EcoSHEDS Northeast Catchment Delineation (NECD)

Original Author:

Kyle O’Neil

Conte Ecology Group

U.S. Geological Survey

November 2015

Updated:

Jeffrey D. Walker, PhD  
Walker Environmental Research LLC

July 2022

Contents

[Overview 4](#_Toc109290871)

[Product Description 6](#_Toc109290872)

[Catchments and Flowlines 6](#_Toc109290873)

[Detailed Flowlines 8](#_Toc109290874)

[Riparian Buffer Polygons 8](#_Toc109290875)

[Product Availability 9](#_Toc109290876)

[System Requirements for Processing 9](#_Toc109290877)

[Source Data 10](#_Toc109290878)

[Flowlines 10](#_Toc109290879)

[Digital Elevation Models (DEM) 10](#_Toc109290880)

[Boundary Layers 11](#_Toc109290881)

[Processing Description 12](#_Toc109290882)

[Pre-Delineation Processing 12](#_Toc109290883)

[Directory Structure 12](#_Toc109290884)

[Create Boundary Files 14](#_Toc109290885)

[DEM Processing 16](#_Toc109290886)

[Streams Processing 17](#_Toc109290887)

[Delineation Processing 20](#_Toc109290888)

[Set Target Locations 20](#_Toc109290889)

[Topography Processing 20](#_Toc109290890)

[Streams Processing 22](#_Toc109290891)

[Remove False Headwaters 24](#_Toc109290892)

[Delineate Catchments 27](#_Toc109290893)

[Coastal Catchment Delineation 28](#_Toc109290894)

[Finalize Hydrography Layers 31](#_Toc109290895)

[Post-Delineation Processing 32](#_Toc109290896)

[Prepare Layers for Editing 32](#_Toc109290897)

[Correct Overlapping Sections 33](#_Toc109290898)

[Correct gaps within each catchment layer 33](#_Toc109290899)

[Correct gaps between catchment layers 33](#_Toc109290900)

[Manual Catchment Adjustments 34](#_Toc109290901)

[Correct overlapping sections within catchment layers 34](#_Toc109290902)

[Export Hydrography Products 35](#_Toc109290903)

[Generate Detailed Flowlines 35](#_Toc109290904)

[Generate Riparian Buffers 35](#_Toc109290905)

[Caveats 37](#_Toc109290906)

[Detached Streams 37](#_Toc109290907)

[Forced Region Boundaries 37](#_Toc109290908)

[Divergences 37](#_Toc109290909)

[Detailed Streams Overlay Method 37](#_Toc109290910)

[Riparian Buffer Creation 38](#_Toc109290911)

[Small Tributary Omission 39](#_Toc109290912)

[References 41](#_Toc109290913)

# Overview

The EcoSHEDS Northeast Catchment Delineation (NECD)[[1]](#footnote-1) is a series of GIS layers representing hydrologic catchments and flowlines spanning the Northeast region of the United States. The catchments are similar to existing products such as the well-known NHD Plus dataset. Catchments are based on the National Hydrography Dataset (NHD) and contain fields to indicate routing paths which can be used to establish the network structure. Some basic information about the layers is also included in the attribute tables. Layers representing riparian buffer zones in each catchment are developed as well.

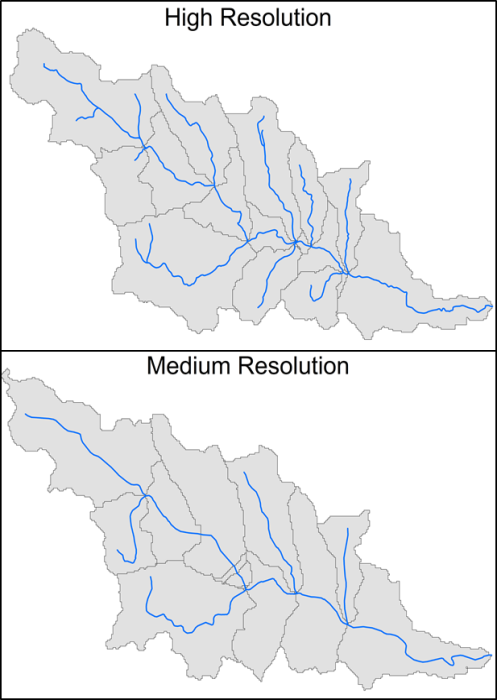


Figure : Catchment resolution comparison in the Westbrook, Whately, MA.

The NECD seeks to address specific aspects of existing hydrography products that have proven to restrict their usefulness. Specifically, improved spatial resolution and consistency are the focus of the development of these products. Catchment delineation is based on the high resolution NHD flowlines. Currently, there are no known large scale catchment products derived from this more detailed dataset[[2]](#footnote-2). One main focus of this product is to capture small headwater streams omitted by existing products such as the medium resolution catchments in NHDPlus Version 2. Representing these streams can be critical to understanding key ecological interests. Figure 1 shows the added catchment detail from the high resolution streams in the Westbrook stream in Whately, MA.

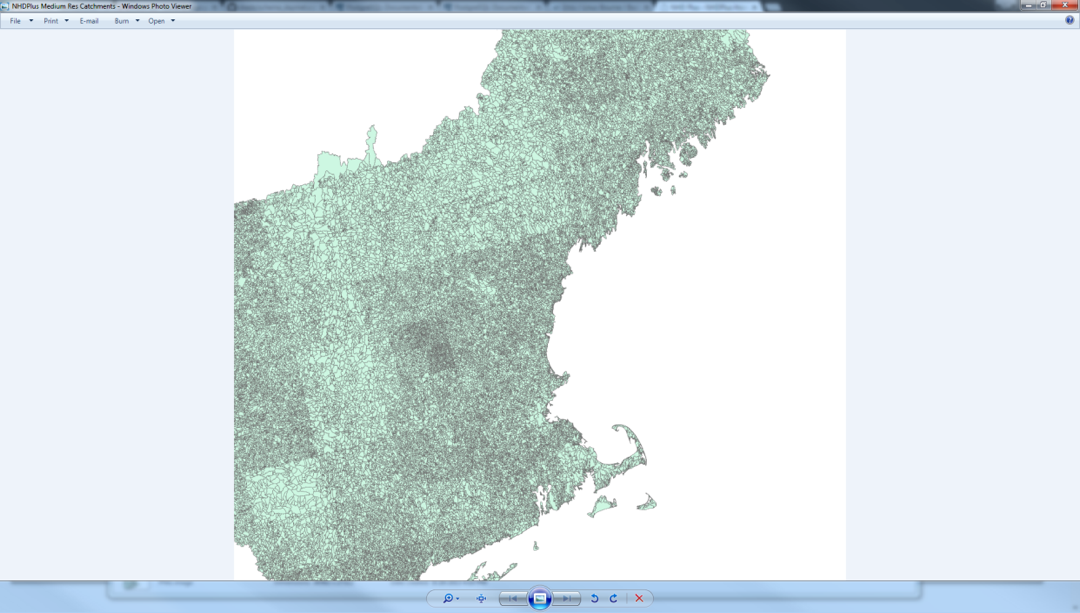


Figure : NHD Plus Version 2 medium resolution catchments in the Northeast.

Resolution consistency is addressed in the creation of the high resolution catchments. The NECD catchments, derived from the medium resolution NHD flowlines, are unnaturally inconsistent in area. The inconsistency is evidenced by Figure 2, which shows a regional view of the catchments. The unnatural catchment area differential shows up as a straight line between lighter and darker shades, indicating lesser and greater density of catchments respectively. These inconsistencies permeate the entire dataset and are a result of the varying resolution of existing hydrography data in the region. The areal inconsistency presents a problem for models relying on drainage area as a driver as well as for visual representation of data layers. The issue shown here in the medium resolution layers also exists in the high resolution dataset. It is evident from examining the flowline layers that the dataset has been processed in a gridded format at differing resolutions. A methodology for improving areal consistency of the catchments has been developed. This methodology is described in the next section and fully detailed in the *Processing Description* section. The final products adhere to an established areal consistency while managing to include higher resolution streams than the medium resolution products.

# Product Description

## Catchments and Flowlines

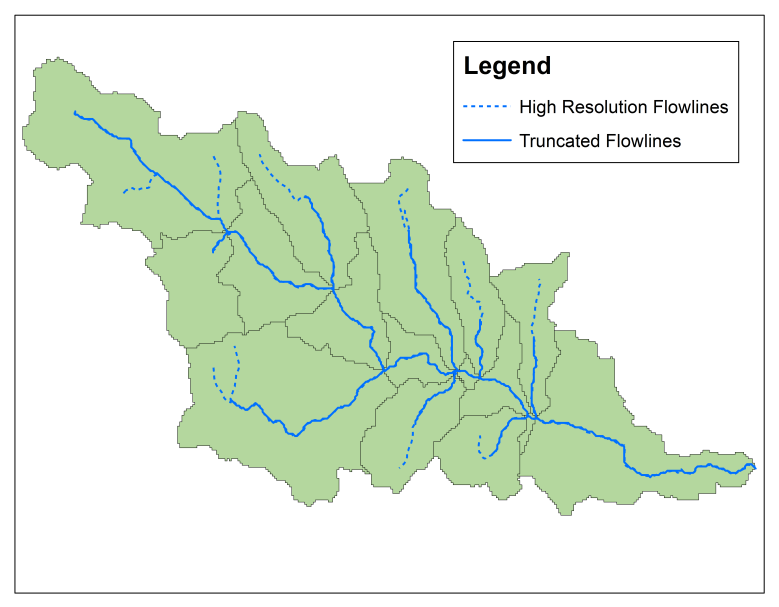


Figure : Comparison of truncated and high resolution flowlines

The hydrography layers are derived from the NHD high resolution flowlines and a 30m resolution digital elevation model (DEM). The NHD flowlines are burned into the DEM with a buffer zone to force the representation of the high resolution network in the layers used for delineation. A flow accumulation grid is generated and used to define a second set of flowlines, termed the “truncated flowlines”, which are initiated at points with a contributing drainage area of 0.75 km2 or greater. A comparison of these flowlines with the high resolution layer is shown in Figure 3. These truncated flowlines define the respective catchments during the delineation processing. The purpose of setting this threshold is to reconcile differences in spatial resolution within the high resolution dataset. The catchment layer generated from the truncated flowlines is more spatially consistent than the raw high resolution dataset. Spatial consistency is a particularly important aspect when used in models relying on drainage area as a driver. Through burning the original flowlines into the DEM, the high resolution hydrography is preserved in the resulting dataset.

Catchments and flowlines are uniquely identified by the “FEATUREID” field in the attribute table. This field links each stream segment to the catchment it falls inside. The first digit in the FEATUREID identifies the product version, the second and third digits identify the hydrologic region, and the remaining digits are an ID that is unique to the region (Figure 4). The “NextDownID” field is used for network routing and grouping catchments into larger watersheds. This field identifies the FEATUREID of the catchment or flowline immediately downstream. The “Source” field identifies the manner in which the catchment or flowline was created (Figure 5). There are three source types: Delineation, Coastal, and Coastal Fill. The “Delineation” source indicates that the feature is created in the primary delineation processing. The “Coastal” source indicates that the feature is created during the delineation specific to coastal areas. “Coastal Fill” catchments are created to fill in the areas along the coastal regions that were too small to be delineated. There are no flowlines associated with the Coastal Fill catchments. These categories are described in more depth in the *Delineation Processing* section.

