	2,762	1,680	39	1,411	6,450	7,619	
Pinkut Terminal	LBN																	1,089	561	1,615
Fulton Terminal	LBN																			
Bulkley-Morice	GWWA	675	1,905	1,289	331	456	278	197	2,085	219	2,391	1,644	333	17	121	602	43	1,976	810	200
Sustut	TAKLA	868	1,050	470	811	1,954	567	862	632	419	526	992	543	624	506	602	135	738	561	817
Total		112,855	164,552	87,856	148,450	110,060	123,909	126,134	132,530	133,669	156,285	99,319	121,237	111,434	90,411	26,099	94,207	118,574	105,668	37,461

Table 5. Revised annual estimates of the harvest of Skeena River sockeye by First Nations in FSC fisheries, 1982-2017.

Fishery	Group	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992	1994	1995	1996	1997	1998
Coastal	TTC	12,770	7,232	7,630	10,655	8,284	9,431	7,334	8,553	10,197	5,420	11,173	8,423	11,821	7,444	4,090	21,124
Marine to Kasiks	TTC	23,816	13,489	14,230	19,872	15,450	17,590	13,678	15,951	19,017	10,108	18,959	26,409	14,905	31,951	37,839	24,105
Coastal to Kasiks ¹	TTC	36,586	20,721	21,860	30,526	23,734	27,021	21,012	24,504	29,214	15,528	30,132	34,832	26,726	39,395	41,929	45,229
Kasiks to Terrace ²	Kalum	8,993	5,093	5,373	7,503	5,834	6,642	5,165	6,023	7,181	3,817	9,045	7,194	10,161	11,633	9,139	3,829
Terrace to Fiddler	Kitselas	11,653	6,600	6,963	9,723	7,560	8,606	6,693	7,805	9,305	4,946	6,982	14,418	9,757	15,744	12,909	15,209
Fiddler to Hazelton ³	GWWA	52,107	40,335	65,134	57,897	43,423	38,598	36,186	43,423	42,458	42,458	33,773	17,931	21,670	10,695	16,699	25,581
Hazelton to L. Babine ³	GWWA	55,893	43,265	69,866	62,103	46,577	41,402	38,814	46,577	45,542	45,542	36,227	19,234	23,244	11,472	17,912	27,439
Babine below Fence	GWWA													6,439	2,802	1,637	195
Babine Fence ⁴	LBN	42,000	20,000	20,500	17,500	23,500	20,296	25,000	22,000	22,000	20,800	73,789	32,300	18,491	39,422	13,699	9,744
Babine Lake ⁴	LB/YECH																
Pinkut Terminal	LBN																
Fulton Terminal	LBN																
Bulkley-Morice ⁵	GWWA	4,500	6,450	5,225	4,000	4,125	4,125	4,250	1,450	7,225	13,000	15,138	12,629	23,912	14,453	15,512	3,674
Sustut	TAKLA												1,302		559	513	768
Total ⁶		211,731	142,464	194,921	189,253	154,753	146,690	137,119	151,782	162,924	146,091	205,087	139,840	140,400	146,175	129,949	131,668

Fishery	Group	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
Coastal	TTC	3,073	37,157	15,035	29,000	18,503	19,396	18,748	22,048	16,006	20,371	18,191	21,922	21,149	16,738	4,421	17,046	22,077	19,702	5,980
Marine to Kasiks	TTC	14,644	27,600	16,897	20,000	20,795	21,799	21,071	24,779	17,989	22,895	20,445	24,638	23,769	18,812	4,969	19,158	24,813	22,143	6,721
Coastal to Kasiks ¹	TTC	17,717	64,757	31,932	49,000	39,297	41,195	39,820	46,827	33,995	43,266	38,636	46,559	44,918	35,550	9,390	36,204	46,890	41,845	12,701
Kasiks to Terrace ²	Kalum	4,503	3,031	6,329	9,627	11,924	13,849	8,558	11,390	9,476	9,856	10,763	10,795	14,124	9,690	1,835	10,911	13,237	14,799	2,751
Terrace to Fiddler	Kitselas	10,131	13,245	13,479	13,680	11,337	12,550	9,098	9,749	9,338	8,535	9,465	10,795	14,124	9,690	1,835	10,911	13,237	14,799	390
Fiddler to Hazelton ³	GWWA	26,336	39,178	21,594	31,536	22,618	25,931	22,040	36,251	14,825	27,616	14,882	39,421	32,161	24,064	9,385	26,344	32,509	25,387	10,579
Hazelton to L. Babine ³	GWWA	28,248	16,915	21,943	22,311	35,086	36,556	42,304	39,409	13,280	29,788	17,509	36,389	29,688	22,212	8,664	24,317	28,601	21,992	13,576
Babine below Fence	GWWA	3,366	2,658	5,000	1,091	533	333	1,273		16,643		107	794	1,279	168	233	321	330	0	0
Babine Fence ⁴	LBN	23,220	23,300	24,080	24,785	32,000	31,441	33,117	38,600	36,070	48,901	58,597	36,252	37,707	36,723	4,500	32,391	40,137	36,893	7,533
Babine Lake ⁴	LB/YECH												2,005	2,762	1,680	39	1,411	6,450	7,619	0
Pinkut Terminal	LBN																	1,089	561	1,615
Fulton Terminal	LBN																			
Bulkley-Morice ⁵	GWWA	675	1,905	1,289	331	456	278	197	2,085	219	2,391	1,644	333	17	121	602	43	1,976	810	200
Sustut	TAKLA	868	1,050	470	811	1,954	567	862	632	419	526	992	543	624	506	602	135	738	561	817
Total ⁶		115,064	166,038	126,117	153,172	155,206	162,701	157,268	184,943	134,266	170,878	152,596	183,887	177,404	140,404	37,085	142,988	185,194	165,266	50,162

¹ Coastal to Kasiks catches for 2001 and 2003-09 were estimated by multiplying the adjusted total FSC catch by the average portion (24.7%) that Coastal to Kasiks catches represented of the reported catch for 1996-2009.

² Kasiks to Terrace catch was expanded by 1.96 for all years except 1993, 2009 and 2017, based on results from TRM studies 2007-08.

³ Revised catch estimates for Fiddler-Hazelton and Hazelton-L. Babine fisheries (dark green shaded value are from SFC)

⁴ Yellow highlight values are revised FSC estimates from Steve Cox-Rogers' Babine stock assessment tables.

⁵ Highlighted values are averages of adjacent FSC catch estimates. Catch estimates for 1986 and 1987 were too large (>20K) to be correct given escapement estimates (<5K).

⁶ Total catch estimates for 2001 and 2003-14 were estimated by expanding the total reported catch for fisheries above Kasiks by 1.32 (i.e. 1/0.747) because, on average from 1996-09, fisheries above Kasiks represent 74.7% of the total FSC harvest of Skeena sockeye.

Table 6. Annual estimates of the harvest of Skeena River sockeye by First Nations in ESSR and in-land “Demonstration” fisheries, 1993-2017. Years with no ESSR harvests are not shown.

Fishery	Group	1993	1994	1995	1996	1997	2000	2001	2006	2007	2008	2010	2011	2012	2014	2015	2016
Kasiks to Terrace	TTC	3,919	267	14,720	10,454	5,093	14,998	0	0	0	0	0	595	0	0	0	0
Terrace to Fiddler	TTC		3,742	2,878	0	0	7,770	0	0	0	0	0	0	0	0	0	0
Fiddler to Hazelton	GWWA	6,500	16,123	79,943	120,340	72,570	139,345	38,957	81,790	26,768	67,289	1,427	21,529	52,186	40,611	1,038	7,409
Hazelton to L. Babine	GWWA	6,000	0		12,038	5,010	180,140	26,112	0	0	0	0	0	2,795	0	0	0
Babine below Fence	GWWA	12,000	6,253	31,880	39,662	18,793	9,995	152,230	92,347	10,003	41,715	0	0	0	0	0	0
Babine Fence	LBN	104,340	12,000	45,000	40,421	69,932	56,203	138,240	138,180	13,777	104,585	1,611	32,848	59,654	55,160	1,118	20,426
Babine Lake	TTC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Pinkut Terminal	LBN	0	0	0	92,597	36,982	65,821	32,220	0	0	37,388	0	0	107,295	30,051	0	0
Fulton Terminal	LBN	0	3,900	35,000	236,080	37,545	310,132	315,220	80,820	0	50,506	0	185,393	108,830	424,505	289,637	77,839
Bulkley-Morice	GWWA	0	0	0	0	1,208	0	0	0	0	0	0	0	0	0	0	0
Total		132,759	42,285	209,421	551,592	247,133	784,404	702,979	393,137	50,548	301,483	3,038	240,365	330,760	550,327	291,793	105,674

Table 7. Harvest rate estimates for each Skeena sockeye sub-stock group in First Nation fisheries conducted in and adjacent to the Skeena watershed, 1982-2017.

Sub-stock	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999
Kluatan+	14.9	21.0	12.3	8.2	16.6	8.9	11.0	16.4	11.3	8.5	8.8	6.3	10.5	10.8	8.2	14.3	22.8	13.9
Motase	11.5	10.2	13.4	7.0	13.9	7.9	6.6	9.4	11.5	7.8	7.8	5.5	8.9	7.8	8.9	13.1	17.8	10.8
Sustut+	12.7	13.3	12.7	7.1	14.4	8.1	7.7	11.6	10.8	7.9	8.1	5.8	32.3	9.9	20.1	26.1	34.2	28.5
Bear+	12.7	13.3	12.7	7.1	14.4	8.1	7.7	11.6	10.8	7.9	8.1	5.8	9.2	9.9	9.6	15.3	19.4	11.8
Slamgeesh	12.7	13.3	12.7	7.1	14.4	8.1	7.7	11.6	10.8	7.9	8.1	5.8	9.2	9.9	9.6	15.3	19.4	11.8
Sicintine	12.7	13.3	12.7	7.1	14.4	8.1	7.7	11.6	10.8	7.9	8.1	5.8	9.2	9.9	9.6	15.3	19.4	11.8
Babine-WE	18.9	22.3	13.8	8.6	19.3	10.3	12.3	18.1	13.0	10.1	16.0	20.4	15.5	16.3	13.8	23.0	24.6	18.6
Babine-WM	16.3	15.8	14.3	7.9	17.4	9.6	9.7	13.8	12.9	9.5	14.2	19.7	14.0	15.8	15.1	24.7	21.4	16.0
Babine-WL	13.2	9.2	15.8	7.8	16.1	8.9	6.8	8.9	14.2	9.2	12.2	12.0	12.1	9.0	11.6	16.5	18.1	13.1
Babine-P	16.3	15.8	14.3	7.9	17.4	9.6	9.7	13.8	12.9	9.5	14.2	19.7	14.0	15.8	27.5	40.2	21.4	16.0
Babine-F	14.5	11.7	15.0	7.7	16.5	9.2	8.0	10.9	13.4	9.2	13.1	17.0	13.9	16.3	32.5	28.0	19.2	14.3
Swan+	9.0	11.4	7.0	4.5	9.2	5.5	5.5	8.7	6.6	4.5	5.5	4.7	7.2	8.4	8.2	13.6	15.3	8.4
Bulkley+	37.5	42.5	47.5	32.0	37.0	31.3	41.5	20.8	39.3	23.5	33.6	24.9	30.1	37.8	19.3	39.6	38.8	12.1
Morice+	37.5	42.5	47.5	32.0	37.0	31.3	41.5	20.8	39.3	23.5	33.6	24.9	30.1	37.8	19.3	39.6	38.8	12.1
Kitwanga	5.7	5.1	5.5	3.3	6.5	4.0	3.3	4.5	5.9	3.2	4.0	3.9	5.6	3.8	4.9	7.7	11.7	5.5
Zymoetz	3.7	3.2	1.4	1.5	3.1	2.2	2.2	3.4	2.4	1.2	2.5	3.3	4.3	2.8	2.0	3.9	8.1	3.5
Kalum	3.2	2.4	1.9	1.5	3.0	2.1	1.7	2.5	2.7	1.2	2.4	3.1	3.4	2.3	2.2	4.0	7.0	2.7
Lakelse	3.4	2.8	1.3	1.3	2.9	2.1	2.1	3.1	2.2	1.1	2.4	3.2	4.0	2.7	1.7	3.5	7.1	3.1
Alastair	2.5	2.0	1.2	1.0	2.1	1.6	1.5	2.2	1.7	0.8	1.7	2.2	2.9	1.4	1.2	2.7	6.3	2.2
Johnston	0.9	0.7	0.3	0.3	0.6	0.5	0.5	0.8	0.5	0.2	0.5	0.7	1.0	0.5	0.4	0.8	2.1	0.8
Total	14.6	12.7	14.8	7.6	16.1	9.3	8.4	11.1	13.4	9.5	12.9	15.3	13.6	15.6	26.2	26.8	18.6	13.7

