**NHDPlusHR VAA Navigator**

**User Guide**

Contents

[NHDPlusHR VAA Navigator Description 1](#_Toc8732532)

[NHDPlusHR VAA Navigator Databases 1](#_Toc8732533)

[Using the NHDPlusHR VAA Navigator ArcMap Toolbar 2](#_Toc8732534)

[Using the NHDPlusHR VAA Navigator from a User-written Program 7](#_Toc8732535)

# NHDPlusHR VAA Navigator Description

The NHDPlusHR VAA Navigator performs navigation on the NHDPlusHR surface water network using NHDPlusHR Value Added Attributes found in the NHDPlusFlowlineVAA table. The Navigator may be used through ArcMap via a special ArcMap toolbar or it may be called from user-written program code. The Navigator will work on any NHDPlusHR gdb. The gdb may contain a single Vector Processing Unit (VPU) or several VPUs that have been appended together.

The Navigator performs four types of navigation: upstream mainstem, upstream with tributaries, downstream mainstem, and downstream with divergences. Navigations can begin and end on full NHDFlowline features or may begin and end as points along features.

Any of the four types of navigation may be stopped based on a user-supplied distance from the starting point. Navigation results may also be filtered based on certain user specified criteria.

# NHDPlusHR VAA Navigator Databases

From the ArcMap toolbar, the first time the VAA Navigator is executed for a given NHDPlusHR gdb, it builds a Microsoft SQL database for that gdb which speeds up future navigations. The database is called V03NavDB\_<nnnnnnnn>.mdf, where <nnnnnnnn> is the name of the NHDPlusHR gdb being navigated. The database is created in a local folder specified by the user. The second and subsequent times that the Navigator is executed, it looks for the database and, if found, uses it.

Recommendation: In the navigation options dialog displayed when the VAA Navigator Toolbar is used, specify that the Navigator databases are to be stored with your installed NHDPlusHR data in a sub-folder that you create for that purpose. You must have read/write access to this folder.

# Using the NHDPlusHR VAA Navigator ArcMap Toolbar

1. **Description:**



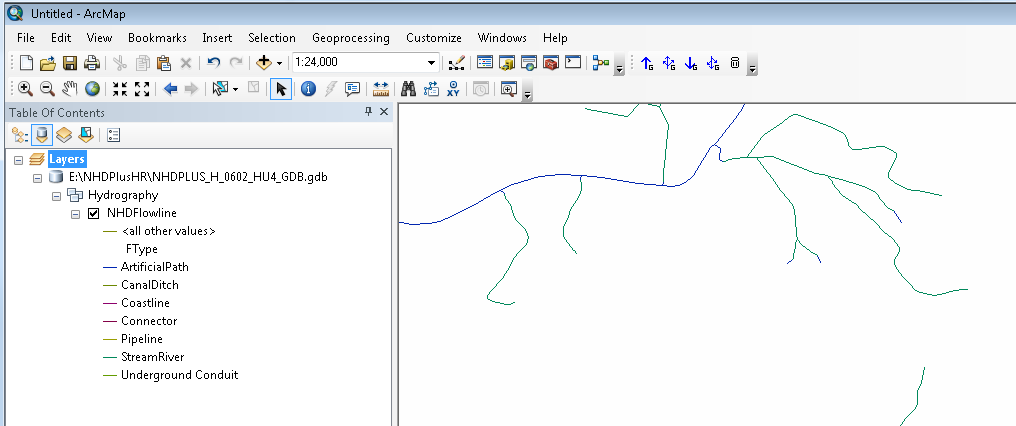
The NHDPlusHR VAA Nav Toolbar is an ArcMap Toolbar with 5 tools representing the 4 primary navigation types and a trashcan tool which allows the user to delete the MS SQL databases created by the navigator. When one of the navigation tools is selected, a form is displayed that allows the user to identify stop conditions and navigation results filtering rules.

The toolbar can be added to ArcMap using the Toolbar Customize function.

