Ministry of Agriculture and Lands
Crown Registry and Geographic Base

Freshwater Atlas v1.3

Stream Elevation (SE) Generation

Technical Specification

Prepared for

Crown Registry and Geographic Base
Integrated Land Management Bureau
Ministry of Agriculture and Lands

Prepared by

Refractions Research Inc.
#300 – 1207 Douglas Street
Victoria, BC
www.refractions.net

Version: 0.7 (DRAFT)
Last Updated: January 19, 2018
ATER ATLAS-SE.TechnicalSpecification.v0.7.doc

THE GEOSPATIAL EXPERTS

REVISION HISTORY

Date	Version	Author	Changes
Jan 25, 2007	0.1	Refractions Research Inc. *Martin Davis* *Kevin Neufeld*	
Feb 19, 2007	0.2	Refractions Research Inc *Emily Gouge*	Stubbed smoothed elevation and gradient sections. Updated database design section. Updated Table1 to reflect changes to rawz processing. Added step 5 to raw z processing for dealing with generated features.
Mar. 26, 2007	0.3	Refractions Research Inc. *Martin Davis* *Kevin Neufeld* *Emily Gouge*	Populated smoothed elevation section.
Mar. 28, 2007	0.4	Refractions Research Inc. *Martin Davis* *Kevin Neufeld* *Emily Gouge*	Revised smoothed elevation section with CRGB's comments.
Apr. 17, 2007	0.5	Refractions Research Inc. *Kevin Neufeld*	Renamed to "SEG" Changed Spike definition
May 15, 2008	0.6	Refractions Research Inc. *Ben Brehmer* *Martin Davis*	Renamed to "SE" Altered and made additions wherever necessary to concur with current deliverable. Artifical Waterfall discussion added.
June 11, 2008	0.7	Refractions Research Inc. *Kevin Neufeld* *Martin Davis*	Corrected the descriptions of the Refined Elevation algorithms Numerous edits to clarify wording and processing descriptions

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

1 INTRODUCTION

1.1 PURPOSE

This document specifies the design of the algorithms and processing performed to generate SE (raw elevation, monotonic/smoothed elevation) data for the Freshwater Atlas database.

1.2 SCOPE

This document includes information about the following:

- Input datasets (including any processing required to prepare them)
- Algorithms used to determine the computed dimensional values
- Processes used to perform computations and update the database

1.3 INTENDED AUDIENCE

The primary audience for this document is CRGB technical staff involved in the Freshwater Atlas project. The information may also be of interest to users of the SEG data.

1.4 RELATED DOCUMENTS

- *SE Generation Technical Issues*
- *TRIM Watershed Atlas Elevation Preprocessing and Stream Gradient Calculation* (Barrodale Computing Services)

2 OVERVIEW OF THE SE PROCESS

The Freshwater Atlas SE (Stream Elevation) Process computes values related to the height (elevation or Z dimension) for specific feature classes in the Freshwater Atlas database. The values computed are:

- raw elevation
- refined (monotonic/smoothed) elevation

These values are computed in separate phases. The values computed in a phase depend on the data computed in the previous phase.

3 TARGET FRESHWATER ATLAS FEATURE CLASSES

The data in the Freshwater Atlas database is organized into many feature classes. Feature classes represent different kinds of natural or man-made phenomena (e.g. lakes, rivers, glaciers, boundaries) and/or different methods by which the data was created (e.g. observed, generated).

In some cases not all of the 3D attributes are applicable to particular Freshwater Atlas feature classes. The table below lists Freshwater Atlas feature classes and which SE attributes are computed for them. For each feature class, the table shows which algorithms are applied to compute the SE attributes. The algorithm codes are supplied below. Each algorithm is described further in following sections. "-" indicates values which are not computed for that feature class.

Table 1: Freshwater Atlas Feature Classes and SE Values Computed

Code	Feature Name	Raw Elev.	Smoothed Elev.
100	Coastline	CID	F
150	Construction line, coastline	ED	F
1000	Single line blueline, main flow	CID	R
1050	Single line blueline, main flow through wetland	CID	R
1100	Single line blueline, secondary flow	CID	R
1150	Single line blueline, secondary flow through wetland	CID	R
1200	Construction line, main flow	ED	R
1250	Construction line, double line river, main flow	ED	R
1300	Construction line, secondary flow	ED	R
1325	Construction line, segment delimiter	ED	R
1350	Construction line, double line river, secondary flow	ED	R
1375	Construction line, river delimiter	ED	R
1400	Construction line, other flow/inferred connection	ED	R
1410	Construction line, network connector	ED	R
1425	Construction line, subsurface flow	ED	R
1450	Construction line, connection	ED	R
1475	Construction line, lake arm	ED	R
1500	Lake shoreline	CID	L
1525	Lake shoreline shared with a wetland	CID	L
1550	Construction line, lakeshore	ED	L
1600	Island shoreline	CID	L
1625	Island shoreline shared with a wetland	CID	L
1700	Wetland shoreline	CD	
1800	Double line blueline, right bank	CID	
1825	Double line blueline, right bank shared with wetland	CID	
1850	Double line blueline, left bank	CID	
1875	Double line blueline, left bank shared with wetland	CID	
1900	Island in river, right bank	CID	
1925	Island in river, right bank shared with wetland	CID	