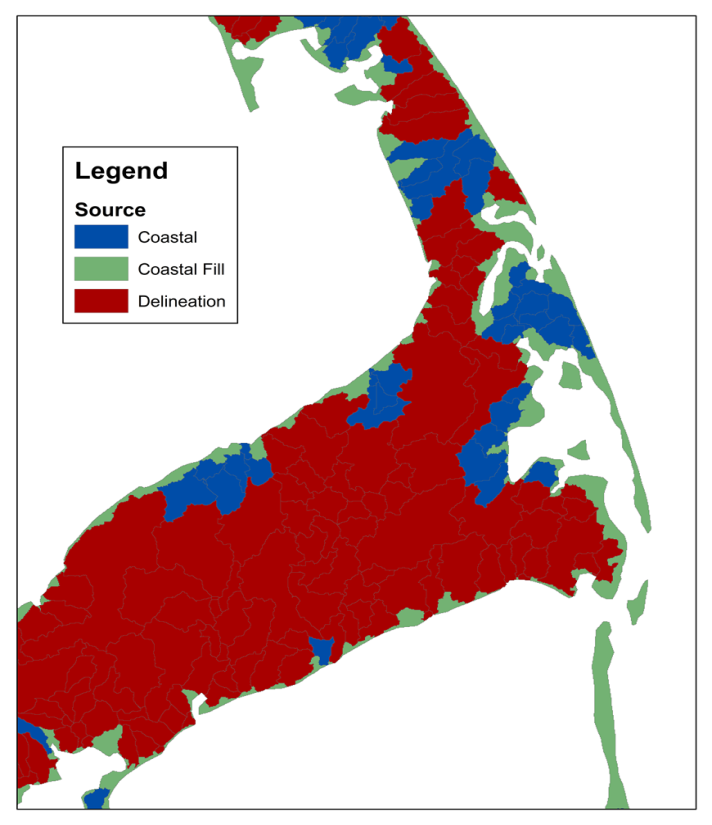


Figure : Catchment classification example

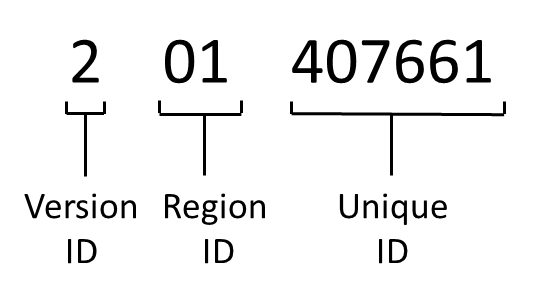


Figure : FEATUREID description

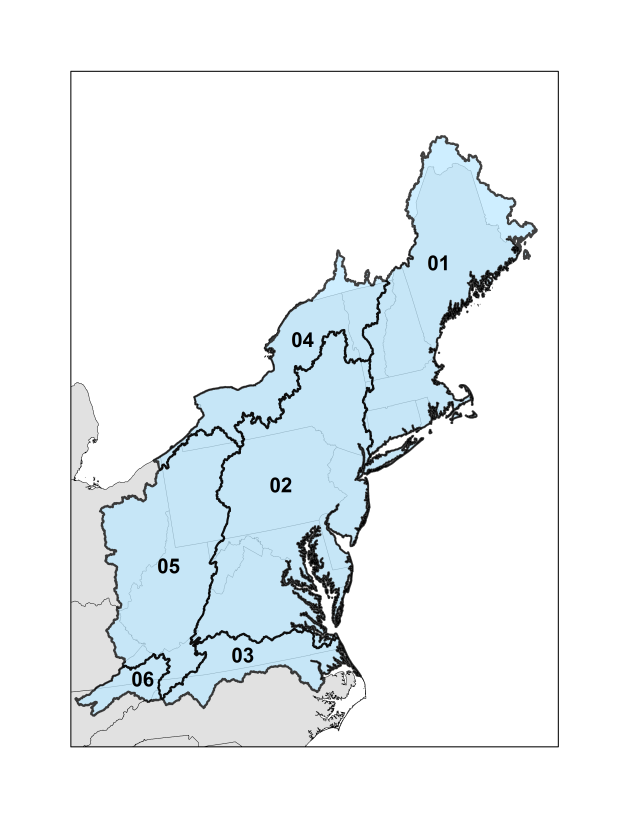


Figure : Hydrologic regions

The hydrologic regions by which the hydrography layers are grouped are derived from the HUC 2 regions established by the high resolution NHD (Figure 6). The full range of the Designing Sustainable Landscapes (DSL) project is included in this product version. The entirety of NHD regions 01 and 02 are processed while only portions of regions 03, 04, 05, and 06 have been completed. Partial regions are split along hydrologic boundaries to avoid artificial breaks in the stream network. In order to adhere to hydrologic boundaries, regional coverage extends beyond the DSL project range, which is denoted by political boundaries. This range expansion is an improvement from Version 1 of this product, which contains watersheds divided along political boundaries. The range of this version ensures that no artificial boundaries are created. The expansion area beyond the DSL project range is shown in Figure 7.

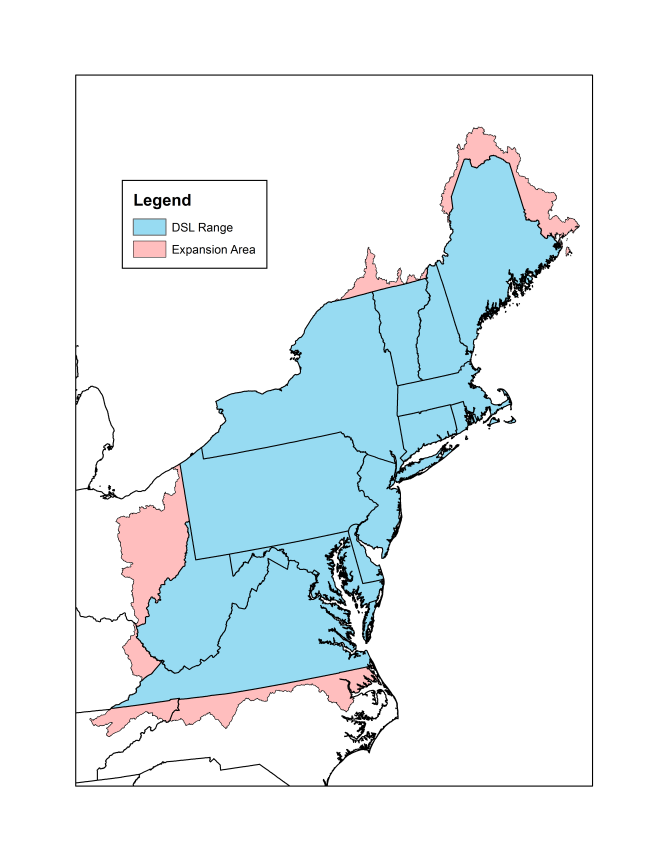


Figure : Range comparison between versions

The flowlines that accompany the catchments are based on the truncated flowlines. Each catchment has one stream reach from the truncated flowlines layer associated with it, whereas multiple reaches from the raw high resolution streams may fall into a single catchments (refer to Figure 3). The truncated flowlines are the product of rasterizing the high resolution flowlines, trimming to the drainage area threshold, converting back to polylines, and smoothing. As a result of these editing steps, the lines will differ slightly from the raw high resolution lines. The delineation resolution may also cause truncated flowline segments to cross catchment boundaries at their endpoints. This is important to note for functional purposes. Hydrography layer statistics by each region can be viewed in Table 1.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Hydrologic Region | Number of Catchments | Mean Catchment Area (km2) | Median Catchment Area (km2) | Number of Flowlines | Mean Flowline Length (km) | Median Flowline Length (km) |
| 01 | 92312 | 1.94 | 1.39 | 90402 | 1.43 | 1.12 |
| 02 | 157861 | 1.6 | 1.22 | 154129 | 1.28 | 1.01 |
| 03 | 32845 | 1.49 | 1.16 | 32361 | 1.2 | 0.94 |
| 04 | 49329 | 1.67 | 1.26 | 48802 | 1.43 | 1.09 |
| 05 | 109313 | 1.44 | 1.16 | 109313 | 1.18 | 0.96 |
| 06 | 12164 | 1.42 | 1.13 | 12164 | 1.18 | 0.92 |

Table : Summary hydrography statistics

## Detailed Flowlines

The high resolution flowlines are mapped to the catchments based on spatial location using a simple overlay methodology to create the “detailed flowlines” layer. All streams within the same catchment are converted to a single feature and assigned the FEATUREID of that catchment (Figure 8). As a result of this simplified mapping technique the finest network structure resolution is on the catchment scale. No intra-catchment network structure is established. There are also some issues involved with the methodology used to create this layer, which are discussed in the *Caveats* section. The detailed flowlines serve the purpose of a visual tool more than a functional network layer. The layer provides the highest available resolution when mapping observed data locations to the network. This layer also serves as the basis for a riparian buffer layer described in the next section. The detailed flowlines are subject to varying resolutions as consistency is not forced in this layer.

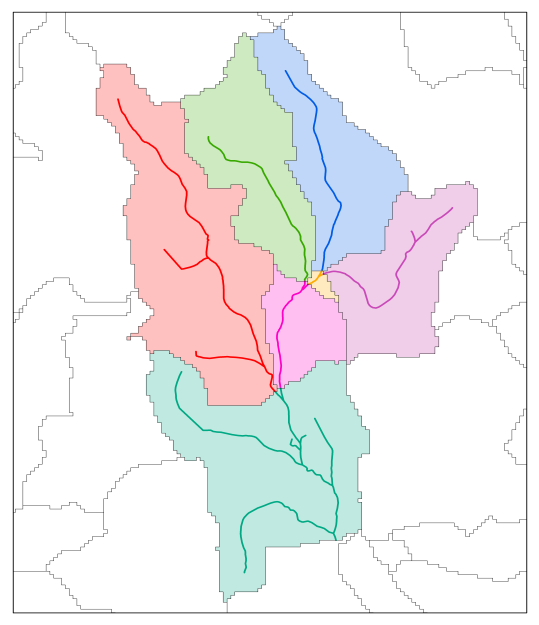


Figure : Detailed flowline features and corresponding catchments

## Riparian Buffer Polygons

Riparian buffer layers exist for the detailed flowlines in each hydrologic region. Each region has three buffer lengths, 50, 100, and 200 feet. Buffer polygons are linked to their respective catchments and flowlines by a common FEATUREID. Buffer polygons are restricted to the catchment within which they fall (Figure 9). This aspect improves spatial functionality of the layer by preventing overlapping polygons within the layer. The restriction also forces the buffer layers to be congruent with the other layers in the delineation by not allowing spatial overlap with adjacent catchments. The method produces a few issues related to the riparian buffer layers are discussed in the Caveats section. Because they are derived from the detailed flowlines, the riparian buffer layers vary in spatial resolution.

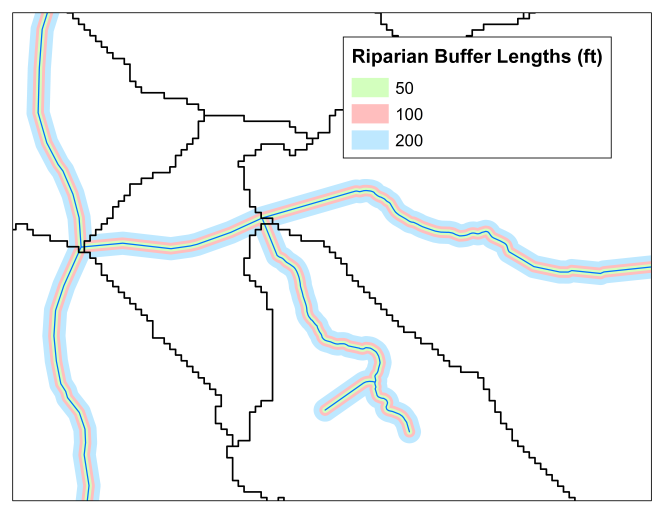


Figure : Example of riparian buffer layers

## Product Availability

Catchment and flowline layers are available from the Spatial Hydro-Ecological Decision System (SHEDS) at http://ecosheds.org/. Hydrologic regions are identified by the 2 digit identifier at the end of the layer name (e.g. Catchments01). All spatial layers are provided in the Albers projected coordinate system:

**Spatial Information**

Projected Coordinate System: NAD\_1983\_Albers

Projection: Albers

False\_Easting: 0

False\_Northing: 0

Central\_Meridian: -96

Standard\_Parallel\_1: 29.5

Standard\_Parallel\_2: 45.5

Latitude\_Of\_Origin: 23.0

Linear Unit: Meter

Geographic Coordinate System: GCS\_North\_American\_1983

Datum: D\_North\_American\_1983

Prime Meridian: Greenwich

Angular Unit: Degree

## System Requirements for Processing

* ArcGIS 10.0 or higher
* ArcGIS Spatial Analyst extension
* Arc Hydro Tools Version 2.0
* R Version 3.1.0 or higher

# Source Data

The NECD takes advantage of some of the layers from the DSL Project which involved extensive work to clean up the NHD high resolution flowlines. This effort is the foundation of the NECD products. While the DSL flowline editing methodology was applied to the expansion area in Figure 7, any further additions to the spatial extent of the delineation products will require a significant effort to edit flowlines outside of this zone.