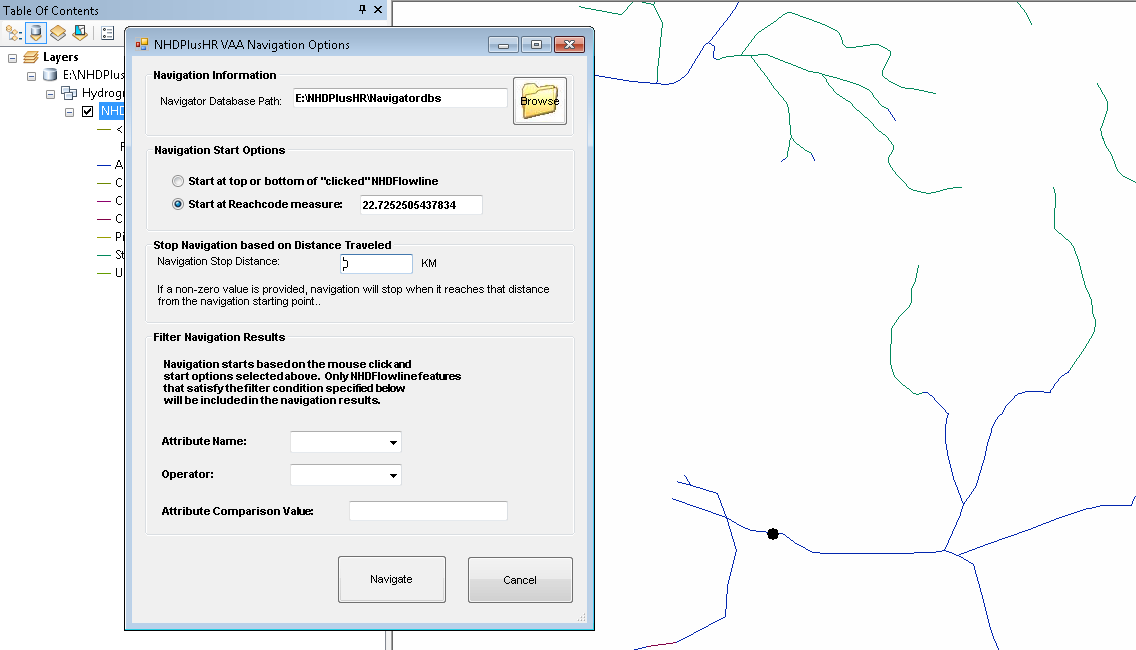
Each navigation involves obtaining information from the user, building a navigation database, if one does not exist, clearing any previous navigation results from the map, calling the NHDPlusHRNavigator, saving the results as an event table in the SQL database, and, finally, rendering the results in the map document. Each navigation for a given NHDPlusHR gdb is stored in the SQL database with a unique name and remains in the database until the user deletes the database.

The NHDPlusHR VAA Nav Toolbar obtains navigation information from the user based on the tool selection, the mouse click on a starting NHDPlusID/measure, and the NHDPlusHR VAA Navigation Options form.

1. **General Usage:**
2. Within ArcMap, manually load a \Hydrography\NHDFlowline feature class from the NHDPlusHR gdb.

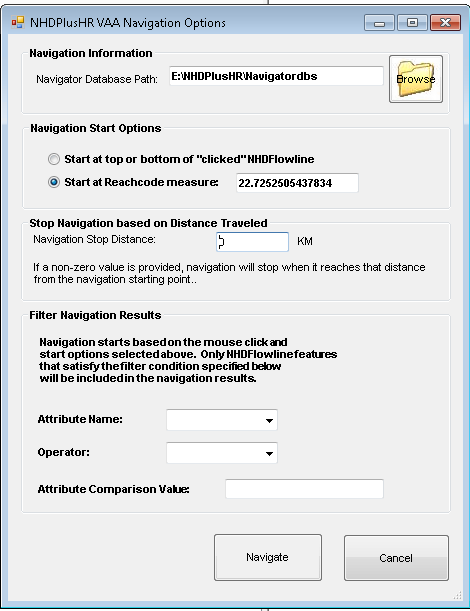


1. Select the desired tool from the NHDPlusHR VAA Navigator Toolbar. (Upstream Mainstem, Upstream with Tributaries, Downstream Mainstem, or Downstream with Divergences)
2. Make sure that the NHDFlowline layer is the active layer and click on the desired starting NHDFlowline feature. Zoom in if necessary.



In the above picture, the black dot has been placed at the location of the mouse click and the Navigation Options dialog is open.