Code	Feature Name	Raw Elev.	Smoothed Elev.
1950	Island in river, left bank	CID	
1975	Island in river, left bank shared with wetland	CID	
2000	Single line, Canal	CID	R
2100	Double line, Canal or Reservoir	CID	
2300	Single-line, Canal, secondary flow	CID	R
5000	Major watershed boundary	ED	
5100	Minor watershed boundary	ED	
5200	Trivial watershed boundary	ED	
5900	Isolated watershed boundary	ED	
6000	International Boundary	ED	
Additional Feature Classes			
	Glaciers		
	Bays		
	Channels		
	Named Points		
	Obstructions (dams, rapids, beaver dams, falls)		

Table 2: Algorithms for Computing SE Values

Code	Algorithm Description
Raw Elevation	
A	Assigned
C	Conflated
E	Endpoint assigned
I	Interpolated
D	Draped
Smoothed Elevation	
F	Leveled to a fixed value
Sp	Spike removal/Cleaning
L	Lake smoothing
R	Stream/River smoothing
I	Interpolated from adjacent

4 COMPUTATION OF RAW ELEVATION

4.1 OVERVIEW

Raw elevation computation involves assigning a raw elevation (z) value to vertices of lakes, lake banks and the stream network.

In order to preserve observed TRIM elevation values as much as possible, the initial assignment of raw elevation is performed via conflation to TRIM features. Conflation involves matching vertices of observed Freshwater Atlas features to the corresponding hydrographic features found in TRIM1. When a vertex is matched the original TRIM1 elevation value is transferred to the Freshwater Atlas vertex.

All remaining unassigned vertex elevations are computed via a combination of constrained assignment, interpolation, and draping the vertices over a surface model. The surface model is defined by data from both TRIM1 and TRIM2, in order to conform to the Freshwater Atlas linework as closely as possible while retaining the most accurate DEM information available.

4.1.1 Goals

The design of the Raw Elevation process is intended to satisfy the following goals:

1. Freshwater Atlas features should have a consistent elevation model. In particular, any two vertices with the same horizontal location must have the same elevation.
2. Wherever possible (i.e. where this does not conflict with other goals), the original TRIM observed elevation should be preserved as the Freshwater Atlas elevation.

4.2 PROCESS FLOW

The process flow (as seen in Figure 1: Raw Z Process) comprises the following steps:

1. Conflate the vertices of observed Freshwater Atlas features to the corresponding hydrographic features found in TRIM1.
    a. Resolve coincident vertices with inconsistent elevations found in TRIM1 features used for conflation.
    b. Conflate observed Freshwater Atlas features to TRIM1.
2. Assign endpoints of non-observed Freshwater Atlas features where they are constrained (e.g. skeleton line endpoints are assigned the same elevation as the lake bank they connect to).
3. Interpolate the Z-value of non-conflated vertices where appropriate (i.e. where the vertex lies between two other vertices on the same feature).

4. Drape remaining unassigned vertices.
    a. Resolve duplicate vertices found in features used to build the surface model (these include TRIM1 hydrographic and TRIM2 non-hydrographic DEM features).
    b. Build a TIN using a horizontally unique set of vertices
    c. Drape all remaining non-elevation-assigned Freshwater Atlas vertices (i.e. network connectors, skeletons, missing / modified streams, etc…) over the generated surface model.

These steps are described in more detail in the following sections.