Sub-stock	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
Kluatan+	19.5	4.9	16.3	9.7	16.4	23.2	13.7	7.0	11.7	11.8	21.8	13.3	10.4	11.9	7.4	9.8	15.3	3.0
Motase	20.5	7.6	16.8	8.6	12.0	14.0	13.2	8.5	13.6	10.2	18.3	11.6	9.3	7.7	6.4	7.8	10.9	4.6
Sustut+	38.9	15.9	30.5	36.3	35.8	56.3	56.7	20.7	72.7	67.7	71.1	33.7	36.0	37.8	17.9	41.9	39.6	25.6
Bear+	23.1	6.2	16.4	8.8	13.2	16.2	13.8	7.6	13.2	10.7	19.7	12.1	10.3	8.7	7.4	8.4	12.2	4.2
Slamgeesh	23.1	6.2	16.4	8.8	13.2	16.2	13.8	7.6	13.2	10.7	19.7	12.1	10.3	8.7	7.4	8.4	12.2	4.2
Sicintine	23.1	6.2	16.4	8.8	13.2	16.2	13.8	7.6	13.2	10.7	19.7	12.1	10.3	8.7	7.4	8.4	12.2	4.2
Babine-WE	25.3	22.6	20.6	13.0	20.2	26.9	22.5	12.4	25.4	22.3	29.4	20.2	20.7	13.7	14.3	13.8	26.7	4.5
Babine-WM	27.5	24.6	20.4	11.7	17.0	21.4	29.9	13.7	29.3	19.1	26.2	18.7	18.5	10.5	13.2	12.0	21.1	5.3
Babine-WL	19.9	19.8	20.7	10.8	13.5	15.8	31.7	16.0	24.1	16.1	22.1	15.8	11.7	8.1	7.7	9.8	14.5	5.4
Babine-P	45.6	38.9	20.4	11.7	17.0	21.4	27.6	13.7	41.1	19.1	26.2	18.7	56.3	10.5	22.8	12.7	21.7	6.5
Babine-F	46.4	61.1	20.4	11.1	14.9	18.0	37.2	15.3	35.7	17.3	23.8	45.6	26.8	9.0	37.2	38.5	29.7	5.6
Swan+	13.0	3.3	12.8	5.9	9.8	10.9	10.8	5.4	8.8	8.5	15.3	9.7	8.9	6.6	6.2	6.6	10.5	1.8
Bulkley+	27.8	14.5	15.5	9.0	12.3	13.6	22.7	6.6	21.2	17.7	21.4	10.4	9.9	13.6	6.2	12.0	13.7	2.2
Morice+	27.8	14.5	15.5	9.0	12.3	13.6	22.7	6.6	21.2	17.7	21.4	10.4	9.9	13.6	6.2	12.0	13.7	2.2
Kitwanga	7.8	3.7	11.9	5.3	7.4	7.7	6.9	5.9	7.2	7.3	11.1	7.4	5.5	4.4	3.7	5.2	7.7	2.2
Zymoetz	3.7	1.6	6.4	3.4	5.8	6.6	3.9	2.6	3.4	5.6	7.5	4.6	2.7	2.7	2.2	3.6	5.5	0.2
Kalum	3.6	1.6	7.0	3.4	5.1	5.7	3.4	3.2	3.7	5.3	6.7	4.2	2.6	2.5	2.0	3.4	4.8	1.2
Lakelse	3.7	1.3	5.7	3.2	5.6	6.5	4.1	2.5	3.3	5.5	6.9	4.3	2.6	2.5	2.1	3.4	5.1	0.2
Alastair	3.4	1.1	5.0	2.3	3.6	4.8	2.9	1.9	2.6	3.9	5.1	2.9	1.9	2.0	1.4	2.4	3.2	0.4
Johnston	1.3	0.3	1.3	0.7	1.2	1.7	1.0	0.5	0.8	1.2	1.7	0.9	0.6	0.6	0.5	0.8	1.1	0.0
Total	39.7	36.0	19.2	10.3	14.2	17.3	32.0	13.9	33.6	16.2	21.6	29.5	27.4	8.1	29.8	26.6	22.0	5.1

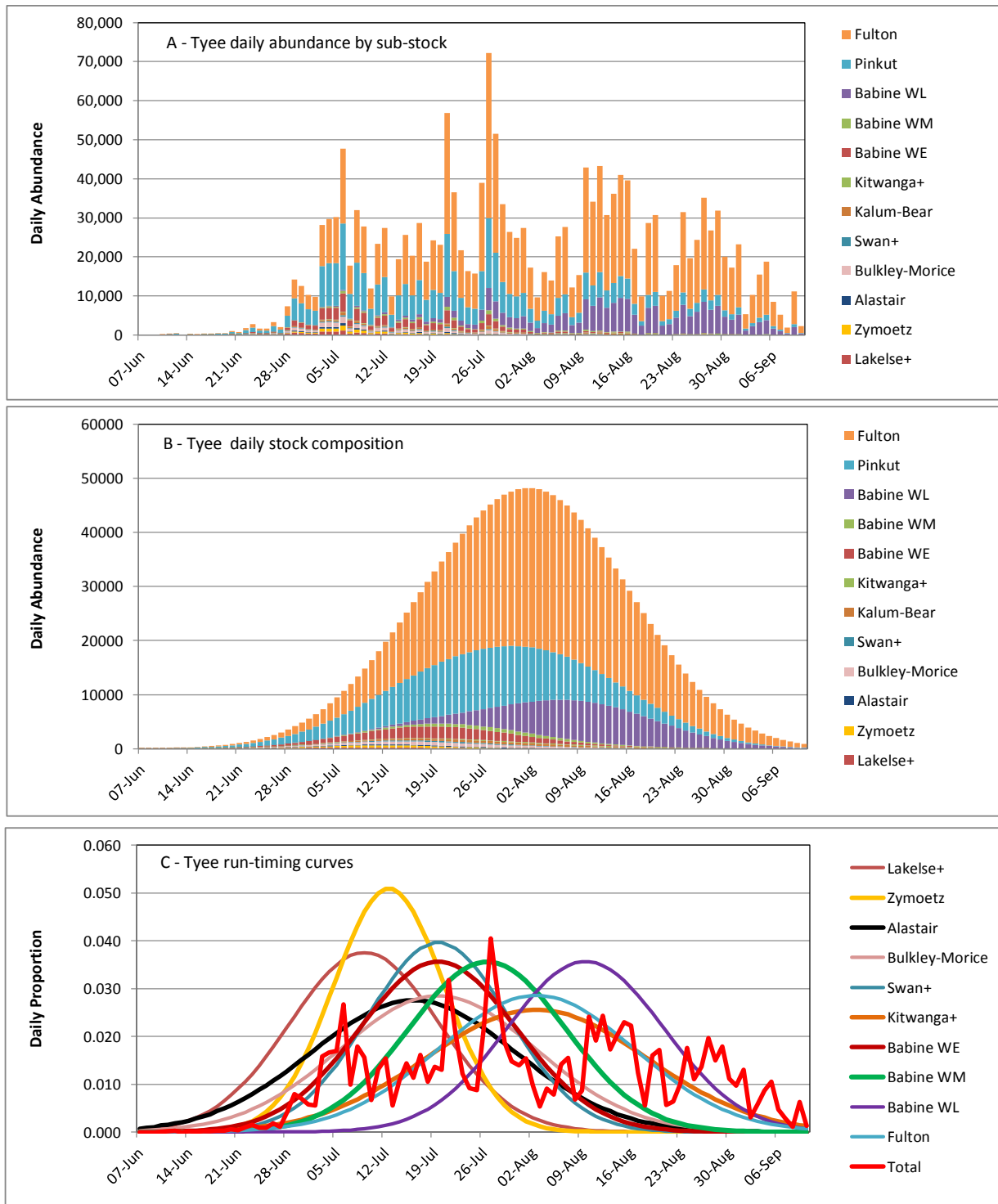


Figure 1. Estimated Tyee daily abundance, stock composition and run-timing curves for Skeena sockeye stocks in 2006 (late run-timing year).

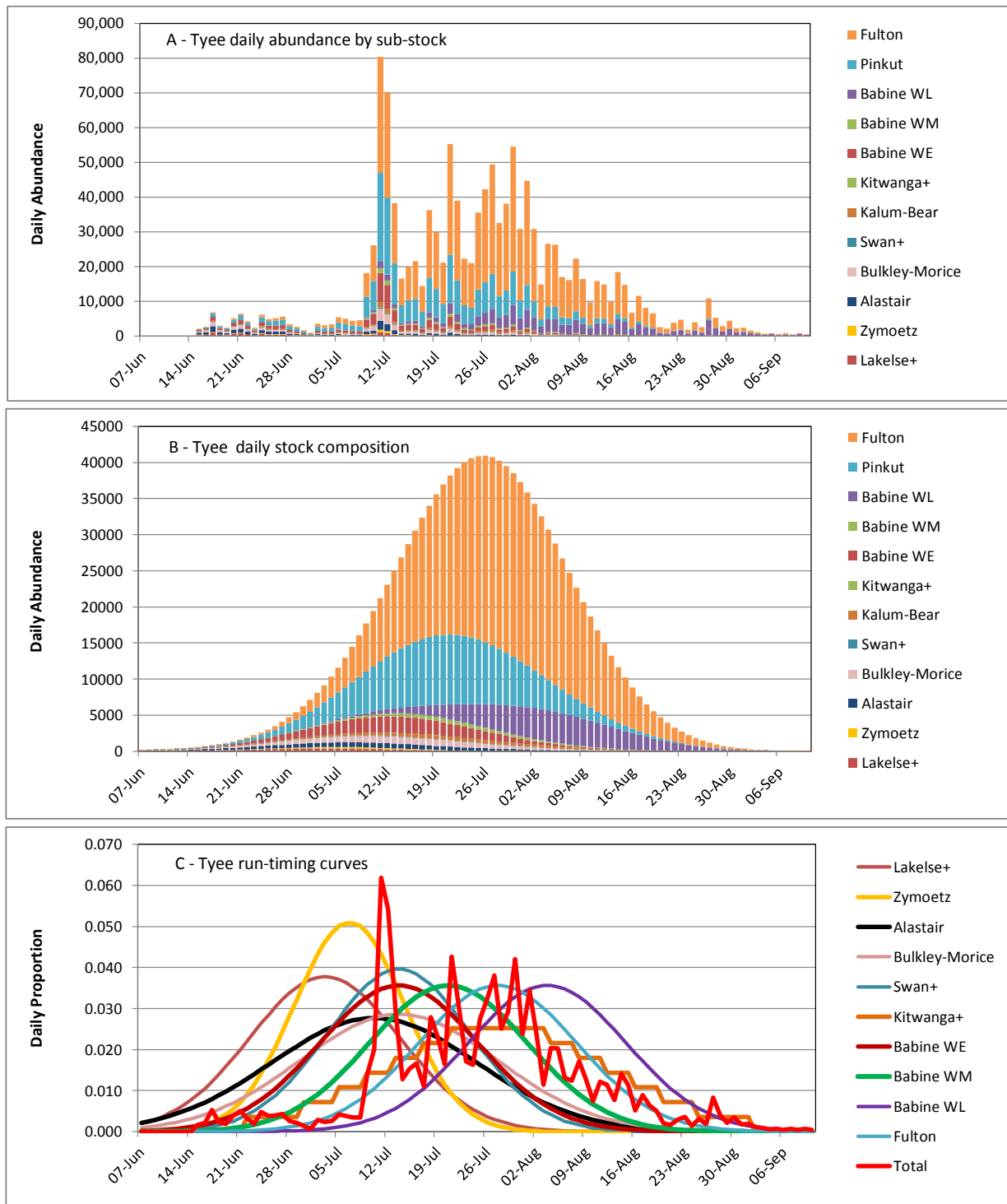


Figure 2. Estimated Tyee daily abundance, stock composition and run-timing curves for Skeena sockeye stocks in 2007 (average run-timing year).