1. The “Navigator Database Path” entry must point to a path where you have read/write privileges. This is the location where the navigator will place the MS SQL database discussed above. This database is created the first time you navigate a specific NHDPlusHR gdb and then reused each time you perform a navigation in the same gdb. The first navigation in a given gdb will take a longer in order for the database to be built. The database is named V03NAVDB\_<NHDPlusHR gdb name>. Once you have set this path in the dialog, it should appear in the dialog during subsequent navigations.
2. Establish start, stop and filtering conditions if necessary using the Navigation Options dialog.



NHDPlusHR VAA Navigation Options Dialog

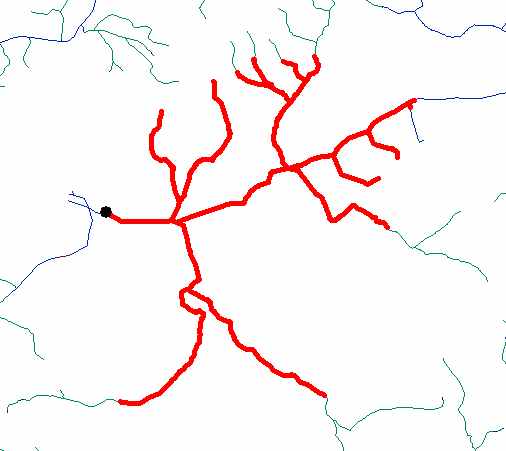
1. Navigation Start Options.

Choose “Start at the top or bottom of the ‘clicked’ NHDFlowline” to include the whole starting NHDFlowline feature in the navigation results. Including the whole starting NHDFlowline feature means that the navigation will begin at the “from” measure (i.e. bottom) of the NHDFlowline feature for upstream navigations and at the “to” measure (i.e. top) of the NHDFlowline feature for downstream navigations.

Choose “Start at Reachcode measure” and provide a measure value to start somewhere else along the NHDFlowline feature. The default value is the measure at the mouse click.

1. Stop navigation based on a stop distance.

Supply a stopping distance other than 0 to stop the navigation when it has traveled the specified distance. 0 indicated that there is no stop distance and the navigation proceeds to the end of the network. The picture below shows an “Upstream with tribs” navigation was stopped after navigating 5 km.



1. Filter navigation results based on a specified value of a selected NHDPlus attribute.

**Select an Attribute Name.**

The possible NHDPlus attribute names are continuous numeric fields shown in the table below. See the NHDPlusHR User Guide for additional information about these NHDPlus attributes.

|  |  |  |
| --- | --- | --- |
| PathLength | Distance to the terminal NHDFlowline feature downstream along the mainpath | Continuous Numeric(13,4) |
| ArbolateSum | Kilometers of stream upstream of the bottom of the NHDFlowline feature | Continuous Numeric(13,4) |
| TotDASqKm | Total Upstream Cumulative Drainage Area at the downstream end of the NHDFlowline feature | Continuous Numeric(14,6) |
| DivDASqKm | Divergence-routed Cumulative Drainage Area at the downstream end of the NHDFlowline feature | Continuous Numeric(14,6) |

**Select an operator**

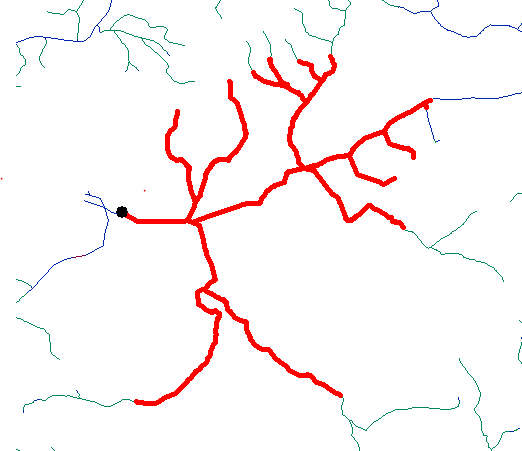
< Less than

<= Less than or equal to

> Greater than

>= Greater than or equal to

The picture below shows an upstream with tribs navigation that was filtered for NHDFlowline features that had TotDASqKM >= 5 sqkm.