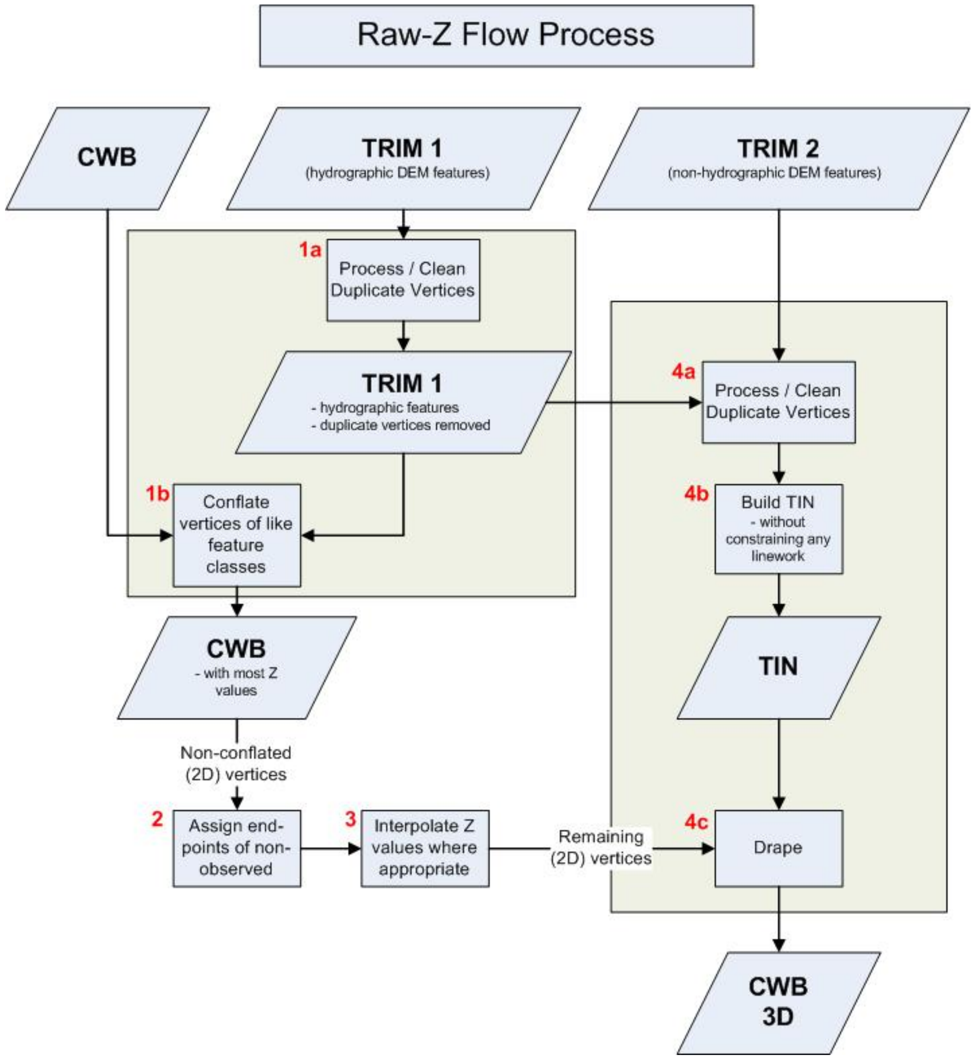


Figure 1: Raw Z Process Flow

4.3 CONFLATION

In order to preserve observed TRIM elevation values as much as possible, the initial assignment of raw elevation is performed via conflation. Conflation involves matching vertices of observed Freshwater Atlas features to the corresponding hydrographic features found in TRIM1. When a vertex is matched the original TRIM1 elevation value is transferred to the Freshwater Atlas vertex.

4.3.1 Input Data

The input data for conflation is the vertex data for all relevant Freshwater Atlas features and reference TRIM1 features. Only certain Freshwater Atlas feature classes participate in conflation (since some have no equivalent in TRIM, e.g. skeleton lines).

In order to minimize elevation inconsistencies, Freshwater Atlas vertices in a given feature class are conflated only to vertices within corresponding TRIM feature classes (e.g. streams to streams, lakes to lakes).

The following table specifies which Freshwater Atlas and TRIM feature classes participate in conflation.

Table 3: Freshwater Atlasto TRIM Feature Class Conflation Correlation

Code	Feature Name	Freshwater Atlas **Feature Code**	**TRIM1 Feature Code**	**TRIM1 Description**
Coastlines				
100	Coastline	GG05800000	GG05800000	Coastline:definite
			GG95800130	Coastline:indefinite
		GE14850000	GE14850000	Island
Lakes				
1500	Lake shoreline	AP09200000	AP09200000	DesignatedArea:dump
		AP90300100	AP90300100	TailingPond
		EA26700110	EA26700110	DesignatedArea:settlingPond
		GB15300000	GB15300000	LakeReservoir:lake:definite
		GB15300130	GB15300130	LakeReservoir:lake:indefinite
		GB15300140	GB15300140	LakeReservoir:lake:intermittent
		GB24300000	GB24300000	LakeReservoir:reservoir:definite
		GB90100000	GB90100000	LakeReservoir:reservoir:indefinite
		GB90100110	GB90100110	LakeReservoir:reservoir:intermittent
1525	Lake shoreline shared with a wetland	GB15300000	GB15300000	LakeReservoir:lake:definite
		GB15300140	GB15300140	LakeReservoir:lake:intermittent
1600	Island shoreline	GE14850000	GE14850000	Island
1625	Island shoreline shared with a wetland	GE14850000	GE14850000	Island
Double-Lined Rivers				
1800	Double line blueline, right bank	WA24200110	GA90000120	RiverStream:rightBank
1825	Double line blueline, right bank shared with wetland	WA24200120	GA90000120	RiverStream:rightBank
1850	Double line blueline, left bank	WA24200130	GA90000110	RiverStream:leftBank
1875	Double line blueline, left bank shared with wetland	WA24200140	GA90000110	RiverStream:leftBank
1900	Island in river, right bank	WA24220110	GE14850000	Island
1925	Island in river, right bank shared with wetland	WA24220120	GE14850000	Island
1950	Island in river, left bank	WA24220130	GE14850000	Island
1975	Island in river, left bank shared with wetland	WA24220140	GE14850000	Island
2100	Double line, Canal or Reservoir	AP90300100	AP90300100	TailingPond
		EA26700110	EA26700110	DesignatedArea:settlingPond
		GA03950000	GA03950000	Canal:canal
		GB24300000	GB24300000	LakeReservoir:reservoir:definite
		GB90100000	GB90100000	LakeReservoir:reservoir:indefinite
		GB90100110	GB90100110	LakeReservoir:reservoir:intermittent
Single-lined Streams				