## Flowlines

The flowline layers are used in delineation processing to establish the stream network throughout the landscape.

*DSL Processed Flowlines*

Layer name: *allstreams*

Source: DSL Project

Description: The NHD high resolution flowlines that have been edited to eliminate network gaps, correct flow direction, and remove coastline. The range of this dataset is depicted in blue in Figure 7.

*NHD High Resolution Flowlines*

Layer name: NHDFlowline

Source: National Hydrography Dataset

Description: The NHD flowlines are downloaded as separate geodatabases based on HUC 4 boundaries. The downloaded geodatabases are saved into separate folders named for each hydrologic region they are a part of (e.g. “NHDH01”). These layers are used to fill in the areas within the processing region that the DSL flowlines do not cover. The processing range of this dataset is depicted in red in Figure 7.

## Digital Elevation Models (DEM)

The DEMs are used to establish the flow paths over the delineation ranges by providing topographic information.

*DSL Processed DEM*

Layer name: *dem*

Source: DSL Project

Description: The DSL processed DEM is a mosaic of 30m resolution National Elevation Dataset (NED) DEMs. It matches the spatial coverage of the DSL flowlines.

*Raw NED 30 Meter DEMs*

Layer name: *grdn45w067\_1*

Source: National Elevation Dataset

Description: These raw 30-meter resolution DEM tiles are downloaded from the NED. All tiles necessary to provide coverage for the red area in Figure 7 are downloaded. The downloaded tiles are saved into separate folders by hydrologic region. Each folder (e.g. “NHDH01”) contains all of the tiles that overlap the region, meaning some tiles are saved in multiple folders. The purpose of this structure is for the automation of DEM mosaicking. Any extra layers in a folder will result in extra processing effort or error. The “Layer Name” above is an example of one of the raw DEMs, which are named by their latitude and longitude coordinates.

## Boundary Layers

The boundary layers are used to generate the hydrologic region boundary layers shown in Figure 6, as well as to segment the flowlines and DEMs for processing.

*NHD Water Boundary Dataset (WBD) Layers*

Layer Name: WBDHU4

Source: National Hydrography Dataset

Description: The HUC4 watershed boundary layers are downloaded as a part of the high resolution NHD geodatabases.

*U.S. State Boundary Layer*

Layer Name: States

Source: The National Map

Description: This shapefile represents state political boundaries of the United States.

*Canada Provinces*

Layer Name: PROVINCE.shp

Source: National Weather Service OST/SEC GIS Map Group

Description: This layer represents all of the provincial boundaries in Canada.

New Brunswick

Layer Name: geonb\_provinciallimits-limitesprovinciales.shp

Source: GeoNB

Description: This additional boundary layer represents the political boundary of New Brunswick and is primarily used to establish the coastline over a portion of hydrologic region 01.

# Processing Description

This section is an in-depth walkthrough of how the hydrography products are created. The *Pre-Delineation Processing* and *Delineation Processing* sections are completed for one hydrologic region at a time. The *Post-Delineation Processing* section works simultaneously with products from all regions and is only started once all of the hydrologic regions in the version have been delineated. The file naming convention is consistent throughout the hydrologic regions. Layers are typically identified by appending the hydrologic region identifier onto their name (e.g. the DEM for region 01 is *dem01*). All input variable names are consistent throughout all scripts and therefore are only described the first time they are mentioned in this section.

## Pre-Delineation Processing

The steps in this section prepare the layers, by hydrologic region, for the catchment delineation process. The primary purpose of this section is to set up the directory structure, establish regional boundaries, and edit stream and DEM layers.

### Directory Structure

A script automatically establishes the directory structure used for the delineation processing. In order to maintain a consistent structure throughout the processing steps, scripts executed later in the delineation process depend on this structure.

#### Repos Created

*Main GIS Directory*

Name: gisFiles

Location: Base Directory

Description: The main GIS directory is the parent folder of all GIS processing.

*Product Directory*

Names: products

Location: Base Directory

Description: The product directory is where the finalized layers are stored.

*Hydrologic Region Directory*

Name: NHDH01

Location: Main GIS Directory

Description: The hydrologic region directory is the parent directory specific to each hydrologic region.

*Arc Hydro Directory*

Name: arcHydro

Location: Hydrologic Region Directory

Description: The Arc Hydro directory is where all of the Arc Hydro processing files are saved.

*Arc Hydro Vector Geodatabase*

Name: vectors.gdb

Location: Arc Hydro Directory

Description: The Arc Hydro vector geodatabase is where Arc Hydro saves all of the vector layers.

*Processing Layers Geodatabase*

Name: processing.gdb

Location: Hydrologic Region Directory

Description: The processing layers geodatabase is where all of the interim files are saved during the pre-delineation processing steps.

*Boundary Layers Geodatabase*

Name: boundaries.gdb

Location: Hydrologic Region Directory

Description: The boundary layers geodatabase holds all of the files establishing boundaries used in the delineation.

*Arc Hydro Input Geodatabase*

Name: arcHydroInput.gdb

Location: Hydrologic Region Directory

Description: The Arc hydro input geodatabase holds the completed flowlines and DEM used as input to the Arc Hydro process.

*Post-Delineation Processing Geodatabase*

Name: postProcessing.gdb

Location: Hydrologic Region Directory

Description: The post-delineation processing geodatabase is used by the post-processing delineation script in joining the separate catchment layers into one final product.

*Main Post Processing Directory*

Name: postProcessing

Location: Main GIS Directory

Description: The main post processing directory is where the inter-layer post processing occurs.

*Manual Edits Geodatabase*

Name: manualEdits.gdb

Location: Main Post Processing Directory

Description: The manual edits geodatabase is where the inter-layer post processing layers are saved for manual editing.

*Product Geodatabase*

Name: hydrography

Location: Product Directory

Description: The product geodatabase is where the final products are saved.

#### Processing Steps

1. The variables in the “Define Inputs” section of the "preprocessing - regionDirectories.py" script are set:

*baseDirectory* – the parent folder of the entire project (e.g. “C:/Projects/Delineation”)

*regionHUC2* – The hydro region identifier (e.g. “01”)

1. The script is executed in Arc Python.

### Create Boundary Files

The delineation is segmented by hydrologic region. In order to establish these regional outlines, various boundary layers are generated. These layers are used both for visual reference and for processing hydrography layers in later scripts. This section describes the creation of the boundary files.

#### Layers Created

*Hydrologic Boundary Layer*

Name: hydroBoundary01

Location: boundaries.gdb

Description: The hydrologic boundary layer is defined by joining all of the HUC 4 boundaries within the WBD included in the HUC 2 NHD region. This boundary is used to segment the delineation processing into separate hydrologic regions.

*Political Boundary Layer*

Name: politicalBoundary01

Location: boundaries.gdb

Description: This layer is the combined boundary of all of the political regions (e.g. states & provinces) that overlap the current hydrologic region.

*Processing Boundary Layer*

Name: processingBoundary01

Location: boundaries.gdb

Description: This layer is a combination of the hydrologic and political boundary layers. It is used to define the area that undergoes delineation processing for a particular hydrologic region. This layer establishes the coastal boundary while maintaining a buffer on land. The purpose of this buffer is to establish the regional boundary through the delineation process as opposed to a pre-defined boundary between regions. This step ensures a delineation product based on the hydrography and not on existing layers from an external source.

*Terrestrial Boundary Layer*

Name: terrestrialBoundary01

Location: boundaries.gdb

Description: On land, this layer is identical to the Hydrologic Boundary Layer. The layer boundary extends out beyond coastal areas to encompass the entire coastline established in the processing boundary layer. This layer is used to separate the streams being delineated from the streams burned into the DEM over the entire processing boundary. The layer is another tool used in the process of establishing hydrologic region boundaries through delineation.

#### Processing Steps

1. If the entire hydrologic region is being processed, the “preProcessing - hydroBoundaryLayer.py” script will create the *hydroBoundary01* layer based on the HUC4 boundary features in the NHD High Resolution dataset. Before running the script in Arc Python, the following variables are set in the "Define Inputs" section:

*baseDirectory* & *regionHUC2*

*projectionDefinitionFile* – The GIS layer used to define the projection used by the delineation. In this version, the DSL DEM is used.

*nhdDirectory* – The directory where all of the downloaded raw high resolution NHD layers are saved by hydrologic region subfolders as outlined in the *Source Data* section.

The script is executed in Arc Python. If the processing area does not encompass the entire hydrologic region, this layer must be created by manually editing WBD polygons together. Care should be taken to avoid unnatural segmentation of watersheds and only establish flow outlets at downstream points.

1. If not already isolated as stand-alone layers, all of the features (states & provinces) in the political boundary layers which overlap the *hydroBoundary01* layer are exported as new layers. If the hydrologic boundary coincides with a state boundary (e.g. the Ohio River) the adjacent states are included.
2. All of these exported layers from step 2 are projected to match the *hydroBoundary01* file created in Step 1.
3. The projected layers are joined together using the *Union* tool (Analyisis Tools 🡪 Overlay 🡪 Union). This step is skipped if all processing is within the United States, as the states are already one layer.
4. The *Dissolve* tool (Data Management Tools 🡪 Generalization 🡪 Dissolve) is run on the joined political boundaries allowing multipart features. The result is saved as *politicalBoundary01*.
5. This *politicalBoundary01* layer is manually edited to remove the gaps created along the border between layers from different sources (e.g. the US and Canada layers).
6. A 5km buffer is added to the *hydroBoundary01* layer using the *Buffer* tool (Analyisis Tools 🡪 Proximity 🡪 Buffer).
7. The result of the buffering step is trimmed to the *politicalBoundary01* layer using the *Clip* (Analyisis Tools 🡪 Extract 🡪 Clip) tool. The result is saved as *processingBoundary01*. This final step creates the coastline boundary, while maintaining a buffer on the inland watershed boundary for establishing the hydrologic boundary naturally through delineation. This clipping step is necessary to maintain a consistent costal boundary, which varies in resolution within the WBD.
8. A copy of the *hydroBoundary01* layer is saved as *terrestrialBoundary01*. This layer is manually edited to extend the border to encompass the entire coastline in the *processingBoundary01* file. On land, this shape is identical to the *hydroBoundary01*. This layer is essentially the opposite of the *processingBoundary01* file, with a buffer into the ocean and an exact boundary on land.

### DEM Processing

This section prepares the DEM layer used in the catchment delineation. This DEM is a combination of the DSL DEM layer and raw DEM tiles downloaded from NED.

#### Layers Created

**Processing DEM**

Name: dem01

Location: arcHydroInput.gdb

Description: The processing DEM is a raster layer created from a combination of the DSL and NED DEMs. The NED tiles are used to fill in the area missed by the DSL DEM. This is the base layer used to delineate the catchments.

**DEM Outline**

Name: demOutline01

Location: boundaries.gdb

Description: The DEM outline is a polygon layer representing the outline of the full processing DEM, which will be the boundary of the catchment delineation.