1. Press the “Navigate” button on the Navigation Options dialog to start the navigation. Depending on the options and the size of the navigation, the navigator many take a few moments. The Navigator sets the cursor to a “busy” style, however, there are several parts of the Navigation when ArcMap takes control of the cursor and it may not be displayed as a “busy” style. To ensure that the navigation has enough time to complete, please wait for (1) the “Navigation Results” layer (in red) to display or (2) for a message box to appear that indicates there was an error during navigation.

# Using the NHDPlusHR VAA Navigator from a User-written Program

The NHDPlusHR VAA Navigator consists of three parts (classes or modules):

**LoadSqlServerDB** –

Loads NHDPlusFlowlineVAA, NHDPlusFlow, and NHDPlusMegaDiv data for a single NHDPlusHR gdb into the working SQL Server database. This module can be called multiple times to load as many NHDPlusHR gdbs as necessary for the navigations.

If AddToExisting is False, any existing SQL database is replaced with a new one. If AddToExisting is True, the NHDPlusHR gdb content is added to the existing SQL database.

**MakeWorkingTable** –

Creates a working table to be used for a single navigation from the data previously loaded into the database via LoadSqlServerDB. The working table is named t<sessionid>\_vaa, where <sessionid> is a value generated by the user’s program using system date and time in the format of yyyymmddhhmmssss. Using a unique sessionid enables the Navigator to be called concurrently by multiple user programs. If the sessionid is unique each time a navigation is performed, the results are permanently saved in the working database and not overwritten during each navigation.

MakeWorkingTable uses the NavType to limit the data placed in the working table. For example, if the NavType is UpMain or UpTrib, only flowlines with Hydroseqno >= the starting Hydroseqno are loaded. If the NavType is DnMain or DnDiv, only flowlines with Hydroseqno <= the starting Hydroseqno are loaded.

The working table contains NHDPlusID, Hydroseq, LevelpathI, Pathlength, terminalpa, uplevelpat, uphydroseq, dnlevelpat, dnminorhyd, dndraincou, divergence, frommeas, tomeas, lengthkm, dnhydroseq. In addition, if the navigation is being filtered by the values in a particular attribute, the attribute is added to the table.

The fields From1, To1 and Selected are added to the table and set to null, null, and 0 respectively. These fields are used to keep track of which flowlines are navigated and which from measure and to measure are included in the navigation. From1 and To1 will be the bottom and top measures of the flowlines during full flowline navigation and may be measures within the flowline when point to point navigation is performed.

**V03Navigator** –

Performs a navigation and places the results in a SQL Server table named t<sessionid>\_navresults. This table is overwritten each time the navigator is called with the same <sessionid>.

If Navtype = UPMain,

The flowlines with levelpathi of the startNHDPlusID and hydroseqno >= that of the StartNHDPlusID are selected.

If the most upstream flowline has Divergence = 2, then the NHDPlusID of the flowline immediately upstream of the Divergence = 2, is used to perform the selection again.

This continues until the most upstream flowline has Divergence <> 2 or the MaxDistance has been reached or the attribute filter comparison is false.

If Navtype = DNMain,

The flowlines with levelpathi of the startNHDPlusID and with hydroseqno <= that of the StartNHDPlusID and with Terminalpa of the startNHDPlusID are selected.

If the most downstream flowline has TerminalFl = 0, then the NHDPlusID of the flowline immediately downstream, is used to perform the selection again.

This continues until the most downstream flowline has TerminalFl = 1 or the MaxDistance has been reached or the attribute filter comparison is false.

If NavType = “UPTrib”,

An UpTrib navigation is a series of UpMain navigations.

UpMain is performed for the StartNHDPlusID.

For each flowline in the results, where LevelPathI <> UpLevelPat <> 0, UpMain is performed using that flowline as a new StartNHDPlusID.

This continues until there are no longer flowlines in the results with LevelPathI <> UpLevelPat <> 0 or the MaxDistance has been reached or the attribute filter comparison is false.

If NavType = “DNDiv”,

A DNDiv navigation is a series of DNMain navigations.

DNMain is performed for the StartNHDPlusID.

For each flowline in the results, where DnCount> 1, DnMain is performed using each of the minor outflowing flowlines as a new StartNHDPlusID.

This continues until there are no longer flowlines in the results with DnCount > 1 or the MaxDistance has been reached or the attribute filter comparison is false.

**Properties:**

The properties in RED are required by the Navigator.