Code	Feature Name	Freshwater Atlas **Feature Code**	**TRIM1 Feature Code**	**TRIM1 Description**
1000	Single line blueline, main flow	GA24850000	GA24850000	RiverStream:definite
		GA24850140	GA24850140	RiverStream:indefinite
		GA24850150	GA24850150	RiverStream:intermittent
1050	Single line blueline, main flow through wetland	GA24850000	GA24850000	RiverStream:definite
		GA24850140	GA24850140	RiverStream:indefinite
		GA24850150	GA24850150	RiverStream:intermittent
1100	Single line blueline, secondary flow	GA24850000	GA24850000	RiverStream:definite
		GA24850140	GA24850140	RiverStream:indefinite
		GA24850150	GA24850150	RiverStream:intermittent
1150	Single line blueline, secondary flow through wetland	GA24850000	GA24850000	RiverStream:definite
		GA24850140	GA24850140	RiverStream:indefinite
		GA24850150	GA24850150	RiverStream:intermittent
2000	Single line, Canal	GA03950000	GA03950000	Canal:canal
		GA08800110	GA08800110	Canal:ditch
2300	Single-line, Canal, secondary flow	GA03950000	GA03950000	Canal:canal
		GA08800110	GA08800110	Canal:ditch
Wetlands				
1700	Wetland shoreline	GB11350110	GB11350110	FloodedLand:inundatedLand
		GB15300000	GB15300000	LakeReservoir:lake:definite
		GC17100000	GC17100000	FloodedLand:marsh
		GC30050000	GC30050000	FloodedLand:swamp
		WA17100000	N/A	
Glaciers				
	Glaciers	N/A	GD12300000	IceMass:glacier
			GD14450000	IceMass:icefield

4.3.2 Resolving Inconsistent TRIM Elevations

In order to resolve duplicate vertices with inconsistent elevations within TRIM1 input, the following approach is taken:

1. For coincident vertices of the same feature class, the elevation is chosen to be the original elevation closest to the distance-weighted average of the elevations of the adjacent vertices on the coincident features.
2. Coincident vertices of different features classes are first reduced to the set from the highest-precedence feature class (as shown in Table 3). If this set contains multiple vertices, Rule 1 is then used to determine the elevation. The feature class precedence is:
    a. Coastlines
    b. Lakes (including reservoirs)
    c. Double-line Rivers (including double-lined canals)
    d. Single-line streams (including man-made water features)
    e. Wetlands
    f. Glaciers / Icefields

For example, if two lake bank vertices are coincident with a single-line stream vertex and the Z-values of the vertices differ, the candidate set is reduced to the lake bank vertices, and the elevation is computed as the one closest to the average of the adjacent vertices on the lake bank features.

4.3.3 Conflation Process

In order to detect which Freshwater Atlas vertices remain unassigned after conflation, elevation values are initialized to a known non-occurring value.

A match tolerance of 1 metre is used.

In order to minimize elevation inconsistencies, Freshwater Atlas vertices in a given feature class are conflated only to vertices within corresponding TRIM feature classes (e.g. streams to streams, lakes to lakes).

4.4 CONSTRAINED ENDPOINT ASSIGNMENT

Constructed features (e.g. skeleton lines) are not elevation-assigned by conflation. However, often they are connected to observed features at their endpoints (e.g. a skeleton line will terminate in a lake bank.). In order to ensure consistency in assigned elevations, the endpoints of constructed features can be assigned from any assigned vertices which have the same location.

4.5 INTERPOLATION

In some cases vertices in the interior of features remain unassigned after conflation. If these vertices are sufficiently close to their neighbours, it is reasonable to simply interpolate their elevation value, because linear interpolation of close vertices produces essentially the same results as draping them on the TIN.

For features known to have a monotonic gradient (e.g. streams and lakes), the interpolation can simply assign height values proportional to the distance along the feature.

The interpolation method used is linear interpolation. To reduce the possibility of incorrectly estimating elevation, vertices are only interpolated if they are within a distance tolerance of 50m. As shown in Table 1 (Freshwater Atlas Feature Classes and SE Values computed), only flowing features are interpolated. Nodes are not interpolated.

4.6 DRAPING

Raw elevations for any vertices not previously assigned are computed by draping over a surface model.

The surface model used is a Triangulated Irregular Network (TIN) defined by specific feature classes from the TRIM datasets. The TIN will not be constrained to linear input features (i.e. a standard Delaunay Triangulation algorithm is used, rather than the more complex Constrained Delaunay Triangulation algorithm). This is justified on the basis that the vertex density of linear features is high enough that in almost all cases the TIN is faithful to the linework. Additionally, only a small percentage of the Freshwater Atlas vertices need to be determined by draping.

It is not possible to build a single surface model for the entire database due to memory constraints, so multiple overlapping surfaces are built according to a convenient partitioning of the provincial area (likely 1:20K mapsheet).

The surface is built to conform to the Freshwater Atlas linework as closely as possible while utilizing the most accurate DEM information available. To achieve this, data from both TRIM1 and TRIM2 is used. Non-hydrographic TRIM2 data (e.g. DEM masspoints) is used since it has undergone cleaning to remove data anomalies. Hydrographic data from TRIM1 is used since it most closely matches the hydrographic data in Freshwater Atlas.

4.6.1 Input Data for Surface Model

The input data for the surface model requires a horizontally-unique set of vertices with elevation assigned. (Since the surface model is not constrained, no line segment data is required.)

The choice of input data for defining the surface model needs to satisfy the following goals:

- The surface should have no data anomalies (these occur in the TRIM1 DEM masspoints, but have been substantially reduced in the TRIM2 DEM)
- The surface should conform to the Freshwater Atlas linework as closely as possible, minimizing obvious conflicts (e.g. in some cases TRIM2 streams & gullies cut across TRIM1/Freshwater Atlas streams)

To achieve this, the surface model is generated by using:

- Hydrographic DEM feature classes from TRIM1 (i.e. lakes, dykes)
- Non-hydrographic DEM feature classes from TRIM2 (i.e. DEM masspoints, non-hydrographic breaklines)

The following table lists the feature classes that define the DEM according to TRIM specifications. Each feature class is flagged to show whether it is included in the vertex set used to define the surface for draping, and if so which dataset it is obtained from.