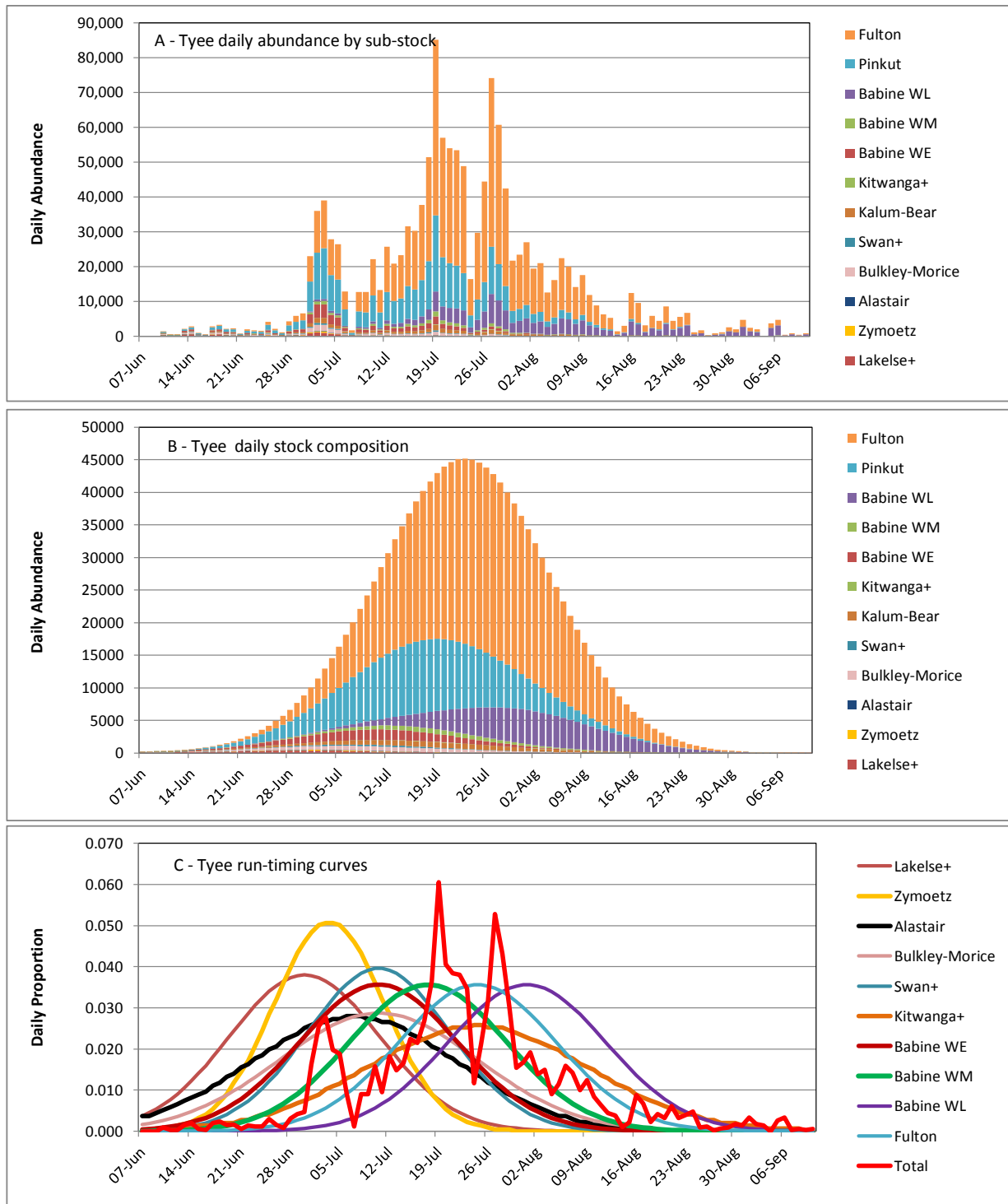


Figure 3. Estimated Tye daily abundance, stock composition and run-timing curves for Skeena sockeye stocks in 2008 (early run-timing year).

APPENDIX A**Summary of catch and escapement results for 1982-2017**

Table A1 provides a summary of the annual escapement for each sub-stock and a comparison of the total output escapement and input escapement from **Table 2**. The difference between these two total “escapement” values is because the **Table 2** totals include catches in terminal fisheries for each of the Babine sub-stocks whereas the output escapement estimates do not include these catches and the escapement estimates for non-Babine stock used to derive the initial abundance at Tyee are larger than the sum of the escapement estimates for non-Babine CUs included in **Table 2**. Consequently, in years without major terminal fisheries for Babine stocks, the model output escapement is larger than the input totals from **Table 2** and in years with major terminal fisheries, the model output escapement estimates are smaller than the input totals from **Table 2**.

Table A2 provides a summary of the annual estimates of the river entry abundance at Tyee for each sub-stock derived by combining the stock proportions from **Table 2**, the relative run-timing data for each sub-stock and the daily estimates of total sockeye abundance at Tyee. The total daily sockeye abundance at Tyee was computed by combining the total river entry escapement estimates (as used in the Northern Boundary run reconstruction analyses) with the daily Tyee test fishery CPUE data. The “Tyee Estimates” at the bottom of **Error! Reference source not found.** are provided to show that the sum of the sub-stock specific river-entry escapement estimates were always within a few fish of the input values for the total “Tyee Estimate”.

Table A3 provides the in-river catch by sub-stock calculated as the difference between the sockeye abundance at Tyee (Table A2) and the escapement (Table A1). The total annual catch estimated for all Skeena stocks is compared with the input catch at the bottom of Table A3 to show that the model accounts for essentially all the catch in most years.

Table A1. Total sockeye escapement results from Run Reconstruction analysis (output file="Escape.xls").

Stock	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999
Kluatan+	1031	1019	816	1308	1012	978	1402	1163	1209	974	983	1334	1015	1048	1404	1044	1014	837
Motase	527	496	522	492	526	1604	96	405	61	314	522	396	252	304	112	222	464	540
Sustut+	3902	3939	3804	4017	4049	3896	4201	4027	4233	3900	4260	4114	3818	3741	4224	3523	3424	3727
Bear+	1087	1064	995	3074	3529	11086	8453	1869	2600	8192	6024	6201	8822	11450	7816	3743	1551	9169
Slamgeesh	975	985	951	1004	1012	974	1050	1007	1058	975	1065	1029	966	935	1048	894	879	944
Sicintine	975	985	951	1004	1012	974	1050	1007	1058	975	1065	1029	966	935	1048	894	879	944
Babine-WE	89982	25596	21612	88442	26014	34852	54658	19641	23749	55332	49767	17707	23573	75430	72918	83673	41140	50132
Babine-WH	4861	8824	7778	17618	4020	15664	25051	7858	7939	25093	9400	24576	7378	6425	11617	34997	14706	21745
Babine-WL	183983	101286	231730	638099	184514	278720	224025	139811	193689	507456	583591	554983	140009	116629	179738	150728	101350	189032
Babine-P	227251	188506	342359	595388	218922	649976	395453	234599	215171	431482	224675	473274	196122	527084	542131	141899	85774	126961
Babine-F	625145	531082	450974	781334	276939	349123	648845	707571	527368	205654	350171	471551	620369	959604	885660	465788	216226	223235
Swan+	26197	28616	26344	50121	23698	36773	63354	30048	27901	42861	47122	72221	38875	18390	20589	15828	24932	28244
Bulkley+	757	742	489	968	764	738	922	1101	827	814	716	1069	792	731	1234	736	804	854
Moric+	8867	11918	6082	8521	8659	10129	6131	9005	11937	49685	34041	40438	37130	48232	78640	33504	8745	15973
Kitwanga	2583	3141	4513	5227	4773	4056	2710	2123	1194	1384	1264	1125	1269	980	404	351	390	372
Zymoetz	2608	12368	710	1624	6771	5558	7491	4842	3685	1146	9908	24520	11314	13611	17114	13900	14732	8390
Kalum	1321	1053	1254	2788	2793	5128	3523	5100	8515	7732	20925	19318	28101	27803	11750	19523	20829	21101
Lakelse	39931	29304	6427	70925	10938	5875	22728	13304	4309	14113	9545	25648	3739	35653	49671	4871	8355	6198
Alastair	12776	19709	7576	14441	28205	12005	25916	24991	16418	26052	18003	25955	16392	22425	47232	33103	16054	2023
Johnston	1371	2193	962	3577	880	1360	3119	1073	1490	1296	3730	12973	11746	16274	21275	11878	20070	13209
Total	1236130	972824	1116850	2289973	809030	1429470	1500177	1210544	1054411	1385432	1376778	1779459	1152649	1887685	1956076	1021098	582319	723632
Input	1225186	981620	1124390	2262236	787784	1409209	1509782	1209603	1046043	1332482	1385824	1906331	1207168	1910149	2163792	1210027	610565	730964
Difference	10944	-8796	-7540	27737	21246	20261	-9605	941	8368	52950	-9046	-126872	-54519	-22464	-207716	-188929	-28246	-7332

Table A2. Incoming abundance to Ytee from Run Reconstruction analysis (output file="Stock_CalcEntry.xls").

Stock	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999
Kluatan+	1212	1290	931	1424	1213	1073	1575	1391	1364	1064	1077	1422	1134	1175	1529	1218	1313	972
Motase	595	552	603	529	611	1741	103	447	69	341	566	420	277	330	123	255	565	606
Sustut+	4470	4544	4357	4325	4728	4240	4551	4556	4747	4232	4635	4369	5636	4152	5286	4766	5204	5212
Bear+	1245	1227	1139	3310	4121	12066	9157	2114	2916	8890	6554	6585	9712	12709	8644	4420	1926	10400
Slamgeesh	1117	1136	1089	1081	1182	1060	1138	1139	1187	1058	1159	1092	1063	1038	1160	1056	1091	1071
Sicintine	1117	1136	1089	1081	1182	1060	1138	1139	1187	1058	1159	1092	1063	1038	1160	1056	1091	1071
Babine-WE	110884	32933	25075	96749	32223	38850	62347	23991	27306	61549	59237	22258	27882	90126	84546	108618	54543	61623
Babine-WH	5805	10481	9080	19133	4864	17324	27741	9118	9110	27722	10955	30603	8578	7630	13683	46462	18717	25880
Babine-WL	211888	111550	275203	691773	219850	305999	240454	153419	225817	558669	664449	631011	159355	128218	203313	180484	123746	215754
Babine-P	271383	223905	399679	646566	264911	718863	437920	272229	246899	476675	261855	589342	228017	625918	747609	237484	109171	151101
Babine-F	730779	601790	530567	846709	331835	384354	705336	794113	608970	226572	402770	568353	720602	1146593	1312704	647223	267768	260535
Swan+	28795	32311	28328	52481	26092	38902	67024	32901	29876	44888	49851	75750	41878	20078	22439	18310	29438	30849
Bulkley+	1212	1290	931	1424	1213	1073	1575	1391	1364	1064	1077	1422	1134	1175	1529	1218	1313	972
Moric+	14184	20723	11580	12536	13745	14741	10481	11373	19680	64963	51237	53819	53157	77489	97396	55488	14292	18178
Kitwanga	2740	3308	4775	5405	5105	4224	2803	2222	1268	1429	1318	1171	1343	1019	425	381	442	394
Zymoetz	2707	12780	720	1648	6987	5686	7657	5012	3776	1160	10162	25350	11826	14003	17472	14467	16030	8693
Kalum	1365	1079	1279	2830	2880	5240	3585	5231	8753	7827	21450	19931	29076	28468	12016	20330	22405	21692
Lakelse	41342	30147	6513	71857	11262	6001	23221	13736	4404	14265	9781	26502	3895	36641	50551	5048	8994	6398
Alastair	13107	20114	7670	14587	28820	12198	26298	25541	16698	26274	18317	26527	16882	22744	47815	34017	17134	2068
Johnston	1383	2208	965	3588	885	1367	3135	1081	1498	1300	3750	13068	11868	16359	21806	11972	20504	13312
Total	1447330	1114504	1311573	2479036	963709	1576062	1637239	1362144	1216889	1531000	1581359	2100087	1334378	2236903	2651206	1394273	715687	838601
Ytee Est.	1447331	1114505	1311576	2479034	963708	1576062	1637237	1362144	1216885	1531001	1581359	2100091	1334376	2236902	2651205	1394274	715687	838599

Table A3. Catch calculated as the difference between Ytee run size and total escapement for each CU.