**Missing DEM outline**

Name: missingDEMZone01

Location: processing.gdb

Description: The missing DEM outline is a polygon layer representing the area within the processing boundary layer that is not covered by the DSL DEM. This layer is used to select the raw high resolution streams used to fill in area missing from the DSL streams.

#### Processing Steps

1. The *processingBoundary01* is visually compared to the DSL DEM. The area missing coverage from this layer is used as a guide to download the appropriate tiles from the NED website.
2. The variables in the "Define Inputs" section of the "preprocessing - DEM.py" script are set:

*baseDirectory* & *regionHUC2*

*demFilePath* – The file path to the DSL DEM

*nedDirectory* – The parent directory where all of the raw NED DEM tile subfolders are saved as described in the *Source Data* section.

1. The script is executed in Arc Python. The portion of the DSL DEM within the current hydrologic region is extracted. The NED DEM tiles are mosaicked to the DSL DEM to fill in the areas missing DEM coverage (red areas in Figure 7). The output of this script is the *dem01*, *demOutline01*, and*missingDEMZone01*layers.

### Streams Processing

This section is similar to the previous section in that it prepares the streams layer used to delineation the catchments. This layer is a combination of the DSL streams and raw NHD flowlines.

#### Layers Created

*Missing Streams*

Name: missingStreams01

Location: processing.gdb

Description: The missing streams vector layer represents the raw high resolution NHD flowlines in the region missing DSL flowlines. "Coastline" and "Pipeline" flowline classifications are removed. This interim layer is used as a visual for editing the final streams layer.

*DSL Streams*

Name: umassStreams01

Location: processing.gdb

Description: The DSL streams vector layer represents the DSL corrected streams that fall within the *demOutline*. This interim layer is used as a visual reference for editing the final streams layer.

*Combined Streams*

Name: allStreams01

Location: processing.gdb

Description: The combined streams vector layer is the raw output of the stream processing script. It is the result of merging the DSL streams and the missing streams layers. The layer is manually edited, as described in the next section, and saved as the final streams layer.

*Full Range Streams*

Name: burnStreams01

Location: arcHydroInput.gdb

Description: The full range streams layer is based on the high resolution flowlines for the current hydrologic region. It is the product of editing the combined streams layer. Similar to the DEM processing, this layer uses the DSL streams layer where possible and completes the missing areas with the raw NHD high resolution streams. The range of this layer is equivalent to the *demOutline* layer. The purpose of this range is to burn the surrounding streams into the DEM so that the delineation process creates a product-specific boundary between hydro regions, rather than forcing the boundary from the NHD boundary layers.

*Delineation Range Streams*

Name: delineationStreams01

Location: arcHydroInput.gdb

Description: The delineation range streams layer is identical to the full range streams layer, but does not extend coverage into the 5km buffer around the hydrologic region. These streams are used to delineate the actual catchments in the current hydrologic region.

#### Processing Steps

##### Merge layers

1. The *missingDEMZone01* layer created from the DEM processing is used to determine which NHD regions are needed to fill in the missing streams. These files are downloaded from the USGS's *The National Map – Get NHD Data* website.
2. The downloaded file geodatabases are saved to a directory specific to the current hydrologic region following the naming scheme: “NHDH” + *regionHUC2* (e.g. "NHDH01"). The parent directory of this folder is the *nhdDirectory* (defined in step 3). Each hydrologic region subfolder should contain all of the HUC4-based geodatabases necessary for filling in the missing area of that region. The script will process all of the geodatabases in this folder, so no other files should be contained within it.
3. The appropriate variables in the "Define Inputs" section of the "preProcessing – streams.py" script are set:

*baseDirectory,* *regionHUC2*, *nhdDirectory,* & *projectionDefinitionFile*

*streamsFile* – The DSL high resolution streams layer

1. The script is executed in Arc Python. The portion of the DSL streams layer within the current hydrologic region is extracted. The raw NHD flowlines are used to fill in the areas missing coverage. The resulting layer, *allStreams01,* is backed up before proceeding to the manual editing steps.
2. The *allStreams01* layer is examined for sections inside the *demOutline01* boundary that are missing streams. This omission is possible in sections of the 5km buffer zone that fall outside of the DSL processing range and current hydrologic region. Running *Select by Location* on the NHD layers containing missing streams using the *demOutline01* will isolate these streams which can be merged with the *allStreams01* layer. Streams that are identical to the ones already in the *allStreams01* layer are removed in the *Select by Location* step. Streams that clearly do not belong (e.g. streams removed in the DSL project) are manually removed. There are only a handful of these cases.

##### Clean up streams layer

1. The *allStreams01* layer is the result of merging the DSL streams (*umassStreams01*) and missing streams (*missingStreams01*). The merge creates some overlapping segments along the boundary of these layers which must be cleaned up manually. The basic concept is to complete the seamless stitching of the stream layers together by editing *allStreams01*.
   1. The *missingStreams01* layer is compared with *allStreams01* and *umassStreams01* to see which segments should be deleted in *allStreams01*.
   2. The *umassStreams01* layer reflects changes made for the DSL project to bound flow direction. Because of this bounding, the *missingStreams01* layer takes precedent over the *umassStreams01* along hydrologic region borders, though the changes in *umassStreams01* are considered. The raw NHD flowlines are helpful in determining the correct changes to make.
   3. Disconnected stream segments are reattached or, if small, deleted. The raw NHD flowlines and the DEM are used for reference in this step.
   4. Stream segments that cross the *terrestrialBoundary01* are examined. If these streams are the downstream terminus of a watershed, they are kept intact. Some streams, typically canals and minor headwater divergences, connect to segments in the buffer zone which flow into the adjacent watershed outside of the *processingBoundary01*. If these streams are not the terminus of the watershed they need to be edited so they do not connect to segments outside of the *terrestrialBoundary01*.
   5. Streams on opposite sides of the *terrestrialBoundary01* that come within 60 meters of each other are edited to avoid the incidental joining during rasterization. Both of these steps avoid the creation of false downstream watershed termini. Differentiation between errors to be fixed and flowlines correctly crossing the boundary is completed manually.
   6. When the manual editing is complete, the layer is exported as *burnStreams01*.
2. A "RasterVal" column (short integer) is added to *burnStreams* and set all values equal to 1. This column is used to rasterize the flowlines.
3. Streams that fall within the *terrestrialBoundary* are selected to capture the network of the current hydrologic region.
4. Some streams belonging to the current hydrologic region will be omitted by the “Select by Location” tool because they fall outside of the pre-defined boundary. A manual check along the hydrologic boundary is performed and streams that belong in the current network are added to the selection.
5. The selected features are exported as the *delineationStreams* layer.

## Delineation Processing

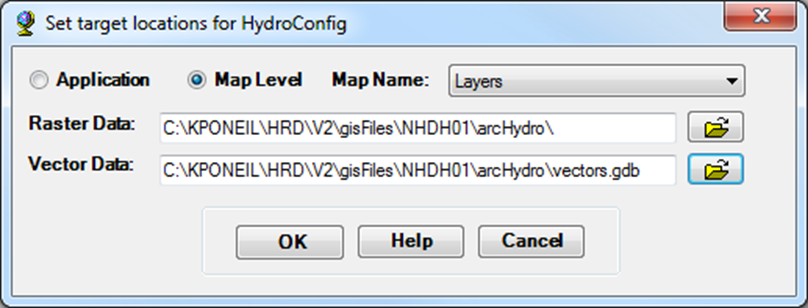
In this section the Arc Hydro Tools are used with the layers developed in the *Pre-Delineation Processing* section to establish a network of catchments and flowlines for each hydrologic region.

### Set Target Locations

Before beginning the processing steps in Arc Hydro, the default output directories must be set. These directories are set by running the *Set Target Locations* tool (Arc Hydro Tools > App Utilities > Set Target Locations) and selecting “HydroConfig”.

Raster Data: …\NHDH01\arcHydro\

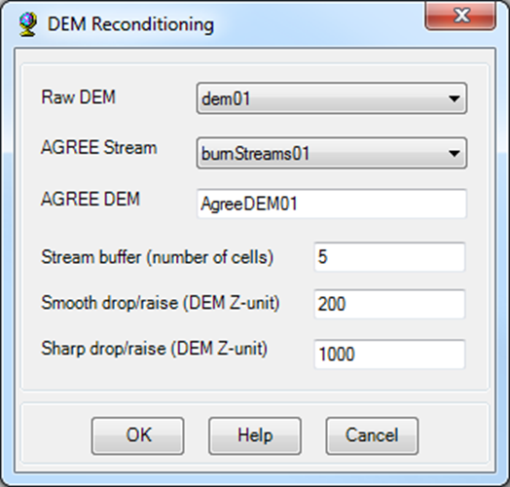
Vector Data: …\NHDH01\arcHydro\vectors.gdb



### Topography Processing

The DEM is edited to reflect the structure of the high resolution stream network. Flow direction and accumulation grids are produced in this section.

#### Burn streams into DEM

The high resolution streams are burned into the 30m DEM using the *DEM Reconditioning* tool (Arc Hydro Tools > Terrain Processing > DEM Manipulation > DEM Reconditioning). This step ensures that the high resolution streams are reflected in the topography used for delineation.

Raw DEM: *dem01*

AGREE Stream: *burnStreams01*

AGREE DEM: *AgreeDEM01*

Stream buffer: 5

Smooth drop/raise: 200

Sharp drop/raise: 1000

#### Remove excess cells

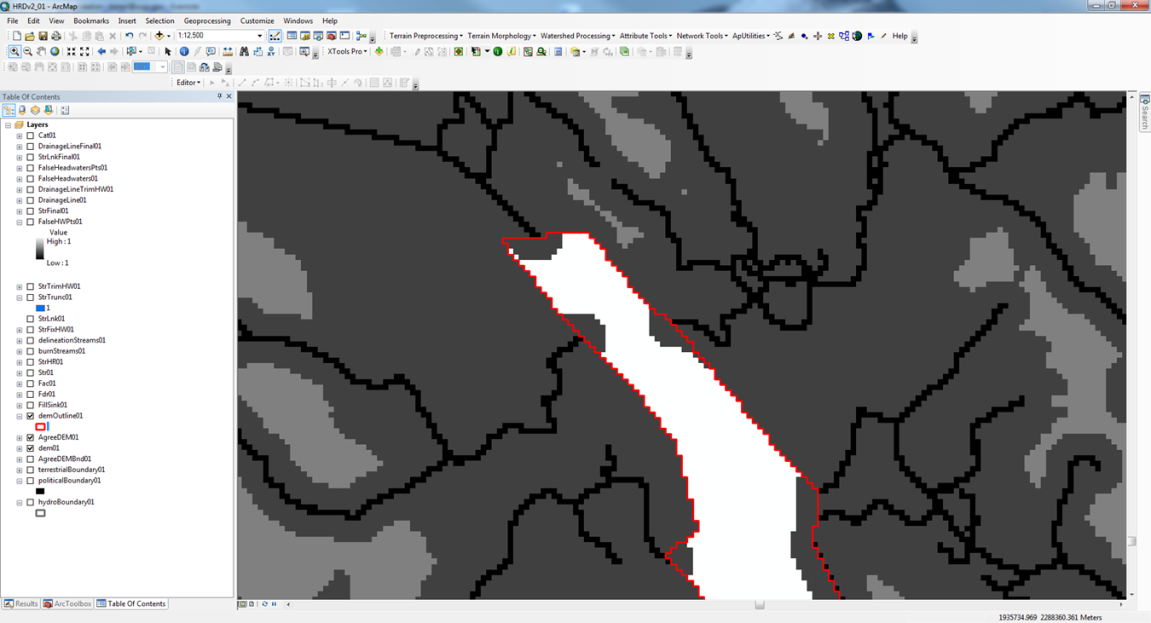
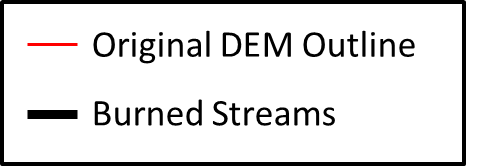


Figure : Extra buffer cells resulting from burning streams into DEM

Burning the streams into the DEM with a buffer creates excess cells around the edges of the DEM (see Figure 10). The DEM outline is used to remove the excess cells by running the Extract By Mask tool (Spatial Analyst Tools > Extraction > Extract by mask).