Table 4: TRIM Feature Classes for Surface Definition

Feature Class	Feature Code	Source	Included?
AreaOfExclusion	HC90000000	TRIM2	Partially – vertices with elevations of 0 meters are not included; all other vertices are included
AreaOfIndefiniteContours			
IceMass			
glacier	GD12300000	TRIM1	Yes
icefield	GD14450000	TRIM1	Yes
DEM			

spotHeight	HA90100000	TRIM2	Yes
indefinite	HA90100110	TRIM2	Yes
hybrid	HA90300000	TRIM2	Yes
Breakline			
anthropogenic	HA90200140	TRIM2	Yes
hypsographic	HA90200120	TRIM2	Yes
round	HA90200110	TRIM2	Yes
sharp	HA90200000	TRIM2	Yes
hydrographic			
BeaverDam	GA08450110	TRIM1	Yes
Canal			
canal	GA03950000	TRIM1	Yes
canalLeftBank	GA90001110	TRIM1	Yes
canalRightBank	GA90001120	TRIM1	Yes
ditch	GA08800110	TRIM1	Yes
Coastline			
definite	GG05800000	TRIM1	Yes
indefinite	GG95800130	TRIM1	Yes
Dyke	GE09400000	TRIM1	Yes
FloodedLand			
inundatedLand	GB11350110	TRIM1	Yes
marsh	GC17100000	TRIM1	Yes
swamp	GC30050000	TRIM1	Yes
Island	GE14850000	TRIM1	Yes
LakeReservoir			
lake			
definite	GB15300000	TRIM1	Yes
indefinite	GB15300130	TRIM1	Yes
intermittent	GB15300140	TRIM1	Yes
proposedMaxLevel	GB90100120	TRIM1	Yes
reservoir			
definite	GB24300000	TRIM1	Yes
indefinite	GB90100000	TRIM1	Yes
intermittent	GB90100110	TRIM1	Yes
RiverStream			
definite	GA24850000	TRIM1	Yes
dry	GA24850130	TRIM1	Yes
indefinite	GA24850140	TRIM1	Yes
intermittent	GA24850150	TRIM1	Yes
leftBank	GA90000110	TRIM1	Yes
rightBank	GA90000120	TRIM1	Yes
Sand/GravelBar	GE90300000	TRIM1	Yes
DesignatedArea			
settlingPond	EA26700110	TRIM1	Yes
TailingPond	AP90300100	TRIM1	Yes

4.6.2 Resolving Inconsistent Elevations

As aforementioned, the input data for the surface model requires a horizontally unique set of vertices with elevations assigned. In order to resolve coincident vertices with inconsistent elevations for building the surface model, the following approach is taken. The surface is initially constructed with all vertices which have a unique elevation value (this includes duplicate vertices with the same elevation). The remaining duplicate vertices with inconsistent elevations are added to the surface incrementally. The elevation of a vertex is chosen to be the elevation which is closest to the current elevation of the surface at that location. This strategy should ensure that spikes or outliers on duplicate vertices do not affect the surface profile.

5 COMPUTATION OF REFINED ELEVATION

5.1 OVERVIEW

Refined elevation computation involves assigning a monotonic, smoothed elevation to most vertices in the Freshwater Atlas database.

As outlined in Table 1 (Freshwater Atlas Feature Classes and SE Values computed) different Freshwater Atlas features will be smoothed using different algorithms. There are four main steps in the smoothing process: fixed value, spike removal, lake smoothing, and stream/river smoothing.

5.1.1 Goals

The design of the Refined Elevation process is intended to satisfy the following goals:

1. The refined stream network and lake banks conform to a consistent elevation model. In particular, any two vertices with the same horizontal location will have the same refined elevation.
2. The refined stream network must have refined elevation values which are monotonically decreasing in the downstream direction. This network includes single line streams and skeletons of double-line rivers and lakes.
3. The stream network refined elevation values should be free from the effects of noise and data errors and of the original elevation precision model (e.g. 'stair-stepping' artifacts); yet retain the slope character of the original stream.
4. Coastline vertices should be assigned a single, fixed elevation.

5.1.2 Glossary

Edge	A linear feature which defines the geometry of the Freshwater Atlas stream network between two nodes. An edge contains two or more vertices, and has various attributes attached to it. Also sometimes referred to as "arc".
Vertex	(*pl*. **vertices**) An explicitly represented point on a stream network edge which defines the location of the edge at that point. Vertices have an (x.y) location, and also attached values defining the elevation at that point.
Segment	A single line segment of an **edge** which runs between two **vertices** on the edge. Segments are implicitly linear.
Section	A consecutive sequence of 1 or more **segments** on a stream **edge**.

Path	A contiguous sequence of 1 or more stream edges. The edges form a path through the stream network. The path may or may not represent a physical hydrological flow.
Z-Spike, Spike	A section of an edge or path which contains a vertex with an elevation local minima or maxima, and which forms an interior angle of 50 degrees or less between that vertex and the endpoints of the section.
Stream Network	The stream network includes all single line observed streams (1000, 1050, 1100, 1150, 2000, 2300), and all lake, river, and wetland skeletons (1200, 1250, 1300, 1350, 1400, 1450, 1475) in BC.
Refined Stream Network	The refined stream network contains all edges of the stream network. In the case of inferred connector edges (1410, 1425, and wetland skeleton edges - 1200, 1300, 1400) no refined elevation is computed, so the Z ordinate is equal to the raw elevation

5.2 SUMMARY OF PROCESSING

Refined elevation is computed by running a number of different algorithms on the stream network and raw elevation data. These algorithms address various issues including spike removal, lake smoothing, fixed value coastline and stream/river smoothing.