Stock	1982	1983	1984	1985	1986	1987	1988	1989	1990	1991	1992	1993	1994	1995	1996	1997	1998	1999
Kluatan+	181	271	115	116	201	95	173	228	155	90	94	88	119	127	125	174	299	135
Motase	68	56	81	37	85	137	7	42	8	27	44	24	25	26	11	33	101	66
Sustut+	568	605	553	308	679	344	350	529	514	332	375	255	1818	411	1062	1243	1780	1485
Bear+	158	163	144	236	592	980	704	245	316	698	530	384	890	1259	828	677	375	1231
Slamgeesh	142	151	138	77	170	86	88	132	129	83	94	63	97	103	112	162	212	127
Sicintine	142	151	138	77	170	86	88	132	129	83	94	63	97	103	112	162	212	127
Babine-WE	20902	7337	3463	8307	6209	3998	7689	4350	3557	6217	9470	4551	4309	14696	11628	24945	13403	11491
Babine-WH	944	1657	1302	1515	844	1660	2690	1260	1171	2629	1555	6027	1200	1205	2066	11465	4011	4135
Babine-WL	27905	10264	43473	53674	35336	27279	16429	13608	32128	51213	80858	76028	19346	11589	23575	29756	22396	28542
Babine-P	44132	35399	57320	51178	45989	68887	42467	37630	31728	45193	37180	116068	31895	98834	205478	95585	23397	24140
Babine-F	105634	70708	79593	65375	54896	35231	56491	86542	81602	20918	52599	96802	100233	186989	427044	181435	51542	37300
Swan+	2598	3695	1984	2360	2394	2129	3670	2853	1975	2027	2729	3529	3003	1688	1850	2482	4506	2605
Bulkley+	455	548	442	456	449	335	653	290	537	250	361	353	342	444	295	482	509	118
Moric+	5317	8805	5498	4015	5086	4612	4350	2368	7743	15278	17196	13381	16027	29257	18756	21984	5547	2205
Kitwanga	157	167	262	178	332	168	93	99	74	45	54	46	74	39	21	30	52	22
Zymoetz	99	412	10	24	216	128	166	170	91	14	254	830	512	392	358	567	1298	303
Kalum	44	26	25	42	87	112	62	131	238	95	525	613	975	665	266	807	1576	591
Lakelse	1411	843	86	932	324	126	493	432	95	152	236	854	156	988	880	177	639	200

Table A1. Total sockeye escapement results from Run Reconstruction analysis (output file="Escape.xls").

Stock	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
Kluatan+	1473	1404	1053	1034	1238	1174	1145	1130	1137	967	767	1019	808	819	1167	1208	1325	1155
Motase	375	191	96	1987	566	278	112	311	106	438	536	867	510	492	515	489	314	1114
Sustut+	4046	4073	3999	4525	1611	937	637	2536	242	563	306	1916	1261	1285	1054	1282	1237	2770
Bear+	5827	5265	1046	12155	3446	1671	1901	2478	9169	7797	10569	8249	15810	6276	23513	5319	6911	16884
Slamgeesh	958	869	398	401	294	191	292	374	156	163	648	696	915	89	398	1175	263	490
Sicintine	1009	1016	1000	933	1003	883	881	1023	1040	1015	876	1013	974	959	994	917	985	980
Babine-WE	104084	246406	30714	53028	44568	24180	40063	53771	26919	17553	14826	59879	43810	6998	22462	54272	42517	30006
Babine-WN	28571	69525	24836	77951	52100	27522	11833	12780	14857	18516	5245	56916	110788	7601	25580	90305	43362	20708
Babine-WL	156810	516039	124764	108199	223067	182032	118078	103023	93269	100679	65832	134007	111748	37855	164593	161430	148021	90877
Babine-P	198801	137663	96174	203149	103129	114057	278168	229170	186740	144277	116906	236403	123997	28915	241569	140140	76602	128446
Babine-F	785474	326419	272535	653979	415868	293711	654636	599057	517029	366228	328664	357228	682922	227175	999826	641760	421540	501921
Swan+	53311	59883	28867	100293	34604	17828	55490	24500	26934	40180	10705	26452	44324	17002	14922	82682	62571	34846
Bulkley+	1321	1263	1063	1042	1298	1322	1026	1135	1014	902	771	1054	813	803	1183	1179	1350	1164
Morice+	7278	7697	9289	12986	12443	12965	12305	18512	13384	13890	3668	5850	6642	6728	20521	31700	24022	21904
Kitwanga	277	257	1168	4268	1454	1290	6363	306	1789	4115	21610	2344	5594	789	19153	6209	1260	474
Zymoetz	20011	18099	9938	10503	9907	9738	4180	363	3461	2596	5016	2321	2431	5713	13735	15851	19275	
Kalum	29240	14968	19059	45692	25025	15911	16249	21143	28032	32171	45691	30815	45724	36699	43411	41757	25624	37732
Lakelse	20841	21701	10001	10543	11206	13211	6119	8907	7539	7359	11928	41063	17580	21060	35145	8295	35390	16698
Alastair	14069	19832	5277	35061	34375	23783	7496	30862	1508	24762	32695	27003	30000	17323	8115	28295	47556	7547
Johnston	8164	19081	5156	13453	4754	5307	4261	3306	3381	2446	1888	2927	1558	1862	3523	4304	5480	3337
Total	1441940	1471652	646432	1351596	982551	748162	1226789	1118505	934606	787486	676728	1000715	1248100	423161	1633358	1316453	962182	938329
Input	1938207	2104020	678091	1407540	1024423	785163	1495446	1154240	1163806	807346	798379	1251057	1567039	442883	2198626	1702220	1084377	952443
Difference	-496267	-632368	-31659	-55944	-41872	-37001	-268657	-35735	-229200	-19864	-121651	-250342	-318939	-19722	-565268	-385767	-122195	-14114

Table A2. Incoming abundance to Tye from Run Reconstruction analysis (output file="Stock_CalcEntry.xls").

Stock	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
Kluatan+	1831	1477	1258	1145	1480	1529	1326	1215	1287	1096	980	1175	902	930	1261	1339	1564	1190
Motase	472	207	116	2174	643	323	128	340	123	488	656	981	562	533	550	530	353	1168
Sustut+	6625	4842	5755	7107	2510	2146	1472	3199	885	1740	1056	2888	1971	2066	1284	2205	2047	3722
Bear+	7575	5611	1251	13332	3971	1993	2205	2683	10562	8726	13154	9381	17630	6873	25381	5807	7870	17631
Slamgeesh	1245	926	476	440	339	228	338	405	180	183	807	791	1021	98	430	1283	300	512
Sicintine	1312	1083	1196	1023	1156	1053	1022	1108	1199	1136	1090	1152	1086	1050	1073	1001	1122	1024
Babine-WE	139417	318284	38705	60945	55846	33062	51714	61385	36072	22584	21008	74993	55270	8109	26217	62955	57979	31413
Babine-WN	39429	92258	31211	88326	62807	35015	16869	14815	21017	22901	7108	70036	135966	8489	29468	102652	54955	21863
Babine-WL	195847	643538	157296	121300	257767	216270	172844	122623	122871	119971	84507	159142	126626	41200	178269	179016	173069	96063
Babine-P	365183	225428	120861	230187	124324	145111	384002	265648	317061	178443	158416	290898	283855	32292	312901	160537	97792	137315
Babine-F	1464721	839509	342301	735348	488882	358237	1041617	707647	803580	442692	431172	656378	932774	249546	1593059	1044259	599723	531783
Swan+	61269	61924	33093	106577	38347	20018	62207	25911	29539	43931	12644	29286	48678	18209	15906	88526	69935	35487
Bulkley+	1831	1477	1258	1145	1480	1529	1326	1215	1287	1096	980	1175	902	930	1261	1339	1564	1190
Morice+	10084	8999	10991	14278	14191	15000	15914	19816	16993	16886	4666	6525	7369	7790	21872	36010	27830	22388
Kitwanga	300	267	1325	4506	1571	1397	6837	325	1927	4438	24297	2533	5919	825	19895	6547	1364	485
Zymoetz	20774	18396	10617	11304	11144	10610	10136	4292	376	3667	2805	5259	2386	2499	5844	14247	16770	19309
Kalum	30327	15204	20500	47301	26359	16870	16824	21833	29099	33978	48987	32180	46961	37625	44317	43213	26905	38208
Lakelse	21651	21981	10611	10895	11872	14123	6378	9139	7795	7784	12812	42908	18057	21601	35901	8583	37273	16723
Alastair	14559	20044	5556	35879	35660	24981	7723	31467	1549	25758	34458	27822	30588	17669	8234	28977	49138	7579
Johnston	8268	19142	5226	13549	4812	5397	4306	3324	3407	2476	1920	2955	1567	1874	3539	4337	5539	3338
Total	2392720	2300597	799603	1506761	1145161	904892	1805188	1298390	1406809	939974	863523	1418458	1720090	460208	2326662	1793363	1233092	988391
Tye Est.	2392717	2300597	799602	1506763	1145163	904893	1805191	1298389	1406809	939972	863525	1418459	1720090	460208	2326662	1793365	1233087	923328

Table A3. Catch calculated as the difference between Tye run size and total escapement for each CU.

Stock	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
Kluatan+	358	73	205	111	242	355	181	85	150	129	213	156	94	111	94	131	239	35
Motase	97	16	20	187	77	45	16	29	17	50	120	114	52	41	35	41	39	54
Sustut+	2579	769	1756	2582	899	1209	835	663	643	1177	750	972	710	781	230	923	810	952
Bear+	1748	346	205	1177	525	322	304	205	1393	929	2585	1132	1820	597	1868	488	959	747
Slamgeesh	287	57	78	39	45	37	46	31	24	20	159	95	106	9	32	108	37	22
Sicintine	303	67	196	90	153	170	141	85	159	121	214	139	112	91	79	84	137	44
Babine-WE	35333	71878	7991	7917	11278	8882	11651	7614	9153	5031	6182	15114	11460	1111	3755	8683	15462	1407
Babine-WN	10858	22733	6375	10375	10707	7493	5036	2035	6160	4385	1863	13120	25178	888	3888	12347	11593	1155
Babine-WL	39037	127499	32532	13101	34700	34238	54766	19600	29602	19292	18675	25135	14878	3345	13676	17586	25048	5186
Babine-P	166382	87765	24687	27038	21195	31054	105834	36478	130321	34166	41510	54495	159858	3377	71332	20397	21190	8869
Babine-F	679247	513090	69766	81369	73014	64526	386981	108590	286551	76644	102508	299150	249852	22371	593233	402499	178183	29862
Swan+	7958	2041	4226	6284	3743	2190	6717	1411	2605	3751	1939	2834	4354	1207	984	5844	7364	641
Bulkley+	510	214	195	103	182	207	300	80	273	194	209	121	89	127	78	160	214	26
Morice+	2806	1302	1702	1292	1748	2035	3609	1304	3609	2996	998	675	727	1062	1351	4310	3808	484
Kitwanga	23	10	157	238	117	107	474	19	138	323	2687	189	325	36	742	338	104	11
Zymoetz	763	297	679	387	641	703	398	112	13	206	209	243	65	68	131	512	919	34
Kalum	1087	236	1441	1609	1334	959	575	690	1067	1807	3296	1365	1237	926	906	1456	1281	476
Lakelse	810	280	610	352	666	912	259	232	256	425	884	1845	477	541				

APPENDIX B

Skeena Sockeye In-River Model Visual Basic Code

' This program implements the Skeena River Sockeye Run Reconstruction
' by Karl English, Bill Gazey and Cam Noble Version 3.1 - March 2015.

' The model reconstructs the 1982-specified sockeye salmon returns to the Skeena River using First Nation provided estimates for in-river harvests,

' Tyee specific escapement numbers and timing provided by Steve Cox Rogers and Karl English, and upstream migration rates.
' The model combines catch data for 11 fisheries with escapement data on a daily time step. 20 sockeye stocks are reconstructed,
' with terminal runs, harvest rates, and catch estimated for each.

' The reconstruction algorithm is based on Cave and Gazey 1994.

' The model currently has the following data limits:

' 12 fisheries
' 20 stocks
' 1982 = Year Start
' 2017 = Year Finish

' Version 3.1
' March 2015

Option Base 1 'Start indexing arrays at 1

'Input data workbook declarations.