SQLDataSource – input property

Type: String

Applies to calls to: LoadSqlServerDB, MakeWorkingTable, V03Navigator

Value: Name of the SQL Server datasource on your local computer. Example: Datasource=”(LocalDB)\v11.0”

SQLCommandTimeout – Input property

Type: Integer

Applies to calls to: LoadSQLServerDB, MakeWorkingTable, V03Navigator

Value: Timeout in seconds

SQLConnectionTimeout – Input property

Type: Integer

Applies to calls to: LoadSQLServerDB, MakeWorkingTable, V03Navigator

Value: Timeout in seconds

AttrName – Input property

Type: String

Applies to calls to: MakeWorkingTable, V03Navigator

Value: Field name of filter attribute. Example: Attrname = “TotDASqKM”

AttrValue – Input property

Type: String

Applies to calls to: MakeWorkingTable, V03Navigator

Value: value for filter attribute. Example: AttrValue = “50000”

AttrComp – Input property

Type: String

Applies to calls to: MakeWorkingTable, V03Navigator

Value: Operator for evaluating filter attribute value. Example: AttrComp = “>=”

TempWorkAreaPath – input property

Type: String

Applies to calls to: LoadSQLServerDB, MakeWorkingTable

Value: Path to a temporary work area. The user must have read/write privileges to this path. Example: TempWorkAreaPath=”D:\Working”

AddToExisting – input property

Type: Boolean

Applies to calls to: LoadSqlServerDB

Value:

False – for the first VPU

True – for second and subsequent VPUs

DatabaseLocation – input property

Type: String

Applies to calls to: LoadSqlServerDB, MakeWorkingTable, V03Navigator

Value: Path to the navigation SQL Server database. The user must have read/write privileges to this path. Example: DatabaseLocation=”D:\NHDPlusHRData\Navdbs”

DatabaseName – input property

Type: String

Applies to calls to: LoadSqlServerDB, MakeWorkingTable, V03Navigator

Value: Name for the navigation SQL Server database

Recommendation: Use a drainageID (DD) or a VPUid (VVVVVVVV) in the name, as appropriate.

InputNHDPlusLocation – input property

Type: String

Applies to calls to: LoadSqlServerDB

Value: Folder location where the NHDPlusHR data is stored. Example: InputNHDPlusLocation = “D:\NHDPlusHRData “

SessionID – input property

Type: StringApplies to calls to: MakeWorkingTable, V03Navigator

Value: SessionID for the navigation, unique value based on the computer system date and time allowing multiple concurrent calls to the Navigator

WorkingTableName – input/output property

Type: String

Applies to calls to: MakeWorkingTable (output), V03Navigator (input)

Value: Table name (of the form t<sessionid>\_VAA) for the working table that holds the VPU data. This table is overwritten each time MakeWorkingTable is called with the same <sessionid>.

Navtype – input property

Type: String

Applies to calls to: V03Navigator

Value: UPMAIN or UPTRIB or DNMAIN or DNDIV

StartNHDPlusID – input property

Type: Numeric

Applies to calls to: V03Navigator

Value: Starting NHDFlowline Comid for the navigation, must be greater than 0

StartMeasure – input property

Type: Numeric

Applies to calls to: V03Navigator

Value: Starting measure for the navigation, must be between 0 and 100 inclusive or -1

For –1, the Navigator starts at the “from” measure (i.e. bottom) of the NHDFlowline feature for upstream navigations and at the “to” measure (i.e. top) of the NHDFlowline feature for downstream navigations.

MaxDistance – input property

Type: Numeric

Applies to calls to: V03Navigator

Value: Maximum travel distance in kilometers, greater than or equal to 0

A value of 0 indicates no maximum travel distance will be applied and that the Navigator wills navigate each path until the path ends (i.e. at a headwater or a network terminus). 0 is the default value.

ProcessStatus – output property

Type: Integer

Applies to calls to: LoadSqlServerDB, MakeWorkingTable, V03Navigator

Value:

900 – unsuccessful completion

0 - successful completion

ProcessMessage – output property

Type: String

Applies to calls to: LoadSqlServerDB, MakeWorkingTable, V03Navigator

Value: For unsuccessful completions, there is a message about the problem that was encountered.