Input Raster: *AgreeDEM01*

Input Raster or feature mask data: *demOutline01*

Output raster: *AgreeDEMBnd01*

Output Location: NHDH01/arcHydro/Layers

#### Fill all sinks in the DEM

Sinks may exist in the DEM as artifacts from creation or from burning disconnected stream segments. The *Fill Sinks* tool (Arc Hydro Tools > Terrain Processing > DEM Manipulation > Fill Sinks) is run to ensure that all cells in the DEM drain into the network.

DEM: *AgreeDEMBnd01*

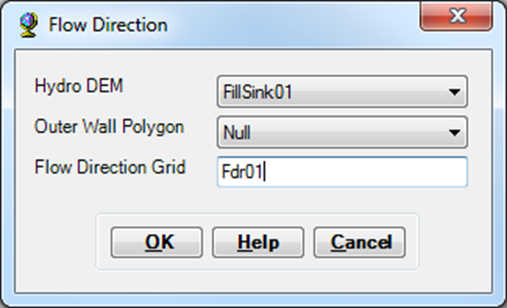
Deranged Polygon: Null

Hydro DEM: *FillSink01*

Use Sink Field: Unchecked

Fill Method: Fill All

#### Generate the flow direction grid

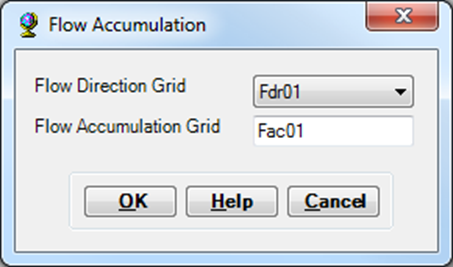
The *Flow Direction* tool (Arc Hydro Tools > Terrain Processing > Flow Direction) is run to create the layer that represents the direction of flow within each cell.

Hydro DEM: *FillSink01*

Outer Wall Polygon: Null

Flow Direction Grid: *Fdr01*

#### Generate the flow accumulation grid

**The *Flow Accumulation* tool (Arc Hydro Tools > Terrain Processing > Flow Accumulation) is run to create the layer that represents the total number of cells flowing into each cell.

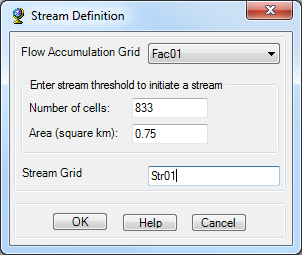
Flow Direction Grid: *Fdr01*

Flow Accumulation Grid: *Fac01*

### Streams Processing

The steps in this section force spatial consistency in the stream network used to delineate the catchments. This improved consistency is achieved by truncating the high resolution streams to a minimum drainage area threshold of 0.75 km2.

#### Define streams

The threshold stream grid is defined based on a minimum contributing drainage area. This step is completed by running the *Stream Definition* tool (Arc Hydro Tools > Terrain Processing > Stream Definition). These streams will be used to trim the high resolution streams and improve resolution consistency.

Flow Accumulation Grid: *Fac01*

Number of Cells: 833

Area (square km): 0.75

Stream Grid: *Str01*

#### Rasterize high resolution streams

The high resolution streams layer is converted to raster format using the *Polyline To Raster* tool (Conversion Tools > To Raster > Polyline To Raster). This layer will be trimmed to the defined drainage area threshold.

Input Features: *delineationStreams01*

Value Field: rasterVal

Output Raster Dataset: *StrHR01*

Call assignment type: MAXIMUM\_LENGTH

Priority Field: None

Cell Size: Same as "*Str01*"

Environment Settings > Processing Extent > Snap Raster: "*Str01*"

Output Location: NHDH01/arcHydro/Layers

#### **Create delineation streams**

The high resolution stream grid is trimmed to the threshold stream grid using the *Extract by Mask* tool (Spatial Analyst Tools > Extraction > Extract by mask).

Input raster: *Str01*

Input raster or feature mask data: *StrHR01*

Output Raster: *StrTrunc01*

Output Location: NHDH01/arcHydro/Layers

#### **Segment stream grid**

The delineation stream grid is segmented at confluences for network definition by running the *Stream Segmentation* tool (Arc Hydro Tools > Terrain Processing > Stream Segmentation).

Flow Direction Grid: *Fdr01*

Stream Grid: *StrTrunc01*

Sink Watershed Grid: Null

Sink Link Grid: Null

Stream Link Grid: *StrLnk01*

#### Create drainage lines

The flowlines used for delineation are generated using the *Drainage Line Processing* tool (Arc Hydro Tools > Terrain Processing > Drainage Line Processing).

Stream Link Grid: *StrLnk01*

Flow Direction Grid: *Fdr01*

Drainage Line: *DrainageLine01*

#### Edit network breaks

Errors in the raw high resolution flowlines can result in breaks in the network. Many of these breaks have been fixed in the DSL streams layer, so the edits should primarily fall in the area of the raw NHD flowlines. Any breaks in the delineation network will result in networks lacking an accurate outlet.

*Select By Attributes* is run on *DrainageLine01* where: NextDownID = -1 to isolate segments requiring visual inspection. These selected stream segments may be exported as a new layer to make inspection easier.

Any network breaks that are not sensible downstream terminus points such as coastal segments or region outlets are corrected in the *burnStreams01* and *delienationStreams01* layers. These layers are manually edited to either connect or remove the stream segments that cause breaks in the network. Coastal termini are checked for segments that should reach the *demOutline01* boundary but fall short. These segments get extended across the boundary to be recognized as a flow outlet. The "RasterVal" column is recalculated as 1 whenever new features are added to a streams layer.

When manual edits are made, all of the existing Arc Hydro outputs are deleted. The *Topography Processing* and *Streams Processing* sections are rerun using the corrected streams layers. If no manual edits are necessary, then processing moves on to the *Remove False Headwaters* section.

### Remove False Headwaters

The process of converting the high resolution streams to raster format and masking them with the threshold streams creates orthogonal cells which are sometimes interpreted by Arc Hydro as independent streams in the network. An example of this is shown in Figure 11. These streams do not actually exist in the high resolution dataset and are marked as false headwater streams to be removed from the network. Step 1 in this section identifies all false headwater segments. Steps 2 - 4 remove the false headwaters by editing polylines. Steps 5 - 8 remove remaining false headwaters that persist through previous steps by rasterizing the known false headwaters to mask these excess cells.

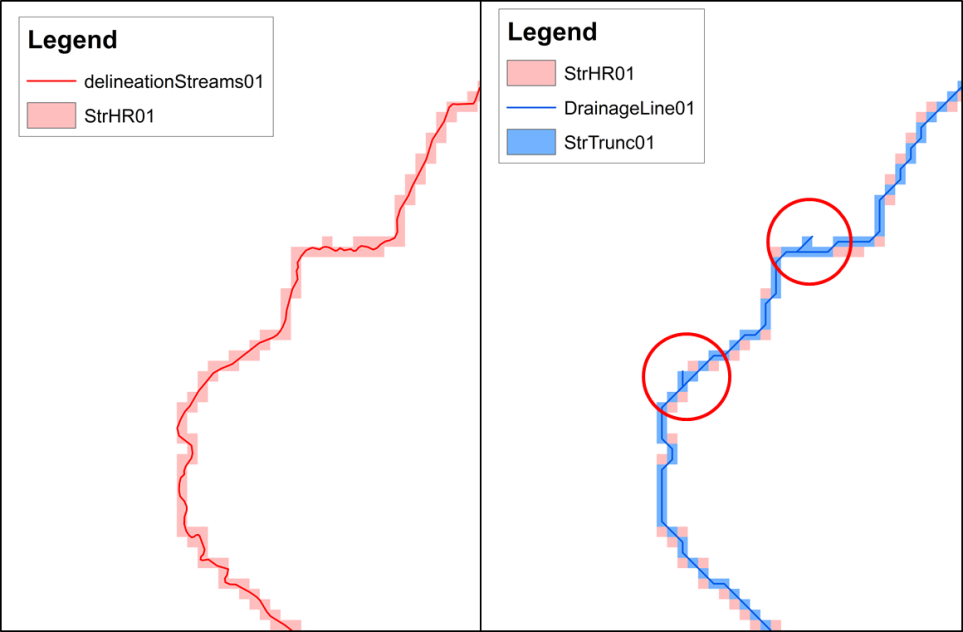


Figure : Origin of false headwater stream segments

#### Mark false headwaters

Headwater streams that are 90 meters or shorter are marked. This value is the maximum length of a segment that can still fit inside of 2 cells, a reasonable cutoff for these false headwaters. False headwater streams are identified by running the R script that checks segment length and whether or not it is classified as a headwater stream.

1. The *DrainageLine01* table is exported as *DrainageLine01.dbf* in the "NHDH01/arcHydro/Layers" directory
2. The “delineationProcessing - indexFalseHeadwaters.R” script is open in R and the following variables are set in the "Define Inputs" section:

*regionHUC2*

*run -*  The script is used at multiple points in the processing. This variable identifies the stage at which it is executed.

The variable “run” set equal to 1 and the script is run in R. A table (*DrainageLine01\_FalseHeadwaters.dbf*) identifying the false headwaters is output. The table is located in the same "NHDH01/arcHydro/Layers" directory.

1. The output table is joined back to *DrainageLine01* by the common "HydroID" field.

#### Mark true flowlines

A corrected layer of flowlines is created by exporting the features not marked as false headwaters.

The *DrainageLine01* attribute table is opened and the segments to keep are selected by setting the *Select by Attributes*: "DrainageLine01\_FalseHeadwaters.remove" = 0

The selected features are then exported as *DrainageLineTrimHW01* to the “NHDH01/arcHydro/vectors.gdb/Layers” directory.

#### Rasterize new flowlines

The edited flowlines are rasterized using the *Point to Raster* tool (Conversion Tools > To Raster > Polyline To Raster).

Input Features: *DrainageLineTrimHW01*

Value Field: rasterVal (= 1)

Output Raster Dataset: *StrTrimHW01*

Cell assignment type: MAXIMUM\_LENGTH

Priority Field: None

Cell Size: Same as "*StrTrunc01*"

Environment Settings > Processing Extent > Snap Raster: "*StrTrunc01*"

Output Location: NHDH01/arcHydro/Layers

#### Trim new flowlines

Rasterizing the new flowlines can create some extra cells that did not exist in the original layer (*StrTrunc01*). The new stream grid gets trimmed to the old one in order to remove these extra cells by using the *Extract by Mask* tool (Spatial Analyst Tools > Extraction > Extract by mask).

Input raster: *StrTrimHW01*

Input raster or feature mask data: *StrTrunc01*

Output Raster: *StrFixHW01*

Output Location: NHDH/arcHydro/Layers

#### Identify false headwaters

A polyline layer representing the false headwater streams is created by exporting the features marked for removal, similar to step 2.

The *DrainageLine01* attribute table is opened and the segments to be removed are selected by setting the *Select by Attributes*: "DrainageLine01\_FalseHeadwaters.remove" = 1

The selected features are then exported as *FalseHeadwaters01* to the “NHDH01/arcHydro/vectors.gdb/Layers” directory

#### Convert false headwaters to points

The false headwater lines are converted to points for rasterization using the *Feature to Point* tool (Data Management Tools > Features > Feature to Point). In most cases the points align with the correct cell they will represent. If they are not aligned, this is corrected for in the next iteration.

Input Features: *FalseHeadwaters01*

Output Feature Class: *FalseHeadwatersPts01*

Inside: Checked

Output Location: NHDH01/arcHydro/vectors.gdb/Layers

#### Convert false headwater points to raster

The false headwaters are rasterized using the *Point to Raster* tool (Conversion Tools > To Raster > Point To Raster). These points are used to correct the new stream grid.