Public Fisheries_WkBk As Workbook 'Contains weekly FSC sockeye catch by fishery, week and year (each year a tab) and
the Tyee aggregate run by day and year
Public FisheriesESSR_WkBk As Workbook 'Contains weekly ESSR sockeye catch by fishery, week and year (each year a tab)
Public Entry_WkBk As Workbook 'Annual entry (escapement from marine fisheries) estimates for each stock.
Public Timing_WkBk As Workbook 'The timing workbook:
' "FISHRES" has the names and residence times for each stock in the fisheries
' "TIMING" has the offsets and standard deviations (days) for each stock from the aggregate at Tyee
' "Timing_Mean" has the mean date of arrival for stock aggregate at Tyee for each year
' Note that 2010 and 2011 do not have estimates yet

'Output workbook declarations

Public Reconstruct_WkBk As Workbook 'Contains catch reconstruction for each year.
Public Catch_WkBk As Workbook 'Annual terminal catch summary for each year.
Public Escape_WkBk As Workbook 'Annual terminal escapement (spawning ground) size by stock.
Public HarvestRate_WkWB As Workbook ' Contains 2 worksheets:
' "StockGroup" is the annual exploitation rate by stock group.
' "Stock" is the annual exploitation rate by stock.
Public Stock_Calc_Entry_WkBk As Workbook ' Annual entry at Tyee as estimated by run recon (CNoble Oct. 11)
Public FisheryHr_WkBk As Workbook 'Daily fishery harvest rates estimated each week
Public WeeklyHr_WkBk As Workbook 'Mean daily fishery harvest rate.
Public Unexpl_WkBk As Workbook 'Weekly unexplained catch by fishery and year

' Constant Declarations

Public Const BaseYear As Integer = 1982 'First year of data
Public Const DaysInModel As Integer = 140 'Number model days -- starts June 1
Public Const NumFisheries As Integer = 12 'Number of fisheries
Public Const NumSGroups As Integer = 2 'Number of stock groups -- Babine and non-Babine
Public Const NumStocks As Integer = 20 'Number of fish stocks (CUs)
Public Const WeeksInModel As Integer = 21 'Number of weeks in model
Public Const NumSectors As Integer = 2 'Number of fishing sectors

'Input Workbooks data offsets

Public Const Timing_Stock_DataRow = 3
Public Const Timing_Stock_DataCol = 2

Public Const Fisheries_DataCol = 4 'Column for first fishery in Fisheries_WkBk (FSC) work sheet (CNoble Oct 11)

Public Const Fisheries_DataRow = 2 'First row of weekly catch data in Fisheries_WkBk (FSC), catch-year worksheet (CNoble Oct 11)

'Output Workbooks data offsets

Public Const Reconst_Datarow As Integer = 2 'First row to begin output of reconstruction data (CNoble Oct 11)
 Public Const Reconst_CatchRow As Integer = NumFisheries + 3 'Row to output total catch of reconstruction data
 Public Const Year_Datacol As Integer = 2 'First column to begin output of yearly related sheets
 Public Const WkRun_Datarow As Integer = 3 'First row to begin output of daily run data (CNoble Oct 11)
 Public Const FishHR_Datarow As Integer = 3 'First row to begin output of daily fishery harvest rate
 Public Const FishHR_FishCol As Integer = 2 'Column number for first fishery in daily fishery harvest rate workbook

'Shared Variables

Public StartYear As Integer 'Start year for this model run
 Public EndYear As Integer 'End year for this model run

'Shared data structures

Public StockToStockGroup(NumStocks) As Integer 'contains the mapping for each stock to its stock group
 Public Residence(NumStocks, NumFisheries) As Integer 'contains the fishery residence times for each stock
 Public Cum_ResTimes(NumStocks, NumFisheries) As Integer 'contains the cumulative fishery residence times from tributary
 ' escapement to each fishery for each stock
 Public Spawn_Peak(NumStocks) As Double 'contains the peak spawning model day (includes spawning in tributary)
 for each stock
 Public Spawn_SD(NumStocks) As Double 'contains the run sd for each stock
 Public Spawn_Entry(NumStocks) As Double 'contains entry size for each stock
 Public Stock_Calc_Entry(NumStocks) As Double 'contains calculated entry size for each stock (CNoble Oct 11)
 Public Stock_Daily_Escape(NumStocks, DaysInModel) As Double 'total daily escapement for each stock
 Public StockGroupYearCatch(NumSGroups) As Double 'contains the total catch for each stock group in a given year
 Public StockYearCatch(NumStocks) As Double 'contains the total catch for each stock in a given year
 Public StockGroupTrun(NumSGroups) As Double
 Public StockTrun(NumStocks) As Double
 Public RunTotal As Double
 Public CatchBySector(NumStocks, NumSectors) As Double 'contains annual catch by sector
 Public FisheryNames(NumFisheries) As String 'contains fishery names
 Public WeeklyHR(NumFisheries, WeeksInModel) As Double 'contains weekly harvest rates by fishery (computed)
 Public FisheryHR(NumFisheries, WeeksInModel) As Double 'contains daily harvest rates by fishery and week

 Public FSC_Catch(NumFisheries, WeeksInModel) As Double
 Public ESSR_Catch(NumFisheries, WeeksInModel) As Double
 Public Total_Catch(NumFisheries) As Double 'KKE added 14 Oct 2012 for entry run size and terminal fishery
 calculations
 Public TerminalHR(NumFisheries) As Double
 Public Week_CatchDay(WeeksInModel) As Integer

Sub Read_Catch(Yr, Year_String)

Const Fisheries_row As Integer = 2
 Const FisheriesDate_col As Integer = 1
 Const FisheriesFirst_col As Integer = 4

Dim iweek As Integer, ifish As Integer
 Dim curdate As Date, bdate As Date
 Dim shtname As String
 Dim wsheet As Worksheet

Erase Total_Catch

With Fisheries_WkBk.Worksheets(Year_String)

For iweek = 1 To WeeksInModel

curdate = .Cells(iweek - 1 + Fisheries_row, FisheriesDate_col)

bdate = DateValue("June 1," & Format(Year(curdate)))

Week_CatchDay(iweek) = DateDiff("d", bdate, curdate) - 5 'set to the start of the week

For ifish = 1 To NumFisheries

FSC_Catch(ifish, iweek) = .Cells(iweek - 1 + Fisheries_row, FisheriesFirst_col + ifish - 1).Value

```

        Total_Catch(ifish) = Total_Catch(ifish) + FSC_Catch(ifish, iweek)
    Next ifish
    Next iweek
End With

shtname = Year_String + "ESSR"
On Error Resume Next
Set wsheet = FisheriesESSR_WkBk.Worksheets(shtname)
If wsheet Is Nothing Then
    Erase ESSR_Catch
Else
    With FisheriesESSR_WkBk.Worksheets(shtname)
        For iweek = 1 To WeeksInModel
            For ifish = 1 To NumFisheries
                ESSR_Catch(ifish, iweek) = .Cells(iweek - 1 + Fisheries_row, FisheriesFirst_col + ifish - 1).Value
                Total_Catch(ifish) = Total_Catch(ifish) + ESSR_Catch(ifish, iweek)
            Next ifish
        Next iweek
    End With
End If
On Error GoTo 0

End Sub

Sub Reconstruction()

Dim Year As Integer, YearCol As Integer, xYear As Integer
Dim Year_String As String 'Used to read the correct First Nation catch worksheet for the given year.

'Turn off display
Application.ScreenUpdating = False

xYear = 0
Call Init(xYear)

'initialize output rows
YearCol = Year_Datacol 'Start column to output data in year related worksheets

'do the run reconstruction for each year requested
For Year = StartYear To EndYear
    'Create the string for this year
    Year_String = Right(Str(Year), 4)
    'Read in the catch
    Call Read_Catch(Year, Year_String)
    'Pick up the duration and timing
    Call FishResSpawn(Year)
    'Calculate the daily escapement for each stock group
    Call Calc_Escape(Year)
    'Do the reconstruction
    Call Reconstruct(Year, Year_String)
    'Output the yearly results
    Call OutputData(Year, Year_String, YearCol)
    'Increment the output row pointers
    YearCol = YearCol + 1
Next Year

'Close the input workbooks
Fisheries_WkBk.Close (False)
FisheriesESSR_WkBk.Close (False)
Entry_WkBk.Close (False)
Timing_WkBk.Close (False)

```

```

'Save and close the output workbooks
Reconstruct_WkBk.Save
Reconstruct_WkBk.Close
Catch_WkBk.Save
Catch_WkBk.Close
Escape_WkBk.Save
Escape_WkBk.Close
Stock_Calc_Entry_WkBk.Save      'CNoble (Oct 11)
Stock_Calc_Entry_WkBk.Close    'CNoble (Oct 11)
HarvestRate_Wkwb.Save
HarvestRate_Wkwb.Close
FisheryHr_WkBk.Save
FisheryHr_WkBk.Close
WeeklyHr_WkBk.Save
WeeklyHr_WkBk.Close
Unexpl_WkBk.Save
Unexpl_WkBk.Close

'Quit Excel
'Application.Quit 'Comment out while debugging

End Sub
Sub Init(xYear)
'Reads in number of fishing days per week, and start/end years to run the model from the Control sheet
'Opens all data workbooks
'Prompts user for output workbook name, and creates output workbook

Dim Col_Num As Integer, StockGroup As Integer, Sheet As Integer, SheetCount As Integer, Year As Integer 'counters
Dim StockNames(NumStocks) As String 'Stock names
Dim StockGroupNames(NumSGroups) As String 'Stock group names
Dim FileNames(4) As String 'file names
Dim drivename As String, filepath As String, testfile As String

'Initialize the stock group names
StockGroupNames(1) = "Non Babine"
StockGroupNames(2) = "Babine"

'Prompt for the number of fishing days per week, start and end year
Worksheets("Control").Activate
With Range("Control")
    StartYear = .Cells(2, 1)
    EndYear = .Cells(2, 2)
End With

'Pick up file names
With Range("Files")
    For i = 1 To 4
        FileNames(i) = .Cells(i, 1)
    Next i
End With

'Get the path to this workbook, extract the drive, and explicitly change to this workbook's drive and folder.
'This sets the default folder for the input and output data workbooks to that of the workbook containing this program
filepath = ThisWorkbook.Path
drivename = Left(filepath, 1) 'Extract the drive
ChDrive (drivename)
ChDir (filepath)

Set Fisheries_WkBk = Workbooks.Open(Filename:=FileNames(1), UpdateLinks:=0)
Set FisheriesESSR_WkBk = Workbooks.Open(Filename:=FileNames(2), UpdateLinks:=0)
Set Entry_WkBk = Workbooks.Open(Filename:=FileNames(4), UpdateLinks:=0)
Set Timing_WkBk = Workbooks.Open(Filename:=FileNames(3), UpdateLinks:=0)

'Read in the stock and fishery names from the Timing workbook

```

```

For Stock = 1 To NumStocks
  StockNames(Stock) = Timing_WkBk.Worksheets("FISHRES").Cells(Timing_Stock_DataRow + Stock - 1,
Timing_Stock_DataCol).Value
Next Stock
For Fishery = 1 To NumFisheries
  FisheryNames(Fishery) = Timing_WkBk.Worksheets("FISHRES").Cells(Timing_Stock_DataRow - 1, Timing_Stock_DataCol +
Fishery + 4)
Next Fishery

'Yearly run reconstruction workbook
testfile = Dir("Reconstruction.xls")
If Len(testfile) > 0 Then Kill ("Reconstruction.xls")
Set Reconstruct_WkBk = Workbooks.Add
Reconstruct_WkBk.SaveAs Filename:="Reconstruction.xls", FileFormat:=xlAddIn
With Reconstruct_WkBk
  Sheet = 1
  SheetCount = .Worksheets.Count
  For Year = StartYear To EndYear
    'Check if need to add a worksheet to hold this year's reconstruction output
    If Sheet > SheetCount Then
      .Worksheets.Add after:=Worksheets(Sheet - 1)
    End If
    .Worksheets(Sheet).Name = Right(Str(Year), 4) 'str function adds leading blank - remove by extract 4 rightmost characters
    'Initialize the column headings
    Col_Num = 1
    With .Worksheets(Sheet)
      .Cells(1, Col_Num).Value = "Fishery"
      Col_Num = Col_Num + 1
      For Stock = 1 To NumStocks
        .Cells(1, Col_Num).Value = StockNames(Stock) 'stock names start in 2nd column
        Col_Num = Col_Num + 1
      Next Stock
      .Cells(1, Col_Num).Value = "Total"
      'Fit the column width to the headings
      .Range(.Cells(1, 1), .Cells(1, Col_Num)).ColumnWidth = 13
      .Range(.Cells(1, 1), .Cells(1, Col_Num)).HorizontalAlignment = xlRight
    End With
    Sheet = Sheet + 1
  Next Year
End With

'catch summary workbook
testfile = Dir("Catch.xls")
If Len(testfile) > 0 Then Kill ("Catch.xls")
Set Catch_WkBk = Workbooks.Add
Catch_WkBk.SaveAs Filename:="Catch.xls", FileFormat:=xlAddIn 'Later => prompt for output filename
Catch_WkBk.Worksheets(1).Name = "Catch"
With Catch_WkBk.Worksheets("Catch")
  .Cells(1, 1).Value = "Stock Group"
  Row_Num = 3
  For StockGroup = 1 To NumSGroups
    .Cells(Row_Num, 1).Value = StockGroupNames(StockGroup) 'stock names start in 2nd row
    Row_Num = Row_Num + 1
  Next StockGroup
  .Cells(Row_Num, 1).Value = "Total"
  'Fit the column width to the headings
  .Columns(1).AutoFit
End With

'Initialize catch by sector worksheets
Sheets.Add
ActiveSheet.Name = "FSCCatchByGroup"
With Catch_WkBk.Worksheets("FSCCatchByGroup")
  .Cells(1, 1).Value = "Stock Group"

