**Process For Getting All Fish Distribution (FD) Layers into a Single Feature Class that is aligned with NHD.**

Notes:

* Both SWIFD and the WDFW Potential Occupancy (DFW) layers appear to be aligned with NHD already for the majority of the study area. The exceptions appear to be in the Wind River subbasin for SWIFD, and in the Wind River subbasin and some of the other smaller upstream watersheds, for WDFW. (I’m not sure why things appeared out of alignment in the screen shares that I sent to you from Pro. There might be an issue with my map. When I bring everything into ArcMap it all lines up. You should see this too, but please confirm).
* For the Wind River basin and other areas where things are out of alignment, the easiest thing to do is probably to copy NHD features into the FD layer where they overlap, and then delete the original unaligned features. Since it’s only a small subset of the overall data this should be relatively easy. THIS SHOULD BE DONE TO ALL FD SOURCE LAYERS BEFORE COMPLETING THE STEPS BELOW. You can use a feature class that I created – ‘NHD\_align\_allFD’ to pull these features from. This layer should include the extent of all FD features that don’t already align with NHD. Note: you will want to preserve some of the attribute field (LLID, LLID\_STRM\_,DISTTYPE\_D, USETYPE\_DE for SWIFD; RID, GNS\_ID, GNIS\_STREAMNAME fo WDFW pot. occupancy) information, so make sure this information gets entered into the corrected features when they are added to the feature class.
* EDT layers don’t appear to align with NHD anywhere, we can deal with those later after getting the SWIFD and WDFW combined into a single layer.
* I created a new base NHD layer, that is consistent with the project extent (*lcrBarrierInv\_ProjArea*). It has the proper projection (NAD83 Washington State Plane South FIPS 4602, US feet) which we’ll be using for all project layers, and is titled ‘*Streams\_NHD\_lcr*’. You should replace the ‘*Streams\_NHD*’ you’ve been using with this updated version.
* We will use the following species separations:
	+ Chum (Combine SWFID and EDT. There is no DFW layer for chum)
	+ Coho (Combine SWIFD, DFW, EDT).
	+ Fall Chinook (Combine SWIFD, DFW, EDT)
	+ Spring Chinook (Combine SWIFD, EDT. There is no DFW layer for spring Chinook)
	+ Steelhead – hold off on this for now, I’m not sure if we’ll combine summer and winter into a single layer, or keep them separate.

Once all of the FD layers have been properly aligned (in the Wind River subbasin and nearby, for SWIFD/DFW) the following process can be followed.

Combining SWIFD and DFWPO. For Coho and Fall Chinook, using Coho as an example:

1. Start with the SWIFD\_2022 shapefile that is clipped to the project area (ie ‘*SWIFD\_2022\_LCR\_Coho’*). Re-project this to NAD83 Washington State Plane South FIPS 4602, US feet projection. Save it as a geodatabase feature class called ‘*LCR\_Coho*’. This will be the baseline FD layer for Coho.

1a) Also reproject the DFW source layer (‘*wdfw\_Coho\_Dist\_Model\_LCR*’) to that same project projection (NAD83 Washington State Plane South FIPS 4602, US feet), and save it to the same geodatabase. Call this ‘*DFW\_Coho*’.

1. In ‘*LCR\_Coho*’ add three integer fields: SWIFD, DFW, EDT. These will be used to flag which datasets each segment overlaps with.
2. For the SWFID field, populate all values with ‘1’ (since all these features came from SWIFD)
3. Next you’ll want to select feature segments from DFW that overlap with SWIFD. To do this easily, you’ll have to first split the SWIFD features at the endpoints of the WDFW features. Then you can use a select by location query and it will give you only the overlapping portions. To do this entire process:
	1. Use the [Polyline Feature Vertices to Points](https://pro.arcgis.com/en/pro-app/latest/tool-reference/data-management/feature-vertices-to-points.htm) gp tool to convert endpoints of ‘*DFW\_Coho*’ to a point feature class. Within the tool you’ll see the option to ‘convert both start and end vertices’. This is what you will choose. Output Feature Class = ‘*DFW\_Coho\_endpoints*’
	2. Use the [Split Line At Points](https://support.esri.com/en-us/knowledge-base/how-to-split-polyline-features-at-points-in-arcgis-pro-000029078) gp tool to split the ‘*LCR\_Coho*’ features at the endpoints of the DFW features. Input features = ‘*LCR\_Coho’*; Point Features = ‘*DFW\_Coho\_endpoints’;* Output Feature Class = ‘*LCR\_Coho\_split*’
	3. Use the Select By Location Query as follows:

Input features: ‘*LCR\_Coho\_split*’; Relationship: Share a line segment with ; Selecting Features: ‘*DFW\_Coho*’ ; Search Distance: leave blank (this seemed to work for me, but if some features are being omitted you might play around with this distance by setting to something very small like 10’ or so).

* 1. For the features that get selected in c. above, assign a value of 1 to the DFW field.
	2. Switch the selection. This will select all the features that don’t overlap. For these, assign a value of 0 to the DFW field.
1. Next we need to select features from the DFW dataset that don’t overlap with SWFID, and add them into our new combined feature class (‘*LCR\_Coho\_split’*). To do this easily we need to repeat the process outlined in 4) above but instead splitting the DFW segments based on the SWIFD endpoints and selecting the resulting isolated DFW segments. This is done as follows:
	1. Use the [Polyline Feature Vertices to Points](https://pro.arcgis.com/en/pro-app/latest/tool-reference/data-management/feature-vertices-to-points.htm) gp tool to convert endpoints of ‘*LCR\_Coho\_split*’ to a point feature class. Within the tool you’ll see the option to ‘convert both start and end vertices’. This is what you will choose. Output Feature Class = ‘*LCR\_Coho\_split\_endpoints’*
	2. Use the [Split Line At Points](https://support.esri.com/en-us/knowledge-base/how-to-split-polyline-features-at-points-in-arcgis-pro-000029078) gp tool to split the ‘*DFW\_Coho’* features at the endpoints of the SWIFD segments. Input features = ‘*DFW\_Coho’*; Point Features = ‘*LCR\_Coho\_split\_endpoints*’*;* Output Feature Class = ‘*DFW\_Coho\_split*’
	3. Use the Select By Location Query as follows:

Input features: ‘*DFW\_Coho\_split*’; Relationship: Share a line segment with ; Selecting Features: ‘*LCR\_Coho\_split*’ ; Search Distance: leave blank (this seemed to work for me, but if some features are being omitted you might play around with this distance by setting to something very small like 10’ or so).

* 1. Invert the selection from c) to get the features that don’t overlap.
	2. I found that even after adjusting distances some features that should have been selected did not get selected. This is because these features do not end at the endpoints we created. For these you’ll have to manually split the DFW features at at the junctions with the ‘*LCR\_Coho\_split*’ segments. Once you split them, you can redo steps c & d above and they should get selected (this should be obvious when doing this step).
	3. For the features that get selected in d. above, copy and paste these selected segments into the ‘*LCR\_Coho\_split*’ feature class. Assign them a value of 0 in the SWIFD field, and 1 in the WDFW field. (Alternatively, if copy and paste does not work in Pro – export the features from ‘*DFW\_Coho\_Split’* that were selected in d) above, and then load these into ‘*LCR\_Coho\_split’)*
1. Now you should have a complete FD layer for Coho (or Fall Chinook) that includes the full extent of the SWIFD and DFW feature classes, but with individual segments.
2. We should be able to do something similar to incorporate the EDT layer. But as mentioned above EDT first needs to be aligned to the NHD layer. You could try doing this using the [Align Features](https://pro.arcgis.com/en/pro-app/latest/tool-reference/editing/align-features.htm) gp tool in Pro. This could be a really simple way to do it if it works. Select the EDT layer as the input and the NHD as the target, and see what happens. Since the feature class gets overwritten, I would first make a copy of the EDT layer and work with that instead of the original.
3. If you get the EDT aligned, you can play around with combining EDT with the other layers using the process outlined above. We’ll want to add non-overlapping EDT segments to the ‘*LCR\_Coho\_split*’ feature class, and for the ones that do overlap we just need to add a 1 value to the EDT attribute field (in ‘*LCR\_Coho\_split’).*