1. **Coastline Clamping:** All coastline elevations are set to an elevation of 1 m.

2. **Node Network Processing**

2.1. **Node Spike Removal:** Elevation spikes in the network which occur at nodes are detected and removed. Occasionally there are nodes which "spike" out of range when compared to the surrounding vertices. These spikes need to be removed, since subsequent algorithms may otherwise return skewed results. Identification is done by walking down the stream network looking for edge sections which constitute spikes. As defined above, a spike section is identified by computing the slope between adjacent vertices. Removal of node spikes is accomplished by determining all 2-edge paths which pass through the node of interest. The Vertex Spike Removal algorithm (see below) is applied to every path to remove the spike occurring on that path. Finally, the average z value at the node vertex is applied to the node.

2.2. **Lake Node Leveling:** Lake node elevation data are leveled in order to eliminate erroneous elevations on lake banks. The erroneous data may exist for several reasons including reservoir draw-down or different data collection dates. The initial leveled elevation of a lake is determined to be the average of the set of elevations of all observed lake bank vertices between the 90th and 95th percentile.

2.3. **Node Network Monotonization:** To monotonize the full network, the first step is to create a fully monotonically decreasing node network. Nodes in the network are monotonized by using the Bidirectional Node Averaging algorithm (see below for details). During this process nodes may move up or down in elevation. In particular, lakes (which are treated as a single effective node) may change in elevation.

2.4. **Lake Vertex Elevation Assignment:** Once the node network has been monotonized, the computed leveled elevation for each lake (which is the same for all nodes of the lake) is applied to all vertices on the lake bank edges.

3. **Stream Edge Refinement:** With the overall node network monotonized, the individual edges are processed to compute the refined elevation for each vertex.

3.1. **Vertex Spike Removal:** internal elevation spikes are removed using the Vertex Spike Removal algorithm.

3.2. **Vertex Elevation Clipping:** the internal vertex elevations are clipped to lie within the endpoint node elevations (thus preserving overall monotonicity).

3.3. **Edge Elevation Monotonization:** The interior vertex elevations are monotonized using the Bidirectional Vertex Averaging algorithm.

3.4. **Edge Elevation Smoothing:** All stream edges (except for Wetland Construction lines [1200,1300,1400] and Inferred lines [1410,1425]) are smoothed in order to remove any "stair-step" artifacts remaining from the TRIM 1m elevation precision model. Smoothing is accomplished by running a statistical averaging algorithm over the edge vertex elevations.

4. **Snap Wetland Construction Lines:** Wetland Construction lines (1200, 1300, 1400) and Inferred lines (1410, 1425) are not monotonized/smoothed, since they cannot be assumed to flow downhill. Instead, their raw elevation values are used as the refined elevation. In order to ensure vertical connectivity, their node elevations are snapped to adjacent refined nodes.

5**. QA:** Due to data anomalies and lake leveling, "artificial waterfalls" may be introduced at node points. A final QA check is run to detect these situations, and some more extreme examples are corrected via manual adjustment to the computed raw Z.

5.3 DATA PRECISION

The raw TRIM elevation data has a vertical precision model of 1m and a vertical absolute accuracy of ± 5m.

All refined elevation data in the Freshwater Atlas dataset is stored with a 10^{-5} precision model. This is consistent with the horizontal precision model used in Freshwater Atlas.

5.4 FIX COASTLINE ELEVATION

The coastline elevation value is set to 1 m.

This implies that no Freshwater Atlas elevation value in BC will be lower than 1 m.

5.5 PROCESS NODE NETWORK

5.5.1 Node Spike Removal

For descriptions and examples of issues see *Section 1.3 Node Spike Examples* in *Issues* document.

Detection

Node spikes are identified by looking for large steep uphill slopes on the most downstream segment.

```
for each flow edge
      if (the difference of the z-value in the 1st and 2nd vertices is greater than 5m)
              flag edge as a candidate
```

See also *Section 1.3.3 Identifying Node Spikes* in issues document.

Resolution

Node spikes are removed using the following algorithm:

```
for each downstream path through the node
      temporarily merge all edges in path.
      Call VERTEX_SPIKE_REMOVAL on the merged path
end for
set the node elevation = average z-value at the node.
```

See also *Section 1.3.4 Resolving Node Spikes* in issues document.

5.5.2 Lake Node Leveling

Lake nodes are leveled to the “cleaned average elevation” of all observed lake bank vertices. (Note: This leveled value may be further adjusted when creating the monotonically decreasing node network.)

The “leveled average” is computed as follows:

```
for each lake
  compute leveled_elevation = average of all lake vertices between the 90th and 
95th percentile
  for each vertex in lake bank edges
      set refined elevation = leveled_elevation
```

See also *Section 1.4 Lake Leveling* in issues document.