```

```

Row_Num = 3
For StockGroup = 1 To NumSGroups
    .Cells(Row_Num, 1).Value = StockGroupNames(StockGroup) 'stock names start in 2nd row
    Row_Num = Row_Num + 1
Next StockGroup
.Cells(Row_Num, 1).Value = "Total"
'Fit the column width to the headings
.Columns(1).AutoFit
End With

```

```

Sheets.Add
ActiveSheet.Name = "ESSRCatchByGroup"
With Catch_WkBk.Worksheets("ESSRCatchByGroup")
    .Cells(1, 1).Value = "Stock Group"
    Row_Num = 3
    For StockGroup = 1 To NumSGroups
        .Cells(Row_Num, 1).Value = StockGroupNames(StockGroup) 'stock names start in 2nd row
        Row_Num = Row_Num + 1
    Next StockGroup
    .Cells(Row_Num, 1).Value = "Total"
    'Fit the column width to the headings
    .Columns(1).AutoFit
End With

```

```

Sheets.Add
ActiveSheet.Name = "FSCCatchByStock"
With Catch_WkBk.Worksheets("FSCCatchByStock")
    .Cells(1, 1).Value = "Stock"
    Row_Num = 3
    For Stock = 1 To NumStocks
        .Cells(Row_Num, 1).Value = StockNames(Stock) 'stock names start in 2nd row
        Row_Num = Row_Num + 1
    Next Stock
    .Cells(Row_Num, 1).Value = "Total"
    'Fit the column width to the headings
    .Columns(1).AutoFit
End With

```

```

Catch_WkBk.Worksheets.Add after:=Worksheets(4)
ActiveSheet.Name = "ESSRCatchByStock"
With Catch_WkBk.Worksheets("ESSRCatchByStock")
    .Cells(1, 1).Value = "Stock"
    Row_Num = 3
    For Stock = 1 To NumStocks
        .Cells(Row_Num, 1).Value = StockNames(Stock) 'stock names start in 2nd row
        Row_Num = Row_Num + 1
    Next Stock
    .Cells(Row_Num, 1).Value = "Total"
    'Fit the column width to the headings
    .Columns(1).AutoFit
End With

```

```

'terminal escapement size workbook
testfile = Dir("Escape.xls")
If Len(testfile) > 0 Then Kill ("Escape.xls")
Set Escape_WkBk = Workbooks.Add
Escape_WkBk.SaveAs Filename:="Escape.xls", FileFormat:=xlAddIn 'Later => prompt for output filename
'Initialize stock group escapement size worksheet
Escape_WkBk.Worksheets(1).Name = "StockGroup"
With Escape_WkBk.Worksheets("StockGroup")
    .Cells(1, 1).Value = "Stock Group"
    Row_Num = 3
    For StockGroup = 1 To NumSGroups
        .Cells(Row_Num, 1).Value = StockGroupNames(StockGroup) 'stock names start in 3'rd row
    
```

```

    Row_Num = Row_Num + 1
Next StockGroup
.Cells(Row_Num, 1).Value = "Total"
'Fit the column width to the headings
.Columns(1).AutoFit
End With

'Stock Calc Entry run size workbook (CNoble Oct 11)
testfile = Dir("Stock_Calc_Entry.xls")
If Len(testfile) > 0 Then Kill ("Stock_Calc_Entry.xls")
Set Stock_Calc_Entry_WkBk = Workbooks.Add
Stock_Calc_Entry_WkBk.SaveAs Filename:="Stock_Calc_Entry.xls", FileFormat:=xlAddIn 'Later => prompt for output filename
'Initialize stock group escapement size worksheet
Stock_Calc_Entry_WkBk.Worksheets(1).Name = "Stock"
With Stock_Calc_Entry_WkBk.Worksheets("Stock")
    .Cells(1, 1).Value = "Stock"
    Row_Num = 3
    For Stock = 1 To NumStocks
        .Cells(Row_Num, 1).Value = StockNames(Stock) 'stock names start in 3'rd row
        Row_Num = Row_Num + 1
    Next Stock
    .Cells(Row_Num, 1).Value = "Total"
    'Fit the column width to the headings
    .Columns(1).AutoFit
End With

'Initialize stock escapement size worksheet
Escape_WkBk.Activate
Sheets.Add
ActiveSheet.Name = "Stock"
With Escape_WkBk.Worksheets("Stock")
    .Cells(1, 1).Value = "Stock"
    Row_Num = 3
    For Stock = 1 To NumStocks
        .Cells(Row_Num, 1).Value = StockNames(Stock) 'stock names start in 3'rd column
        Row_Num = Row_Num + 1
    Next Stock
    .Cells(Row_Num, 1).Value = "Total"
    'Fit the column width to the headings
    .Columns(1).AutoFit
End With

'Exploitation rate workbook
testfile = Dir("HarvestRate.xls")
If Len(testfile) > 0 Then Kill ("HarvestRate.xls")
Set HarvestRate_WkWB = Workbooks.Add
HarvestRate_WkWB.SaveAs Filename:="HarvestRate.xls", FileFormat:=xlAddIn 'Later => prompt for output filename
'Initialize stock group harvest rate worksheet
HarvestRate_WkWB.Worksheets(1).Name = "StockGroup"
With HarvestRate_WkWB.Worksheets("StockGroup")
    .Cells(1, 1).Value = "Stock Group"
    Row_Num = 3
    For StockGroup = 1 To NumSGroups
        .Cells(Row_Num, 1).Value = StockGroupNames(StockGroup) 'stock group names start in 3rd row
        Row_Num = Row_Num + 1
    Next StockGroup
    .Cells(Row_Num, 1).Value = "Total"
    'Fit the column width to the headings
    .Columns(1).AutoFit
    'Format all cells as text to force display of decimal place when 0
    .Range(.Cells(3, 2), .Cells(NumSGroups + 3, EndYear - StartYear + Year_Datacol)).Columns.NumberFormat = "0.0"
End With

'Initialize stock exploitation rate worksheet

```

```

HarvestRate_WkWb.Activate
Sheets.Add
ActiveSheet.Name = "Stock"
With HarvestRate_WkWb.Worksheets("Stock")
    .Cells(1, 1).Value = "Stock"
    Row_Num = 3
    For Stock = 1 To NumStocks
        .Cells(Row_Num, 1).Value = StockNames(Stock) 'stock names start in 3rd row
        Row_Num = Row_Num + 1
    Next Stock
    .Cells(Row_Num, 1).Value = "Total"
    'Fit the column width to the headings
    .Columns(1).AutoFit
    'Format all cells as text to force display of decimal place when 0
    .Range(.Cells(3, 2), .Cells(NumStocks + 3, EndYear - StartYear + Year_Datacol)).Columns.NumberFormat = "0.0"
End With

'daily fishery harvest rate workbook
testfile = Dir("FisheryHr.xls")
If Len(testfile) > 0 Then Kill ("FisheryHr.xls")
Set FisheryHr_WkBk = Workbooks.Add
FisheryHr_WkBk.SaveAs Filename:="FisheryHR.xls", FileFormat:=xlAddIn 'Later => prompt for output filename
With FisheryHr_WkBk
    Sheet = 1
    SheetCount = .Worksheets.Count
    For Year = StartYear To EndYear
        'Check if need to add a worksheet to hold this year's reconstruction output
        If Sheet > SheetCount Then
            .Worksheets.Add after:=Worksheets(Sheet - 1)
        End If
        .Worksheets(Sheet).Name = Right(Str(Year), 4) 'str function adds leading blank - remove by extract 4 rightmost characters
        'Initialize the column headings
        With .Worksheets(Sheet)
            .Cells(1, 1).Value = "Week"
            For iweek = 1 To WeeksInModel
                .Cells(FishHR_Datarow + iweek - 1, 1).Value = iweek 'week numbers are in first column
            Next iweek
            For Fishery = 1 To NumFisheries
                .Cells(1, FishHR_FishCol + Fishery - 1).Value = FisheryNames(Fishery) 'fishery names start in 2nd column
            Next Fishery
            .Range(.Cells(1, 1), .Cells(1, NumFisheries + 1)).ColumnWidth = 6
            .Range(.Cells(1, 1), .Cells(1, NumFisheries + 1)).Orientation = 90
        End With
        Sheet = Sheet + 1
    Next Year
End With

'weekly harvest rate workbook
testfile = Dir("WeeklyHr.xls")
If Len(testfile) > 0 Then Kill ("WeeklyHr.xls")
Set WeeklyHr_WkBk = Workbooks.Add
WeeklyHr_WkBk.SaveAs Filename:="WeeklyHR.xls", FileFormat:=xlAddIn 'Later => prompt for output filename
With WeeklyHr_WkBk
    Sheet = 1
    SheetCount = .Worksheets.Count
    For Year = StartYear To EndYear
        'Check if need to add a worksheet to hold this year's reconstruction output
        If Sheet > SheetCount Then
            .Worksheets.Add after:=Worksheets(Sheet - 1)
        End If
        .Worksheets(Sheet).Name = Right(Str(Year), 4) 'str function adds leading blank - remove by extract 4 rightmost characters
        'Initialize the column headings
        With .Worksheets(Sheet)
            .Cells(1, 1).Value = "Week"

```



```

For iweek = 1 To WeeksInModel
    .Cells(FishHR_Datarow + iweek - 1, 1).Value = iweek 'week numbers are in first column
Next iweek
For Fishery = 1 To NumFisheries
    .Cells(1, FishHR_FishCol + Fishery - 1).Value = FisheryNames(Fishery) 'fishery names start in 2nd column
Next Fishery
.Range(.Cells(1, 1), .Cells(1, NumFisheries + 1)).ColumnWidth = 6
.Range(.Cells(1, 1), .Cells(1, NumFisheries + 1)).Orientation = 90
End With
Sheet = Sheet + 1
Next Year
End With

'weekly unexplained catch
testfile = Dir("Unexplained.xls")
If Len(testfile) > 0 Then Kill ("Unexplained.xls")
Set Unexpl_WkBk = Workbooks.Add
Unexpl_WkBk.SaveAs Filename:="Unexplained.xls", FileFormat:=xlAddIn 'Later => prompt for output filename
With Unexpl_WkBk
    Sheet = 1
    SheetCount = .Worksheets.Count
    For Year = StartYear To EndYear
        'Check if need to add a worksheet to hold this year's reconstruction output
        If Sheet > SheetCount Then
            .Worksheets.Add after:=Worksheets(Sheet - 1)
        End If
        .Worksheets(Sheet).Name = Right(Str(Year), 4) 'str function adds leading blank - remove by extract 4 rightmost characters
        'Initialize the column headings
        With .Worksheets(Sheet)
            .Cells(1, 1).Value = "Week"
            For Week = 1 To WeeksInModel
                .Cells(FishHR_Datarow + Week - 1, 1).Value = Week 'week numbers are in first column
            Next Week
            .Cells(FishHR_Datarow + WeeksInModel, 1).Value = "Total"
            For Fishery = 1 To NumFisheries
                .Cells(1, FishHR_FishCol + Fishery - 1).Value = FisheryNames(Fishery) 'fishery names start in 2nd column
            Next Fishery
            .Range(.Cells(1, 1), .Cells(1, NumFisheries + 1)).ColumnWidth = 6
            .Range(.Cells(1, 1), .Cells(1, NumFisheries + 1)).Orientation = 90
        End With
        Sheet = Sheet + 1
    Next Year
End With

End Sub

Sub FishResSpawn(Yr)
'Reads in the fishery residence times for each stock and determines the cumulative time from tributary escapment to
'each fishery.
'Reads in Tyee peak, stock offset and stock SD for the year.
'Reads in the entry size by stock for the year

'data offsets in fish residence times worksheet
Const fishres_datarow As Integer = 3
Const stockgrp_col As Integer = 4
Const fishery1_col As Integer = 7

'data offsets in fish spawning timing worksheet
Const Timing_off_col As Integer = 5 'Column number for offset
Const Timing_off_row As Integer = 5 'Row number for offset
Const Timing_med_col As Integer = 2 'Column number for Tyee median
Const Timing_med_row As Integer = 2 'row for Base Year

'data offsets in entry size