Input Features: *FalseHeadwatersPts01*

Value Field: rasterVal (= 1)

Output Raster Dataset: *FalseHWPts01*

Call assignment type: MOST\_FREQUENT

Priority Field: None

Cell Size: Same as "*StrFixHW01*"

Environment Settings > Processing Extent > Snap Raster: "*StrFixHW01*"

Output Location: NHDH01/arcHydro/Layers

#### Remove remaining headwater cells

The remaining false headwater cells are removed from the new stream grid layer using the *Raster Calculator* (Spatial Analyst Tools > Map Algebra > Raster Calculator). The processing extent trim is set to equal that of the DEM to avoid spatial restriction issues in later steps of catchment delineation.

Map Algebra Expression: SetNull(~(IsNull("FalseHWPts01")),"StrFixHW01")

OutputRaster: *StrFinal01*

Environment Settings > Processing Extent > Extent: "*dem01*"

Environment Settings > Processing Extent > Snap Raster: "*StrFixHW01*"

Output Location: NHDH01/arcHydro/Layers

#### Segment stream grid

The delineation stream grid is segmented at confluences for network definition by running the *Stream Segmentation* tool (Arc Hydro Tools > Terrain Processing > Stream Segmentation).

Flow Direction Grid: *Fdr01*

Stream Grid: *StrFinal01*

Sink Watershed Grid: Null

Sink Link Grid: Null

Stream Link Grid: *StrLnkFinal01*

#### Create new drainage lines

The flowlines used for delineation are generated by running the *Drainage Line Processing* tool (Arc Hydro Tools > Terrain Processing > Drainage Line Processing).

Stream Link Grid: *StrLnkFinal01*

Flow Direction Grid: *Fdr01*

Drainage Line: *DrainageLineFinal01*

#### Manually check flowlines

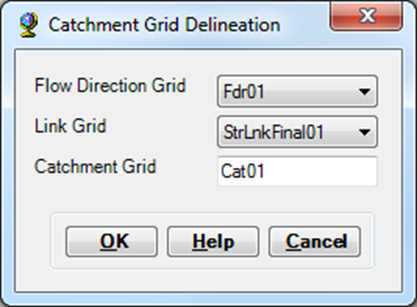
The "delineationProcessing - indexFalseHeadwaters.R" script is rerun to check that all of the false headwaters have been removed. This time the “run” variable is set equal to 2. The resulting dataframe "x" should be empty. Any rows in this dataframe identify remaining false headwaters. Additionally, step 8 has the potential to create network breaks. The resulting breaks are indexed by following *Select by Attributes* method in Step 6 in the *Streams Processing* section. If any false headwaters or breaks are found during either of these checks, the points are manually adjusted in the *FalseHeadwatersPts01* layer to align with the correct cell to be masked. This is done by placing the points to either eliminate the false headwaters or to prevent network breaks. If new points are added the “RasterVal” column is recalculated to equal 1.

If any corrections are made, delete: *FalseHWPts01*, *StrFinal01*, *StrLnkFinal01*, *DrainageLineFinal01*, *DrainageLineFinal01.dbf* and steps 7-10 are rerun. If no corrections are necessary, processing moves on to the next section.

### Delineate Catchments

This section simply uses the hydrography layers and the corrected stream network to generate catchments. The catchments generated in this section fall into the “Delineation” category discussed in the *Product Description* section.

#### Delineate catchment grid

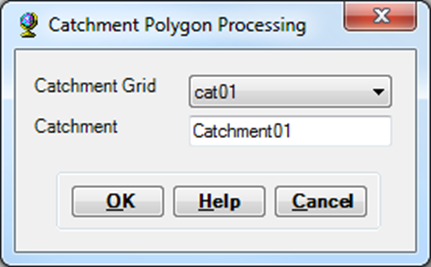
**Catchments are delineated based on the corrected drainage line layer and the flow direction grid using the *Catchment Grid Delineation* tool (Arc Hydro Tools > Terrain Processing > Catchment Grid Delineation).

Flow Direction Grid: *Fdr01*

Link Grid: *StrLnkFinal01*

Catchment Grid: *Cat01*

#### Convert catchment grid to polygons

**The catchment raster is converted to polygons using the *Catchment Polygon Processing* tool (Arc Hydro Tools > Terrain Processing > Catchment Polygon Processing).

Catchment Grid: *Cat01*

Catchment: *Catchment01*

### Coastal Catchment Delineation

Some of the coastal areas cannot be initially processed because they do not have high resolution streams present or do not meet the minimum drainage area threshold. A separate delineation process is necessary for this case. Within the area of missing catchments, the threshold stream grid (*Str01*) created in step 1 of the *Streams Processing* section is used to delineate coastal catchments in the same fashion as the previous section. After this is complete, filler catchments are created in any areas still missing coverage.

If the hydrologic region is landlocked, the processing skips to the *Finalize Hydrography Layers* section.

#### Identify the coastal zone

The existing catchments are removed from the processing range using the *Erase* tool (System Toolboxes > Analysis Tools > Overlay > Erase), producing a polygon of the area missing catchments.

Input Features: *demOutline01*

Erase Features: *Catchment01*

Output Feature Class: *CoastalErase01*

Output Location: NHDH01/arcHydro/vectors.gdb/Layers

#### Convert from multipart to singlepart

For editing purposes, the costal zone polygon is converted to single-part format using the *Multipart To Singlepart* tool (Data Management Tools > Features > Multipart To Singlepart).

Input Features: *CoastalErase01*

Output Feature Class: *CoastalZone01*

Output Location: NHDH01/arcHydro/vectors.gdb/Layers

#### Manually edit coastal zone

Inland polygons that clearly do not represent coastal areas are deleted manually. The final shape along the coastal region is edited to ensure it falls within the *terrestrialBoundary01* layer.

Output Feature Class: *CoastalZone01*

Output Location: NHDH01/arcHydro/vectors.gdb/Layers

#### Trim the stream grid

The threshold stream grid is trimmed to the coastal zone using the *Extract by Mask* tool (Spatial Analyst Tools > Extraction > Extract by Mask).

Input raster: *Str01*

Input raster or feature mask data: *CoastalZone01*

Output Raster: *StrCoast01*

Environment Settings > Processing Extent > Snap Raster: "*Str01*"

Output Location: NHDH01/arcHydro/Layers

#### Trim the flow direction

The flow direction grid is trimmed to the coastal zone using the *Extract by Mask* tool (Spatial Analyst Tools > Extraction > Extract by Mask).

Input raster: *Fdr01*

Input raster or feature mask data: *CoastalZone01*

Output Raster: *FdrCoast01*

Environment Settings > Processing Extent > Snap Raster: "*Fdr01*"

Output Location: NHDH01/arcHydro/Layers

#### Segment stream grid

The threshold stream grid is segmented at confluences for network definition using the *Stream Segmentation* tool (Arc Hydro Tools > Terrain Processing > Stream Segmentation).

Flow Direction Grid: *FdrCoast01*

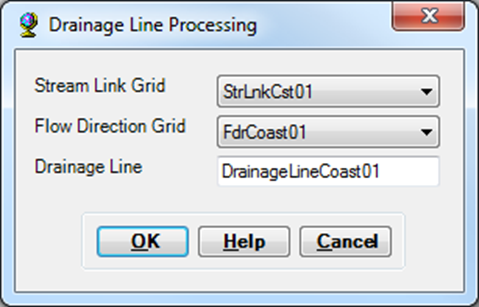
Stream Grid: *StrCoast01*

Sink Watershed Grid: Null

Sink Link Grid: Null

Stream Link Grid: *StrLnkCst01*

#### Create New Drainage Lines

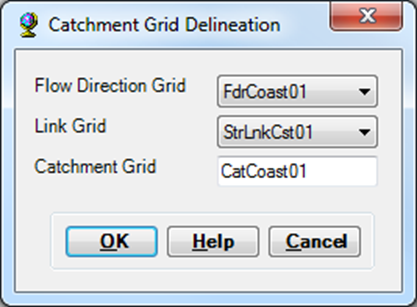
**Drainage lines are generated from the coastal segmented streams using the *Drainage Line Processing* tool (Arc Hydro Tools > Terrain Processing > Drainage Line Processing).

Stream Link Grid: *StrLnkCst01*

Flow Direction Grid: *FdrCoast01*

Drainage Line: *DrainageLineCoast01*

#### Delineate Catchment Grid

Coastal catchments are delineated based on the coastal drainage lines and flow direction grid using the *Catchment Grid Delineation* tool (Arc Hydro Tools > Terrain Processing > Catchment Grid Delineation).

Flow Direction Grid: *FdrCoast01*

Link Grid: *StrLnkCst01*

Catchment Grid: *CatCoast01*

#### Convert Catchment Grid to Polygons

The coastal catchments raster is converted to a polygon layer using the *Catchment Polygon Processing* tool (Arc Hydro Tools > Terrain Processing > Catchment Polygon Processing). The result of this step is the “Coastal” classification of catchments discussed in the *Product Description* section.

Catchment Grid: *CatCoast01*

Catchment: *CatchmentCoast01*

1. Generate the area still missing coverage

The *Erase* tool (System Toolboxes > Analysis Tools > Overlay > Erase) is run again to define the coastal areas within the processing boundary that are still missing catchments.

Input Features: *CoastalZone01*

Erase Features: *CatchmentCoast01*

Output Feature Class: *CoastalFill01*

Output Location: NHDH01/arcHydro/vectors.gdb/Layers

#### Split apart the polygons

The missing area polygons are split into individual polygons using the *Multipart to Singlepart* tool (Data Management Tools > Features > Multipart To Singlepart).

Input Features: *CoastalFill01*

Output Feature Class: *CoastFillSingle01*

Output Location: NHDH01/arcHydro/vectors.gdb/Layers

#### Remove tiny features

Catchments that are too small to be functional are removed from the catchment fill layer to prevent an overwhelming number of very small catchments. Based on comparable size and the number of catchments eliminated, an area of 10,000 m2 is chosen as the minimum catchment area. All features that meet the minimum area threshold are selected using the *Select by Attributes* tool. The selection is exported to a new layer. The result of this step is the “Coastal Fill” catchments described in the *Product Description* section.

Select the features where: Shape\_Area >= 10000

Export the selected features as: *CatchmentFill01*

Output Location: arcHydro/vectors.gdb/Layers

### Finalize Hydrography Layers

This section combines the primary and coastal layers into the full coverage output layers.

#### Merge hydrography layers

The primary catchment and flowline layers are merged with the coastal layers using the “delineationProcessing - joinHydrographyLayers.py” script. This script also edits the catchment and flowline identifiers (FEATUREID) to reflect the version and hydrologic region. The variables in the “Define Inputs” section of the script are set:

*baseDirectory* & *regionHUC2*

*version* – The delineation version, in this case “2”. This is used in the renaming of the catchment and flowline identifiers

The script is run in Arc Python.

#### Check for Errors

Once the hydrography layers have been merged a brief quality check is completed before beginning the post processing. An R script performs this check for errors such as duplicate FEATUREIDs, flowlines without paired catchments, or NextDownIDs that do not exist as catchments. The variables in the “Define Inputs” section of the “postProcessing – checkNetworkErrors.R” script are set:

*baseDirectory*

*hydroRegions* – This can be either one element or a list of hydrologic region identifiers (e.g. “01”).

*run* – This indicates the step when the script is being applied.

In this step, only the current hydrologic region is listed as the *hydroRegions* variable. The *run* variable is set to 1 to indicate the first check for errors after the *Delineation Processing* section. If the results show no errors then the *Post-Delineation Processing* section is started.