**Methods: After declaring and setting the properties above, the user program calls the each of the following methods to execute a navigation.**

LoadSQLServerDB –

Function: Loads \NHDPlusAttributes\PlusFlowlinevaa, PlusFlow and MegaDiv for a VPU into the working database. Call this function once for each NHDPlusHR gdb that is included in the navigation.

MakeWorkingTable –

Function: Creates the working table to be used for a navigation from the VPU data loaded into the working database. Call this function once for each navigation.

VAANavigate –

Function: Performs a navigation. Returns 1 if there is a known problem, 0 otherwise. Results of the navigation are placed in t<sessionid>\_NavResults in the working SQLServer database. Call this function once for each navigation.

**Python Example**

# import the required libraries

import sys, os, time, string, win32com.client, shutil, datetime, csv, arcpy

from arcpy import env

# Declare the name of the SQL Server database that will be used, within this script,

# by the NHDPlusHRVAANavigator to store the NHDPlus data and navigation results

DBname = "DSNavDBs"

# Declare the path to store the NHDPlusHRVAANavigator SQL Server database

# used to process the navigation

DBlocation = "D:\NHDPlusHRData\NavigatorDBs"

# Declare the name of the comma delimited text file containing the paths to the NHDPlus data needed for navigations in this run.

# For example:

# D:\NHDPlusHRData\NHDPlusHR0601.gdb

# Declare the name of the working folder for data processing.

# This folder may or may not exist and will be created if it doesn't.

WorkingFolder = "D:\Working"

# Initialize the NHDPlusHRVAANavigator objects

o1 = win32com.client.Dispatch("NHDPlusHRVAANavigator.clsLoadSQLServerDB")

o2 = win32com.client.Dispatch("NHDPlusHRVAANavigator.clsMakeWorkingTable")

o3 = win32com.client.Dispatch("NHDPlusHRVAANavigator.clsV2Navigator")

# Create a session\_id for this run which is added to the working folder. Each execution of the script has a unique session id.

now = datetime.datetime.now()

session\_id = datetime.datetime.strftime(now, "%Y%m%d%H%M%S%f")[:16]

# Create the temporary working folder

TmpWorkAreaPath = WorkingFolder + "\\" + session\_id

if not os.path.exists(TmpWorkAreaPath):

os.makedirs(TmpWorkAreaPath)

# Load the SQL Server database (DBname) with the NHDPlusHR gdb needed for the navigations in this run.

# Set the variable for the return value from LoadSQLServerDB

intReturn = 0

# Set common properties for LoadSQLServerDB

o1.DatabaseLocation = DBlocation

o1.DatabaseName = DBname

o1.TempWorkAreaPath = TmpWorkAreaPath

o1.SQLConnectionTimeout = 120

o1.SQLCommandTimeout = 120

o1.AddToExisting = False

o1.InputNHDPlusLocation = NHDPlusDataPath

#Call the method to load the data from an NHDPlusHR gdb

intReturn = o1.LoadSQLServerDB

# Check for any errors returned from LoadSQLServerDB

if intReturn > 0:

exit()

# For each navigation, call MakeWorkingTable followed by VAANavigate

# Set the variable for the return value from MakeWorkingTable

intReturn = 0

# Set common properties for MakeWorkingTable

o2.DatabaseLocation = DBlocation

o2.DatabaseName = DBname

o2.TempWorkAreaPath = TmpWorkAreaPath

o2.SessionID = session\_id

o2.SQLConnectionTimeout = 120

o2.SQLCommandTimeout = 120

o2.StartComid = ComID

o2.Navtype = "DNDIV"

intReturn = o2.MakeWorkingTable

# Check for any errors returned from MakeWorkingTable

if intReturn > 0:

exit()

## VAANavigate ##

# Set the variable for the return value from VAANavigate

intReturn = 0

# Set common properties for VAANavigate

o3.DatabaseLocation = DBlocation

o3.DatabaseName = DBname

o3.TempWorkAreaPath = TmpWorkAreaPath

o3.SessionID = session\_id

o3.SQLConnectionTimeout = 120

o3.SQLCommandTimeout = 120

o3.WorkingTableName =o2.WorkingTableName

o3.StartComid = ComID

o3.StartMeasure = Measure

o3.NavType = "DNDIV"

o3.MaxDistance = MaxDist

o3.SaveResults = False

intReturn = o3.VAANavigate

# Check for any errors returned from VAANavigate

if intReturn > 0:

exit()