```

```
Const Entry_row As Integer = 6
Const Entry_col As Integer = 2
```

```
Dim cum, Fishery, fishres_row, ResTime, Stock, StockGroup As Integer 'local variables for residence data
Dim timing_row, trib_time As Integer 'local variables for spawning timing data
Dim Peak, date1, date2, x
```

```
'Read in the fishery residence times for each stock.
```

```
Sht_res = "FISHRES"
fishres_row = fishres_datarow 'Skip the header rows
With Timing_WkBk.Worksheets(Sht_res)
  For Stock = 1 To NumStocks
    StockToStockGroup(Stock) = .Cells(fishres_row, stockgrp_col).Value
    For Fishery = 1 To NumFisheries
      'Test for empty cell => causes crash
      If IsEmpty(.Cells(fishres_row, fishery1_col + Fishery - 1).Value) Or _
        (.Cells(fishres_row, fishery1_col + Fishery - 1).Value = "") Then
        ResTime = 0
      Else
        ResTime = .Cells(fishres_row, fishery1_col + Fishery - 1).Value
      End If
      Residence(Stock, Fishery) = ResTime
    Next Fishery
    fishres_row = fishres_row + 1
  Next Stock
End With
```

```
'Calculate the cumulative residence times from Tye entry for each stock and fishery
```

```
For Stock = 1 To NumStocks
  cum = 0
  For Fishery = 1 To NumFisheries
    cum = cum + Residence(Stock, Fishery)
    Cum_ResTimes(Stock, Fishery) = cum
  Next Fishery
Next Stock
```

```
'read in Tye peak
```

```
date1 = Timing_WkBk.Worksheets("TIMING_Mean").Cells(Timing_med_row + Yr - BaseYear, Timing_med_col)
date2 = DateValue("June 1, " & Format(Year(date1)))
Peak = DateDiff("d", date2, date1) + 1
```

```
'Read in the SD and offset.
```

```
Sht_res = "TIMING"
With Timing_WkBk.Worksheets(Sht_res)
  For Stock = 1 To NumStocks
    Spawn_Peak(Stock) = .Cells(Timing_off_row + Stock - 1, Timing_off_col)
    Spawn_Peak(Stock) = Spawn_Peak(Stock) + Peak
    Spawn_SD(Stock) = .Cells(Timing_off_row + Stock - 1, Timing_off_col + 2 + Yr - BaseYear)
  Next Stock
End With
```

```
'Read in the entry size by stock
```

```
Sht_res = "Skeena Escape"
With Entry_WkBk.Worksheets(Sht_res)
  For Stock = 1 To NumStocks
    x = .Cells(Entry_row + Yr - BaseYear, Entry_col + Stock - 1).Value
    If IsEmpty(x) Or x = "" Then
      Spawn_Entry(Stock) = 0
    Else
      Spawn_Entry(Stock) = x
    End If
  Next Stock
End With
```

```

'Add terminal catch and escapement to get entry stock size for Sustut and Bulkley sockeye (KKE - 22 Jan. 2013)
'Spawn Entry numbers for Babine stocks include catches at and above the Babine fence
  Spawn_Entry(3) = Spawn_Entry(3) + Total_Catch(12)    'Sustut
  Spawn_Entry(14) = Spawn_Entry(14) + Total_Catch(11)  'Bulkley

End Sub

Sub Calc_Escape(Yr)
'Calculates the daily escapement for each stock in the year specified

Dim curdate As Date, bdate As Date
Dim irow As Integer, iday As Integer, Stock As Integer, Sector As Integer, isg As Integer
Dim agg As Double, psum As Double
Dim p(NumStocks) As Double

Const Tyee_row As Integer = 5
Const Tyee_col As Integer = 1

irow = Tyee_row
Erase Stock_Calc_Entry          ' CNoble Oct 11
With Fisheries_WkBk.Worksheets("TyeeRunbyDay")
  curdate = .Cells(irow, Tyee_col)
  Do While curdate > 0
    bdate = DateValue("June 1," & Format(Year(curdate)))
    iday = DateDiff("d", bdate, curdate) + 1
    agg = .Cells(irow, Tyee_col + Yr - BaseYear + 1)
    psum = 0
    For Stock = 1 To NumStocks
      p(Stock) = Spawn_Entry(Stock) * WorksheetFunction.NormDist(iday, Spawn_Peak(Stock), Spawn_SD(Stock), False)
      psum = psum + p(Stock)
    Next Stock
    For Stock = 1 To NumStocks
      If psum > 0 Then
        Stock_Daily_Escape(Stock, iday) = p(Stock) * agg / psum
        Stock_Calc_Entry(Stock) = Stock_Calc_Entry(Stock) + Stock_Daily_Escape(Stock, iday)  ' CNoble Oct 11
      Else
        Stock_Daily_Escape(Stock, iday) = 0
      End If
    Next Stock
    irow = irow + 1
    curdate = .Cells(irow, Tyee_col)
  Loop
End With

'totals
Erase StockGroupTrun, StockTrun
RunTotal = 0
For Stock = 1 To NumStocks
  For iday = 1 To DaysInModel
    agg = Stock_Daily_Escape(Stock, iday)
    StockTrun(Stock) = StockTrun(Stock) + agg
    isg = StockToStockGroup(Stock)
    StockGroupTrun(isg) = StockGroupTrun(isg) + agg
  Next iday
  RunTotal = RunTotal + StockTrun(Stock)
Next Stock

Erase CatchBySector          ' CNoble Oct 11

'Carry over from old code -- required??
'For Stock = 1 To NumStocks
'Initialize catch by sector to zero for each stock each year
'For Sector = 1 To NumSectors

```

```

    ' CatchBySector(Stock, Sector) = 0
    'Next Sector
    'Next Stock

End Sub

Function gfs(HRate, T, Fshry, FDaysPerWk) As Double
'Returns catch for the given harvest rate (forward algorithm with entry timing)

Dim Day, i, k, ResidenceDays, Stock As Integer
Dim CaTest, x, surv(50) As Double

CaTest = 0
For Stock = 1 To NumStocks
    'Only do if the stock go through the fishery, i.e. residence time > 0
    ResidenceDays = Residence(Stock, Fshry)
    If ResidenceDays > 0 Then
        'Initialize survival to 100 percent
        For i = 1 To FDaysPerWk + ResidenceDays + 1
            surv(i) = 1
        Next i

        'Work forward through the fishing days
        For Day = 1 To FDaysPerWk
            For k = 1 To ResidenceDays
                'i is fishery chunks being harvested
                'j is escapement day
                i = Day + k - 2
                j = i + T - Cum_ResTimes(Stock, Fshry)

                'Pick up chunk
                If (j < 1 Or j > DaysInModel) Then
                    x = 0
                Else
                    x = Stock_Daily_Escape(Stock, j)
                End If

                'Calculate survival and catch
                If (HRate < 1) And (x > 0) Then
                    CaTest = CaTest + HRate * surv(i + 1) * x
                    surv(i + 1) = surv(i + 1) * (1 - HRate)
                End If
            Next k
        Next Day
    End If
Next Stock

gfs = CaTest
End Function

Function CalcHarvRate(Wk, Yr, Fishry, FDaysPerWeek, Catch, TeeDay) As Double

Const Tol As Double = 0.001
Dim j, k, Stock As Integer
Dim ESum, HarvRate, High, Low, test As Double

'Calculates the harvest rate for a fishery
HarvRate = 0

If only one day of fishing, sum up the escapement
If FDaysPerWeek = 1 Then
    ESum = 0
    For Stock = 1 To NumStocks
        For k = 1 To Residence(Stock, Fishry)

```

```

    j = TeeDay + k - 1 - Cum_ResTimes(Stock, Fishry)
    If (j < 1 Or j < DaysInModel) Then
        ESum = ESum + Stock_Daily_Escape(Stock, j)
    End If
Next k
Next Stock
If (ESum = 0) Then
    HarvRate = -0.99
Else
    HarvRate = Catch / (Catch + ESum)
End If
Else
    'Use the bisection method to estimate the harvest rate
    Low = 0
    High = 1
    HarvRate = 0.5 'Start with 50% harvest
    test = 0

    Do Until (Abs(Catch - test) / Catch * 100) < Tol
        test = gfs(HarvRate, TeeDay, Fishry, FDaysPerWeek)
        If test = 0 Then
            HarvRate = -0.99
            Exit Do
        End If
        If Catch > test Then
            Low = HarvRate
        Else
            High = HarvRate
        End If
        HarvRate = (Low + High) / 2
    Loop
End If

CalcHarvRate = HarvRate

End Function

Sub Reconstruct(Yr, Year_String)
'Does a forward reconstruction for the given year.

Dim Day As Integer, EscDay As Integer, Fishery As Integer, FishDays As Integer
Dim ResDay As Integer, StockGroup As Integer, Week As Integer
Dim interc(NumFisheries, NumStocks) As Double 'Total catch for each stock in each fishery
Dim HarvRate As Double, FisheryWkHR As Double, StockCatch As Double, PSport As Double
Dim WklyCat(WeeksInModel) As Double, WklyAA(WeeksInModel) As Double, WklyUnexpl(WeeksInModel) As Double

Dim iweek As Integer
Dim WeeklyCatch As Double, pFSC As Double, TerminalRun As Double

OutputRow = 2 'Initialize row to begin writing data in the run reconstruction workbook
Erase interc
Erase TerminalHR
savcat = 0: precat = 0
For Fishery = 1 To NumFisheries

    Erase WklyCat, WklyAA, WklyUnexpl
    FishDays = 7 'fishing 7 days a week

    'Calculate harvest rate for Bulkley and Sustut terminal fisheries (KKE - 29 Sep 2015)
    'Calculations for Pinkut and Fulton terminal fisheries was adjusted due to changes in ESSR catch timing
    'The following old code was replaced with the alternative code below
    'If Fishery = 11 Or Fishery = 12 Then
    '    If Fishery = 11 Then TerminalRun = Spawn_Entry(13) + Spawn_Entry(14)
    '    If Fishery = 12 Then TerminalRun = Spawn_Entry(3)

```

```

' TerminalHR(Fishery) = 0
' If TerminalRun > 0 Then TerminalHR(Fishery) = Total_Catch(Fishery) / TerminalRun
End If

'Alternative calculation for terminal fishery harvest rate (KKE - 29 Sep 2015)
'This code produced more accurate catch estimates after accounting for mainstem harvests
If Fishery = 11 Or Fishery = 12 Then
  TerminalRun = 0
  For EscDay = 1 To DaysInModel
    If Fishery = 11 Then TerminalRun = TerminalRun + Stock_Daily_Escape(13, EscDay) + Stock_Daily_Escape(14, EscDay)
    If Fishery = 12 Then TerminalRun = TerminalRun + Stock_Daily_Escape(3, EscDay)
  Next EscDay
  TerminalHR(Fishery) = 0
  If TerminalRun > 0 Then TerminalHR(Fishery) = Total_Catch(Fishery) / TerminalRun
End If

'Work forwards through each week of the fishery
Erase FisheryHR
For iweek = 1 To WeeksInModel
  WeeklyCatch = FSC_Catch(Fishery, iweek) + ESSR_Catch(Fishery, iweek)
  If WeeklyCatch > 0 Then
    pFSC = FSC_Catch(Fishery, iweek) / WeeklyCatch
    savcat = savcat + WeeklyCatch
    'Calculate the harvest rate for todays catch in this fishery
    Tday = Week_CatchDay(iweek)
    'Use terminal harvest rate for Bulkley-Morice and Sustut Fisheries (KKE 29 Sep 2015)
    If Fishery = 11 Or Fishery = 12 Then
      HarvRate = TerminalHR(Fishery)
    Else
      HarvRate = CalcHarvRate(iweek, Yr, Fishery, FishDays, WeeklyCatch, Tday)
    End If

    'Save Daily Harvest Rate - 15 March 2012
    FisheryHR(Fishery, iweek) = HarvRate

    'Update Run Size
    If HarvRate > 0 Then
      For Stock = 1 To NumStocks
        For ResDay = 1 To Residence(Stock, Fishery)
          For Day = 1 To FishDays
            EscDay = Tday - Cum_ResTimes(Stock, Fishery) + ResDay + Day - 2
            If (EscDay > 0 And EscDay <= DaysInModel) Then

              'Calculate the catch for this stock and update the yearly catch for it
              StockCatch = HarvRate * Stock_Daily_Escape(Stock, EscDay)
              StockYearCatch(Stock) = StockYearCatch(Stock) + StockCatch
              WklyCat(iweek) = WklyCat(iweek) + StockCatch

              'overwrite the stock daily escapement with the number of fish afet harvesting
              Stock_Daily_Escape(Stock, EscDay) = Stock_Daily_Escape(Stock, EscDay) * (1 - HarvRate)

              'Update the available abundance into this fishery for this week
              WklyAA(iweek) = WklyAA(iweek) + Stock_Daily_Escape(Stock, EscDay)

              'Sum catch by sector
              CatchBySector(Stock, 1) = CatchBySector(Stock, 1) + StockCatch * pFSC
              CatchBySector(Stock, 2) = CatchBySector(Stock, 2) + StockCatch * (1 - pFSC)

              'update the catch for this stock group and fishery
              StockGroup = StockToStockGroup(Stock)
              interc(Fishery, Stock) = interc(Fishery, Stock) + StockCatch
              precat = precat + StockCatch
            End If
          Next Day
        Next Stock
      End If
    End If
  End If
Next iweek