5.5.3 Node Network Monotonization

Description:

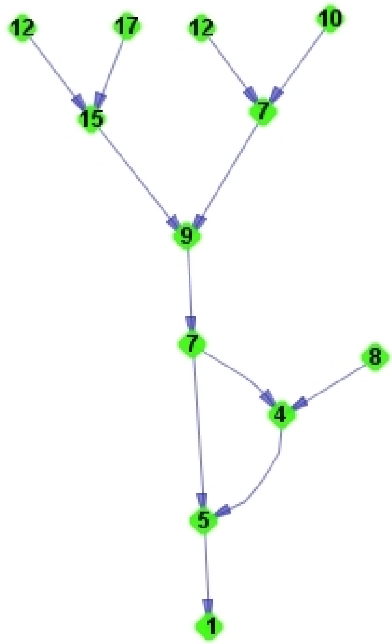
The first step in creating a monotonically decreasing stream network is to ensure that the network of nodes is monotonically decreasing.

Algorithm:

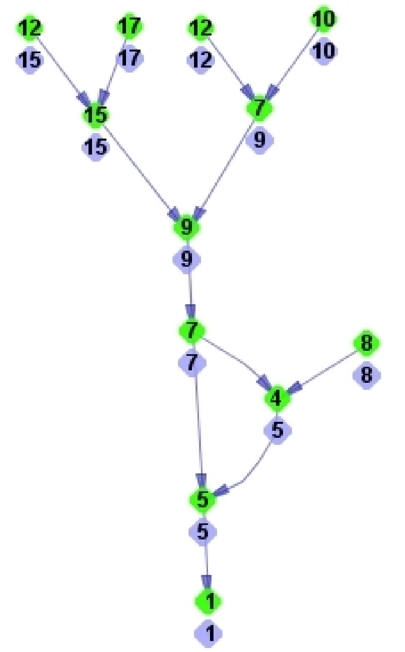
The algorithm for computing a monotonically decreasing node network (allowing certain nodes to be fixed) is:

```
Algorithm BIDIRECTIONAL_NODE_AVERAGING
walk up the node network for each node:
      if isfixed(node) then
             running_fixed = maximum(running_fixed, current_elevation)
      end if
current_elevation = maximum(running_fixed, current_elevation)
walk down the node network for each node:
      if isfixed(node) then
             running_fixed = minimum(running_fixed, current_elevation)
      end if
      current_elevation = minimum(running_fixed, current_elevation)
walk up the node network for each node:
running_maximum = max (running_maximum, current_elevation)
current_elevation = running_maximum
walk down the node network for each node:
      running_minimum = min (running_minimum, current_elevation)
      current_elevation = running_minimum
for each node:
      node_elevation = average(running_minimum, running_maximum)
```

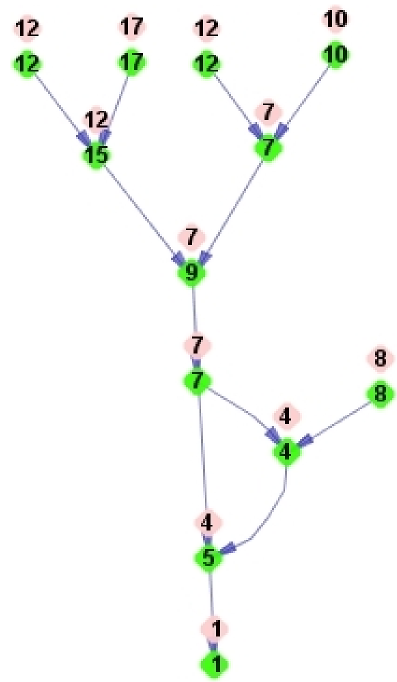
The implementation treats lakes as nodes; so that if one part of the lake is modified the entire lake is modified. This ensures that the nodes for each lake have the same leveled elevation in the output.



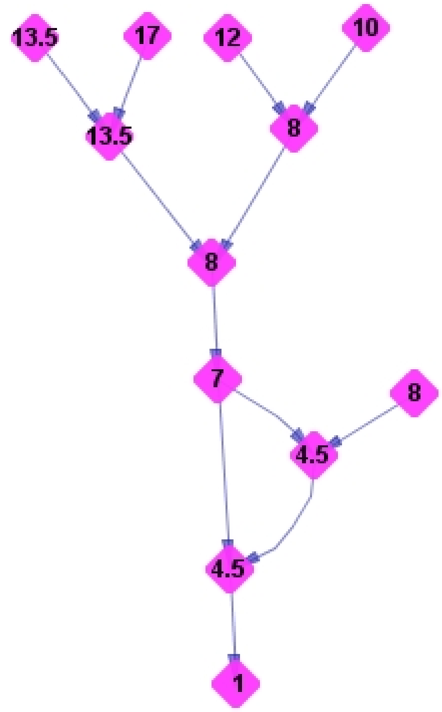
Refined stream network with raw elevation



Refined stream network with raw running maximum elevation



Refined stream network with raw and running minimum elevation



resulting monotonically decreasing
refined stream network

Figure 2: Monotonic algorithm.

5.6 PROCESS STREAM EDGES

Once the nodes in the network have been assigned monotonic elevations, the vertices along segments between the nodes are processed individually to ensure that they are monotonic and smoothed.