## Post-Delineation Processing

Post-delineation processing ties up any loose ends and polishes the final layers. This section is only begun once the *Delineation Processing* section has been completed for all of the hydrologic regions in the product version. Gaps and overlapping layers are the two main issues addressed in this section. Gaps within a catchment layer can occur in the coastal areas, where tiny catchments have been removed in step 12 of the *Coastal Catchment Delineation* section. Gaps can also occur along hydrologic region boundaries where minor differences in pre-processing edits between regions may have resulted in a few unassigned cells. These minor processing differences may also cause overlap between catchment layers in different hydrologic regions. This stage in processing is broken into separate boundary stretches. These stretches are defined as either hydrologic region boundaries (e.g. between 01 and 02) or coastal sections for a single region. All of the editing steps described here focus on one stretch of boundary. Only when all of the corrections in one stretch are completed is the editing of the next stretch initiated.

### Prepare Layers for Editing

This section prepares the layers used in the manual editing process. These steps are taken to preserve the original layers for backup.

#### Export hydrography layers

The hydrography layers are exported from the individual hydrologic region folders to the editing geodatabase. For example, *allCatchments01.shp* is exported as *correctedCats01* in the “gisFiles/postProcessing/Edits.gdb” directory. The *allFlowlines01.shp* layer is exported as *correctedFlowlines01* in the same directory. All manual edits are performed on these exported layers in the *Edits.gdb* directory.

#### Clip adjacent layers

The *Clip* tool (Analysis Tools > Extract > Clip) is run on each pair of adjacent hydrologic region catchment layers. This action produces a polygon layer of the overlapping areas assigned to multiple catchments.

Input Features: *allCatchments02.shp*

Clip Features: *allCatchments01.shp*

Output Feature Class: *clip0102*

Output Location: gisFiles/postProcessing/Edits.gdb

#### Dissolve Catchments

The catchment polygons are dissolved using the *Dissolve* tool (Data Management Tools > Generalization > Dissolve).

Input Features: *allCatchments01*

Output Feature Class: *rawOutline01*

Create multipart features: Checked

Output Location: gisFiles/postProcessing/Edits.gdb

### Correct Overlapping Sections

Overlapping catchments are corrected by inspecting each of the features in the clipped layers (e.g. *clip0102*). The adjacent catchment layers are manually edited to correct the overlapping sections, which are typically only the size of one or two raster cells.

### Correct gaps within each catchment layer

The catchment outline (*rawOutline01*) is visually inspected along the coastal areas for gaps that may exist in the layer. These are more easily spotted by setting the background color to contrast the outline layer color. When found, these gaps are manually closed by editing the catchment polygon layer. Care is taken to account for the fact that some gaps may be a part of the coastline, particularly near river mouths. These cases are verified by comparison with the raw high resolution streams.

### Correct gaps between catchment layers

The regional border area between adjacent catchment layers is visually examined for gaps. Any gaps are closed by manual editing. The background color is again set to contrast the catchment layer color to make spotting of the gaps easier.

### Manual Catchment Adjustments

Further catchment editing was necessary in two regions.

#### Region 03

The southeastern area of region 03 is comprised mostly of wetlands. In this area the flowlines are mostly gridded, making the delineation prone to error in flow direction. This section was out of the essential delineation range (DSL project range) and was removed. Removal was completed such that no hydrologic networks were disrupted.

#### Region 04

The northernmost sections of this region are located in Canada and do not have flowline information across the hydrologic boundary. This lack of stream data creates an inability to define the hydrologic boundary through delineation, resulting in a handful of very large and inaccurate catchments. These catchments were manually edited to fall within the existing hydrologic boundary established by the NHD. As with the edits to region 03, the catchments exist outside of the essential delineation range.

### Correct overlapping sections within catchment layers

#### Check overlapping polygons within a layer

It is possible that manual editing mistakes created areas of slightly overlapping polygons. This is likely from instances where polygon nodes were snapped incorrectly. The Intersect tool (Analysis Tools > Overlay > Intersect) is run on each layer alone to identify these instances. An empty output indicates that the layer is correct. Any resulting features identify the overlapping areas to be fixed.

Input Features: *correctedCats01*

Output Feature Class: *intersect01*

Output Location: gisFiles/postProcessing/Edits.gdb

#### Check overlapping polygons between layers

To make sure all corrections were made on the overlapping layers, rerun the *Clip* tool on all adjacent catchment layers as done in the *Prepare Layers for Editing* section. If the output is empty the layer is correct.

Input Features: *correctedCats01*

Clip Features: *correctedCats02*

Output Feature Class: *clipOverlap0102*

Output Location: gisFiles/postProcessing/Edits.gdb

### Export Hydrography Products

#### Export layers

The corrected layers are exported in both shapefile and geodatabase format. Basic metadata like catchment area and stream segment length are calculated for these final layers. The flowlines undergo a smoothing process to make them visually more realistic than the raster-based vector layer output from the Arc Hydro tools. These actions are completed by running the “postProcessing – finalizeProducts.py” script. Open the script and set the variables in the “Define Inputs” section:

*baseDriectory*

*hydroRegions* – A list of all of the hydrologic region identifiers. This is “01” through “06” for version 2.

#### Check for errors

The “postProcessing – checkNetworkErrors.R” script is again used for a basic quality check. The script is open. This time all of the *hydroRegions* are listed and *run* is set equal to 2. If the results show no errors then the final layers are complete.

### Generate Detailed Flowlines

The flowlines output from the delineation process have been truncated to adhere to the minimum initiation drainage area requirement of 0.75 km2. In some cases, the full high resolution version of the flowlines may be necessary for use. This step performs a simple mapping of the raw high resolution lines to the catchments. The variables in the “Define Inputs” section of the “postProcessing – detailedFlowlines.py” script are set:

*baseDirectory* & *hydroRegions*

All of the hydrologic regions in the delineation range are listed in the *hydroRegions* variable.

The flowline mapping is accomplished using a simple overlay method. Any portion of a flowline that falls inside of a catchment gets assigned to that catchment. This method has a few issues which are discussed in the *Caveats* section. While other methods exist, this one is chosen because of its simplicity and relatively manageable processing requirements.

### Generate Riparian Buffers

Riparian buffer layers are generated for the detailed flowlines and mapped to the catchments by the same overlay methodology used in the *Generate Detailed Flowlines* section. These layers are created by the execution of an Arc Python script. The variables in the “Define Inputs” section of the “postProcessing – createRiparianBuffersTiled.py” script are set:

*baseDirectory* & *hydroRegions*

*bufferDistancesFT* – A vector of character strings listing the riparian buffer distances (in feet) to be generated from the script.

The script is run in Arc Python. The original process uses a large amount of memory and is prone to crashing. Because of this problem, the processing is tiled into sections and merged at the end.

# Caveats

## Detached Streams

Prior to delineation, a number of raw NHD high resolution streams are manually edited to be joined to the network (see: *Pre-Delineation Processing* section). This is a blanket approach to prevent catchments that have no downstream destination within the watershed (these would essentially be sinks). These detached stream segments, mostly smaller headwaters, are small impoundments or possibly naturally intermittent streams. The streams are small enough that these impoundments serve as a termination point of the stream with the water likely being diverted for use. The stream then picks up further down gradient. The delineation process does not account for the diversions or natural intermittence and the segmented streams instead become part of the main network.

## Forced Region Boundaries

In some cases, the raw stream networks are joined across the established boundaries between hydrologic regions. These rare cases are either canals or small headwater streams, likely connected through a pond or diversion. All cases of cross-boundary streams are split apart by manual editing before delineation (see: *Pre-Delineation Processing* section). These edits are necessary to prevent the crossing from becoming a downstream point in the network, creating an incorrect flow direction in the delineation.

## Divergences

Divergences do not exist in the final delineated network. Stream segments are defined based on flow direction and accumulation. This translates to a forcing of flow into single channels. In this way, divergences or braided channels are forced into a single channel based on the DEM layer. These features do exist in the detailed flowlines layers, but are not specified in the network structure.

## Detailed Streams Overlay Method

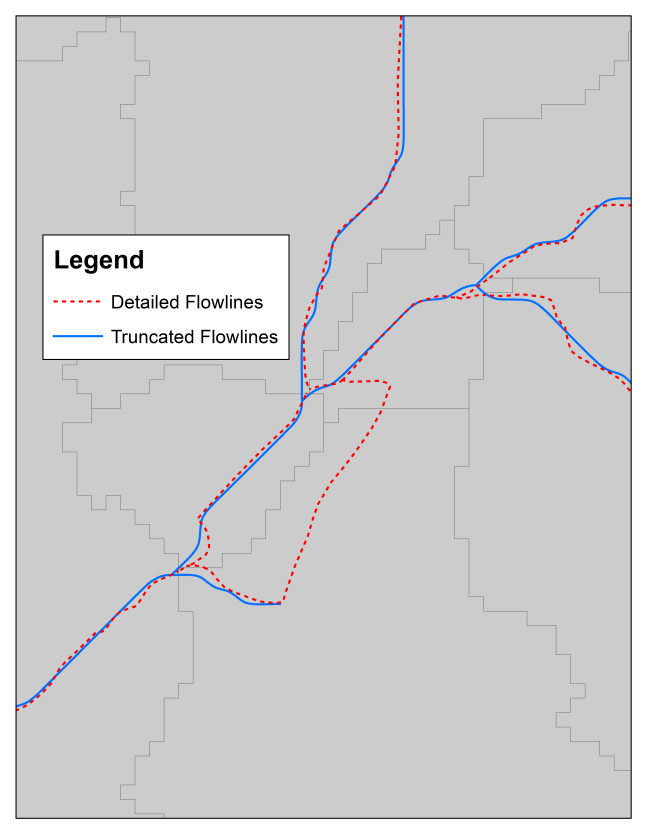


Figure : Detailed flowlines overlay method

The overlay method used to link the high resolution streams to the catchments essentially uses the catchments layer as a cookie cutter. This process creates some caveats because the catchments are based off of the simplified flowlines. Intra-catchment network structure between the high resolution stream segments is not accounted for. This omission occasionally produces some inaccurate intersections. For example, when the detailed streams in (Figure 12) are assigned to catchments, the flow path in the braid is broken since the network has been established according to the truncated flowlines. The two ends of the lone dotted red line fall into their respective catchments which are not directly linked. The next section discusses the further implications of this method that arise when creating the riparian buffer layers.

## Riparian Buffer Creation

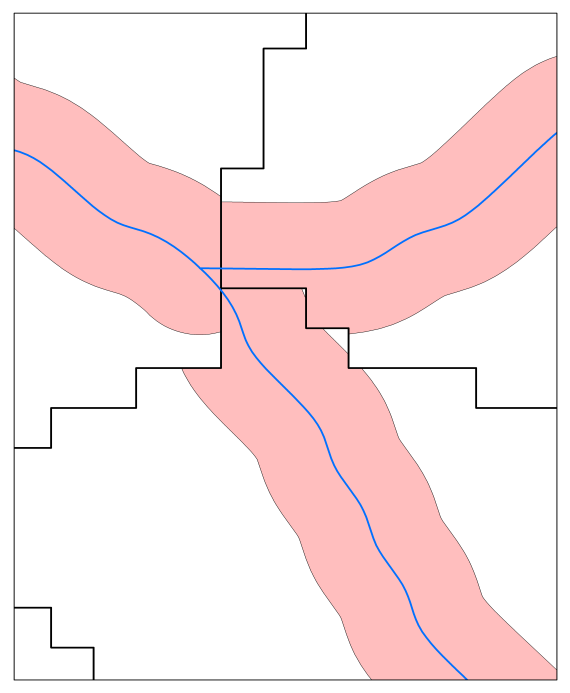


Figure : Riparian buffer misalignment

The riparian buffer layers based on the detailed flowlines may experience some misalignment due to resolution differences. The riparian buffers are directly generated from the vector flowlines, while the catchments are converted from raster format of 30m cell size. This unavoidable resolution difference is evident in Figure 13. While functionally this is the most accurate product, a separate riparian buffer layer may be more desirable for visualization purposes.