```

```

        Next ResDay
    Next Stock
    Else
        If HarvRate < 0 Then WklyUnexpl(iweek) = WklyUnexpl(iweek) + WeeklyCatch
    End If
    End If
Next iweek

'write to the daily fisheries harvest rate workbook
For iweek = 1 To WeeksInModel
    hr = FisheryHR(Fishery, iweek) * 100
    FisheryHr_WkBk.Worksheets(Year_String).Cells(FishHR_Datarow + iweek - 1, FishHR_FishCol + Fishery - 1).Value =
Format(hr, "##0.0") 'Round to 1 decimal place
Next iweek

'Calculate the weekly harvest rate for this fishery and write to the fisheries harvest rate workbook
UnexplSum = 0
For iweek = 1 To WeeksInModel
    aa = WklyAA(iweek)
    cc = WklyCat(iweek)
    If aa > 0 Then
        FisheryWkHR = cc / (cc + aa) * 100
    Else
        FisheryWkHR = 0 ' Do we want to print this or just leave blank?
    End If
    WeeklyHr_WkBk.Worksheets(Year_String).Cells(FishHR_Datarow + iweek - 1, FishHR_FishCol + Fishery - 1).Value =
Format(FisheryWkHR, "##0.0") 'Round to 1 decimal place
    If WklyUnexpl(iweek) > 0 Then Unexpl_WkBk.Worksheets(Year_String).Cells(FishHR_Datarow + iweek - 1,
FishHR_FishCol + Fishery - 1).Value = Round(WklyUnexpl(iweek))
    UnexplSum = UnexplSum + WklyUnexpl(iweek)
Next iweek
If UnexplSum > 0 Then Unexpl_WkBk.Worksheets(Year_String).Cells(FishHR_Datarow + WeeksInModel, FishHR_FishCol +
Fishery - 1).Value = Round(UnexplSum)

'Output the catch for each stock in this fishery to the run reconstruction workbook
With Reconstruct_WkBk.Worksheets(Year_String)
    .Cells(OutputRow, 1).Value = FisheryNames(Fishery)
    CatchTotal = 0
    For Stock = 1 To NumStocks
        StockCatch = interc(Fishery, Stock)
        CatchTotal = CatchTotal + StockCatch
        StockGroup = StockToStockGroup(Stock)
        StockGroupYearCatch(StockGroup) = StockGroupYearCatch(StockGroup) + StockCatch 'Sum the catch for all fisheries
for this stock group
        .Cells(OutputRow, Stock + 1).Value = Round(StockCatch) 'Round to nearest integer for output
    Next Stock
    .Cells(OutputRow, Stock + 1).Value = Round(CatchTotal) 'Output total catch for this fishery
End With
    OutputRow = OutputRow + 1
Next Fishery
End Sub

Sub OutputData(Yr, Year_String, YearCol)
Dim Day As Integer, Stock As Integer, StockGroup As Integer, StockCatch As Long
Dim StockEsc(NumStocks) As Double, StockGroupEsc(NumSGroups) As Double 'Total ending escapement for each stock and
stock group
Dim TotalSectorCatch(NumSectors) As Double 'Total Catch by Sector
Dim GroupCatchBySector(NumSGroups, NumSectors) As Double 'Total Catch by Group and Sector
Dim HR_Total As Double, Stock_HR As Double
Dim CatchTotal As Double, SCatchTotal As Double, TotalEscape As Double
Dim harvest_rate As String

Erase StockEsc, StockGroupEsc
TotalEscape = 0

```

```

For Stock = 1 To NumStocks
  StockGroup = StockToStockGroup(Stock)
  For Day = 1 To DaysInModel
    xxx = Stock_Daily_Escape(Stock, Day)
    StockEsc(Stock) = StockEsc(Stock) + xxx
    StockGroupEsc(StockGroup) = StockGroupEsc(StockGroup) + xxx
  Next Day
  TotalEscape = TotalEscape + StockEsc(Stock)
Next Stock

'Write the Stock Calc Entry data (CNoble Oct 11)
Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow, 1).Value = "Catch"
Stock_Calc_Entry_WkBk.Worksheets("Stock").Cells(1, YearCol).Value = Year_String
For Stock = 1 To NumStocks
  Stock_Calc_Entry(Stock) = Round(Stock_Calc_Entry(Stock)) 'Round to nearest integer
  Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow, StockGroup + 1).Value = StockCatch
  Stock_Calc_Entry_WkBk.Worksheets("Stock").Cells(Stock + 2, YearCol).Value = Stock_Calc_Entry(Stock)
Next Stock

'Write the total catch for each stock group to both the Run Reconstruction and Annual Terminal Catch workbooks
CatchTotal = 0
Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow, 1).Value = "Catch"
Catch_WkBk.Worksheets("Catch").Cells(1, YearCol).Value = Year_String
For StockGroup = 1 To NumSGroups
  StockCatch = Round(StockGroupYearCatch(StockGroup)) 'Round to nearest integer
  Catch_WkBk.Worksheets("Catch").Cells(StockGroup + 2, YearCol).Value = StockCatch
  CatchTotal = CatchTotal + StockCatch
Next StockGroup
For Stock = 1 To NumStocks
  Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow, Stock + 1).Value = Round(StockYearCatch(Stock))
Next Stock
'Write out the total catch across all stock groups
Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow, NumStocks + 2).Value = CatchTotal
Catch_WkBk.Worksheets("Catch").Cells(NumSGroups + 3, YearCol).Value = CatchTotal
SCatchTotal = CatchTotal

'Write the catch by sector for each stock group to the annual catch workbook
For Stock = 1 To NumStocks
  StockGroup = StockToStockGroup(Stock)
  For Sector = 1 To NumSectors
    GroupCatchBySector(StockGroup, Sector) = GroupCatchBySector(StockGroup, Sector) + CatchBySector(Stock, Sector)
  Next Sector
Next Stock
CatchTotal = 0
Catch_WkBk.Worksheets("FSSCCatchByGroup").Cells(1, YearCol).Value = Year_String
For StockGroup = 1 To NumSGroups
  StockCatch = Round(GroupCatchBySector(StockGroup, 1)) 'Round to nearest integer
  Catch_WkBk.Worksheets("FSSCCatchByGroup").Cells(StockGroup + 2, YearCol).Value = StockCatch
  CatchTotal = CatchTotal + StockCatch
Next StockGroup
'Write out the total catch across all stock groups
Catch_WkBk.Worksheets("FSSCCatchByGroup").Cells(NumSGroups + 3, YearCol).Value = CatchTotal

CatchTotal = 0
Catch_WkBk.Worksheets("ESSRCatchByGroup").Cells(1, YearCol).Value = Year_String
For StockGroup = 1 To NumSGroups
  StockCatch = Round(GroupCatchBySector(StockGroup, 2)) 'Round to nearest integer
  Catch_WkBk.Worksheets("ESSRCatchByGroup").Cells(StockGroup + 2, YearCol).Value = StockCatch
  CatchTotal = CatchTotal + StockCatch
Next StockGroup
'Write out the total catch across all stock groups
Catch_WkBk.Worksheets("ESSRCatchByGroup").Cells(NumSGroups + 3, YearCol).Value = CatchTotal

'Calculate the total run and harvest rates for each stock group and write to the Run Reconstruction,

```



```

'Annual Terminal Run and Harvest Rate workbooks
Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow + 1, 1).Value = "Run"
Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow + 2, 1).Value = "ER"
Reconstruct_WkBk.Worksheets(Year_String).Rows(Reconst_CatchRow + 2).NumberFormat = "0.0" 'Format cells to force display
of decimal place when 0
Reconstruct_WkBk.Worksheets(Year_String).Columns(1).AutoFit
HarvestRate_WkWB.Worksheets("StockGroup").Cells(1, YearCol).Value = Year_String
HarvestRate_WkWB.Worksheets("Stock").Cells(1, YearCol).Value = Year_String
Escape_WkBk.Worksheets("Stockgroup").Cells(1, YearCol).Value = Year_String
Escape_WkBk.Worksheets("Stock").Cells(1, YearCol).Value = Year_String
Catch_WkBk.Worksheets("FSCCatchByGroup").Cells(1, YearCol).Value = Year_String
Catch_WkBk.Worksheets("ESSRCatchByGroup").Cells(1, YearCol).Value = Year_String
Catch_WkBk.Worksheets("FSCCatchByStock").Cells(1, YearCol).Value = Year_String
Catch_WkBk.Worksheets("ESSRCatchByStock").Cells(1, YearCol).Value = Year_String

For Sector = 1 To NumSectors
    TotalSectorCatch(Sector) = 0
Next Sector
For Stock = 1 To NumStocks
    Catch_WkBk.Worksheets("FSCCatchByStock").Cells(Stock + 2, YearCol).Value = Round(CatchBySector(Stock, 1))
    Catch_WkBk.Worksheets("ESSRCatchByStock").Cells(Stock + 2, YearCol).Value = Round(CatchBySector(Stock, 2))
    For Sector = 1 To NumSectors
        TotalSectorCatch(Sector) = TotalSectorCatch(Sector) + Round(CatchBySector(Stock, Sector))
    Next Sector
Next Stock
Catch_WkBk.Worksheets("FSCCatchByStock").Cells(Stock + 2, YearCol).Value = TotalSectorCatch(1)
Catch_WkBk.Worksheets("ESSRCatchByStock").Cells(Stock + 2, YearCol).Value = TotalSectorCatch(2)

For StockGroup = 1 To NumSGroups
    'Calculate the harvest rate for this stock group
    If StockGroupTrun(StockGroup) > 0 Then
        Stock_HR = (StockGroupYearCatch(StockGroup) / StockGroupTrun(StockGroup)) * 100
    Else
        Stock_HR = 0
    End If
    StockGroupYearCatch(StockGroup) = 0 'Zero for the next year calculation
    harvest_rate = Format(Stock_HR, "###.0") 'Round to 1 decimal place
    'Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow + 2, StockGroup + 1).Value = harvest_rate
    HarvestRate_WkWB.Worksheets("StockGroup").Cells(StockGroup + 2, YearCol).Value = Stock_HR
    Escape_WkBk.Worksheets("StockGroup").Cells(StockGroup + 2, YearCol).Value = Round(StockGroupEsc(StockGroup))
Next StockGroup

For Stock = 1 To NumStocks
    Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow + 1, Stock + 1).Value = Round(StockTrun(Stock))
    If StockTrun(Stock) > 0 Then
        Stock_HR = StockYearCatch(Stock) / StockTrun(Stock) * 100
    Else
        Stock_HR = 0
    End If
    Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow + 2, Stock + 1).Value = Format(Stock_HR, "###.0")
Next Stock

'Calculate the harvest rate for each stock and write to the stock harvest rate worksheet
For Stock = 1 To NumStocks
    If StockTrun(Stock) > 0 Then
        Stock_HR = (StockYearCatch(Stock) / StockTrun(Stock)) * 100
    Else
        Stock_HR = 0
    End If
    StockYearCatch(Stock) = 0 'Zero for the next year calculation
    For Sector = 1 To NumSectors
        CatchBySector(Stock, Sector) = 0
    Next Sector
    HarvestRate_WkWB.Worksheets("Stock").Cells(Stock + 2, YearCol).Value = Stock_HR

```

```
Escape_WkBk.Worksheets("Stock").Cells(Stock + 2, YearCol).Value = Round(StockEsc(Stock))
Next Stock
```

'Write out the total run across all stock groups

```
Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow + 1, NumStocks + 2).Value = Round(RunTotal)
```

'Write out the total harvest rate across all stock groups

If RunTotal > 0 Then

```
HR_Total = SCatchTotal / RunTotal * 100
```

Else

```
HR_Total = 0
```

End If

```
harvest_rate = Format(HR_Total, "###.0") 'Round to 1 decimal place
```

```
Reconstruct_WkBk.Worksheets(Year_String).Cells(Reconst_CatchRow + 2, NumStocks + 2).Value = harvest_rate
```

```
HarvestRate_WkWB.Worksheets("StockGroup").Cells(NumSGroups + 3, YearCol).Value = HR_Total
```

```
HarvestRate_WkWB.Worksheets("Stock").Cells(NumStocks + 3, YearCol).Value = HR_Total
```

```
Escape_WkBk.Worksheets("StockGroup").Cells(NumSGroups + 3, YearCol).Value = Round(TotalEscape)
```

```
Escape_WkBk.Worksheets("Stock").Cells(NumStocks + 3, YearCol).Value = Round(TotalEscape)
```

End Sub