Another caveat related to the riparian buffer layers is similar to that described in the previous section. The overlay method used to map the detailed layers to the catchments does not account for network structure. Some riparian areas may not contribute to the downstream point of the catchment in which they fall. This is the case in the bolded catchment in Figure 14. The riparian buffer around the segment represented solely by the detailed flowlines is counted as part of the contributing area to the downstream terminus of the catchment. The reality though, is that this section contributes to the neighboring catchment. Though these situations are not frequent, they do exist throughout the dataset and should be acknowledged when using the detailed riparian buffer layers. In the case of basin characteristics, the local statistics might be off slightly in these situations, but the aggregated upstream statistics will still be accurate for catchments downstream of the confluence between the two separate catchments.

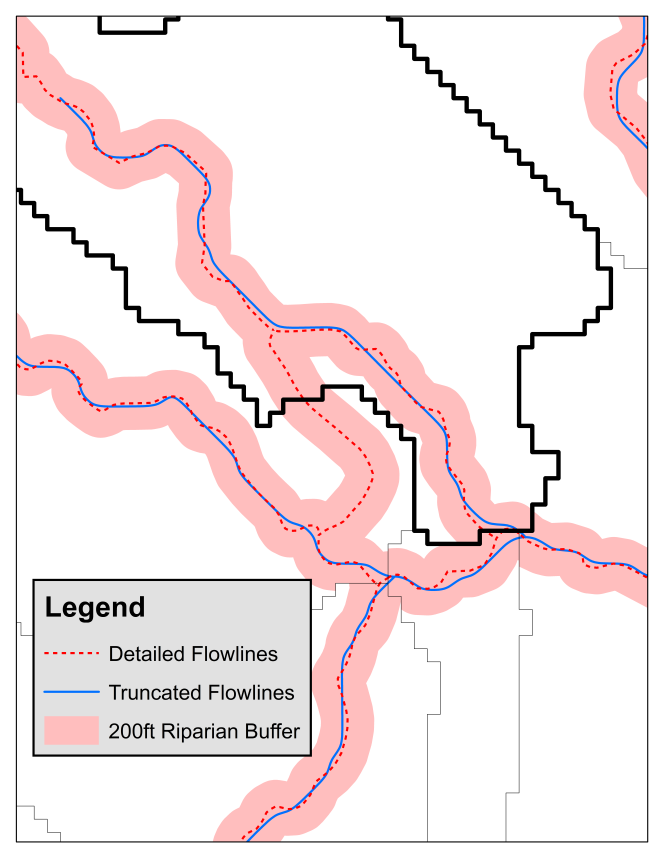


Figure : Detailed riparian buffer at a braid

## Small Tributary Omission

Forcing areal consistency among catchments results in the omission of some of the high resolution streams. Though burning the high resolution streams into the DEM incorporates the entire dataset into the delineation, it does not ensure that each stream segment is paired exclusively with a catchment. Catchments are only defined for each stream segment that meets the minimum drainage area threshold used for stream definition. These streams are defined as the “truncated flowlines” in the *Product Description* section.  In many cases, multiple small headwater streams below the threshold share the same catchment. Figure 15 shows a headwater region of a watershed where streams sharing a catchment are likely to have similar attributes such as size, discharge, and stream order. Representing this area with one catchment is considered acceptable given the minor hydrologic differences between the streams. In some cases small tributary streams, not large enough to warrant their own catchments, are incorporated into the catchment of an adjacent segment of a large river (Figure 16). In these cases the known streams within the same catchment are far more likely to have significantly varying attributes. These differences bring into question the assignment of catchment attributes for the larger river segments to the areas feeding the small, omitted tributaries. The result is an inaccurate assignment of spatial attributes or modeling results to the tributaries.

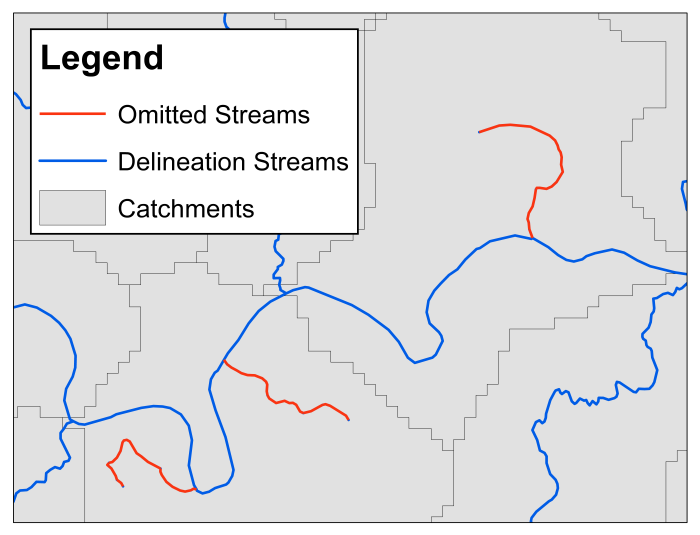


Figure 16: Small tributaries grouped with a large river within a single catchment

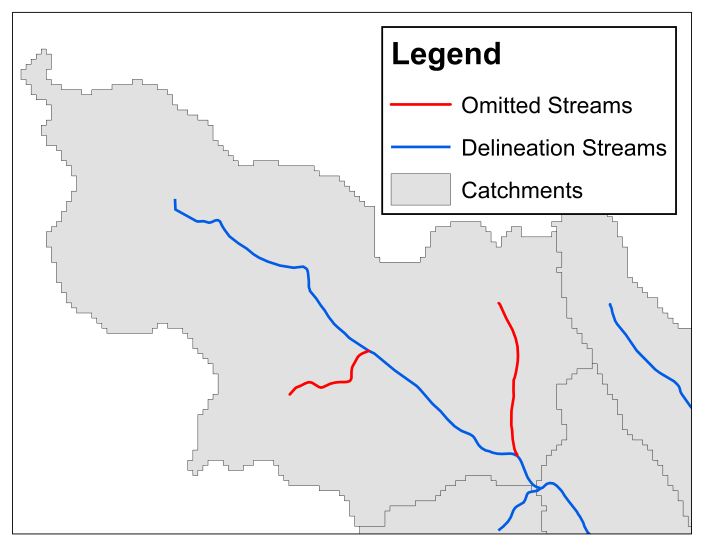


Figure : Headwater streams grouped within a single catchment

Alternative options are explored to address the omission of small headwaters contributing directly to large rivers. The primary alternative, a variation on the catchment delineation of large rivers, aims to split the original catchment into three new catchments in order to separate small tributaries from the larger rivers they contribute to directly. For all catchments with a contributing drainage area of 200 km2 or greater, one main channel catchment represents the footprint of the stream raster and the other two lateral catchments represent the drainage areas contributing flow directly to the main channel reach (Figure 17). These catchments are created through the spatial process of splitting the original catchment with the stream segment as opposed to running an actual delineation of the smaller tributaries. Explicit delineation of the small tributaries is also explored.

Though capturing the small tributaries directly contributing flow to large rivers maintains some benefit, there are significant concerns with the implementation of this update. Distinguishing between relatively homogenous streams in headwater regions versus small tributaries on large rivers requiring unique delineation proves to be an arbitrary process without an in-depth spatial analysis. Once identified, altering these catchments to not adhere to the minimum drainage area threshold for stream definition would violate one of the core features of the delineation product, areal consistency. Additionally, the method of assigning catchments solely based on position in an existing catchment rather than contribution to a known stream reach creates a large number of new lateral catchments that would not actually contain known streams. Violation of the areal consistency of the product and complexities from the terrestrial representation of aquatic systems prevent the implementation of this version update. Ultimately, the omission of the highest resolution streams is accepted as necessary in order to maintain spatially consistent catchments.

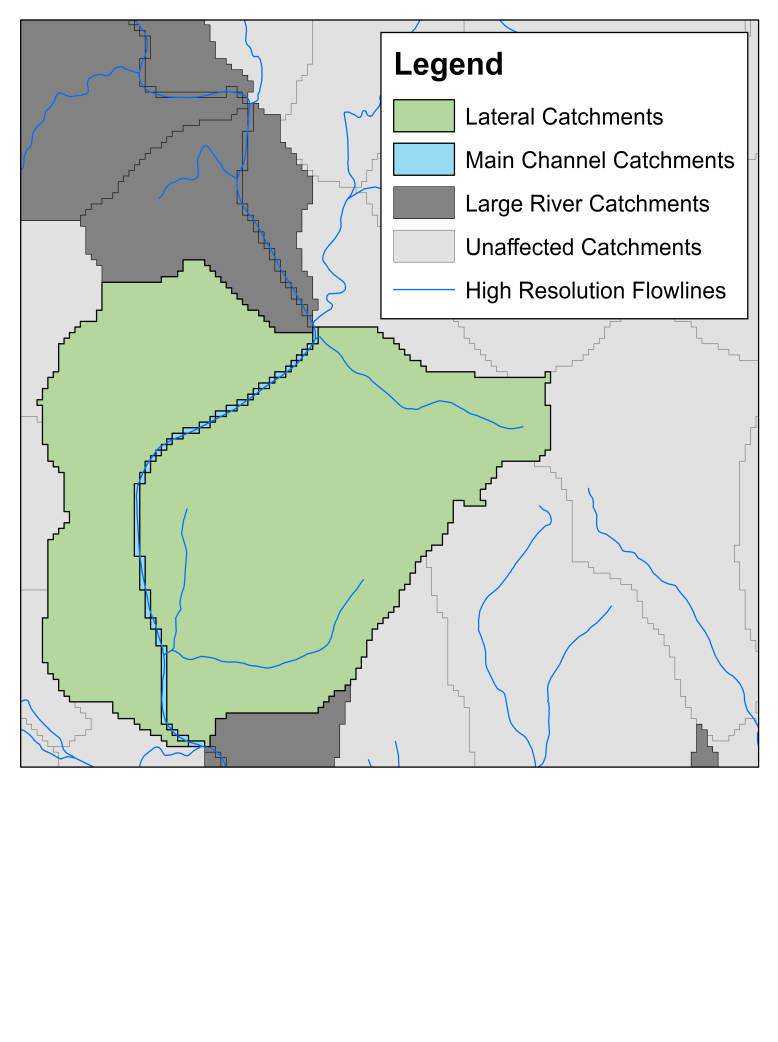


Figure 17: Example of adjusted catchment

The proposed update, termed Version 2.1 of the delineation, which includes the new main channel and lateral catchments for large rivers, will remain unreleased. However, the new layers exist if they are desired for analysis that accepts the caveats described above. Future work on explicit delineation of omitted streams is possible, but would require significant resources and likely need to be considered as a separate product.

# References

Canadian Provinces. “National Weather Service OST/SEC GIS Map Group”. <http://www.nws.noaa.gov/geodata/catalog/national/html/province.htm>. (May 26, 2015)

Designing Sustainable Landscapes. “Landscape Ecology Lab, University of Massachusetts Amherst” <http://www.umass.edu/landeco/research/dsl/dsl.html>

National Elevation Dataset. “USGS – The National Map Viewer and Download Platform”. <http://viewer.nationalmap.gov/viewer/>. (May 26, 2015)

National Hydrography Dataset - High Resolution. “USGS – The National Map Get NHD Data”. <ftp://nhdftp.usgs.gov/DataSets/Staged/SubRegions/FileGDB/HighResolution/>. (May 2015)

Provincial Limits (New Brunswick). “GeoNB”.

<<http://www.snb.ca/geonb1/e/DC/catalogue-E.asp>>. (May 26, 2015).

State Boundaires, One Million-Scale. “USGS - The National Map Small-Scale Collection”. <http://nationalmap.gov/small\_scale/atlasftp.html?openChapters=chpbound#chpbound>. (June 2014)

1. This dataset was formerly named the NHDHRDV2. To avoid confusion with the USGS NHDPlus High Resolution (HR) dataset, it was renamed in 2022. [↑](#footnote-ref-1)
2. Note that the NECD dataset was created in 2015, before the NHDPlus HR dataset had been completed. [↑](#footnote-ref-2)