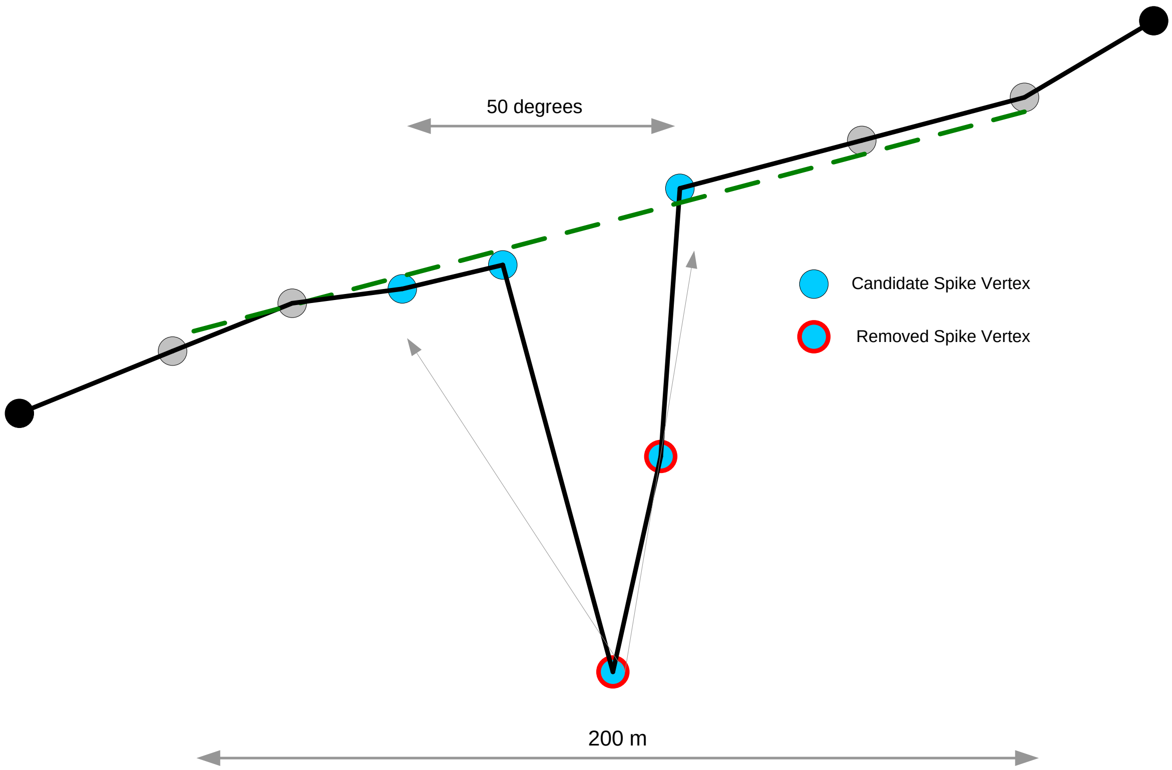
```
For every stream edge
      Spikes are removed from stream edge
      Interior vertices are clipped to the values of the endpoints
      Monotonicity is enforced on all interior edge vertices
      The edge is smoothed using a chi-square line-fitting algorithm
      Clipping and Monotonicity are again enforced.
```

5.6.1 Vertex Spike Removal

Vertex spikes within a stream edge will skew the smoothing of the edge. They are removed at the start of processing.

The spike removal algorithm is designed to detect and remove spikes while causing no or minimal change to vertices which do **not** form part of a spike. This is a challenging goal, which requires a couple of different heuristic approaches to handle all observed situations.

The following diagram shows how candidate spikes are detected and removed using line-fitting.



The vertex spike removal algorithm:

```
VERTEX_SPIKE_REMOVAL(edge)

for each edge interior vertex v

    if v is a local minima or maxima AND the adjacent vertices downstream and
    upstream form a spike[1]

      fit a line to the points that lie within 100m on either side of the spike
      (excluding the vertices in the spike)

      interp_z = interpolated Z value from fitted line

      if v.z is more than 6 m from interp_z

             // this prevents altering non-spike vertices

             set v.z = interp_z

// Some spike situations are not handled by the previous code (e.g. spikes less
than 6 m high, spikes close to endpoints, repeated alternating spikes).  The
following less-selective algorithm handles these cases.

for each edge interior vertex v

      if v is still a single vertex spike
```

[1] Given vertices A, B, and C, a vertex spike is determined to be located at vertex B where the angle between AB and BC is greater than 50°

```
interpolate the single vertex spike using a line from the adjacent
vertices.
```

5.6.2 Vertex Elevation Clipping

The elevations of interior vertices of a stream edge must not exceed the maximum or minimum elevation values of the edge's end nodes. To ensure this, the elevation values of these vertices are clipped prior to smoothing.

Clipping is performed after spike removal, since it would render spikes harder to detect.

```
for each internal vertex v
      if (v.z > upstream_node_elevation) then
             v.z = upstream_node_elevation
      end if
      if (v.z < downsteam_node_elevation) then
             v.z = downstream_node_elevation
      end if
end for
```

5.6.3 Edge Elevation Monotonization

This process adjusts the elevations of edge interior vertices to ensure that all vertices within an edge are monotonically decreasing.

The algorithm described below follows the approach outlined in the "*TRIM Watershed Atlas Elevation Preprocessing and Stream Gradient Calculation*" document.

```
Algorithm BIDIRECTIONAL_VERTEX_AVERAGING
walk up the edge - for each vertex:
      running_maximum = max (running_maximum, current_elevation)
      current_elevation = running_maximum
walk down the edge - for each vertex:
      running_minimum = min (running_minimum, current_elevation)
      current_elevation = running_minimum
for each vertex:
      vertex_elevation = average(running_minimum, running_maximum)
```

5.6.4 Edge Elevation Smoothing

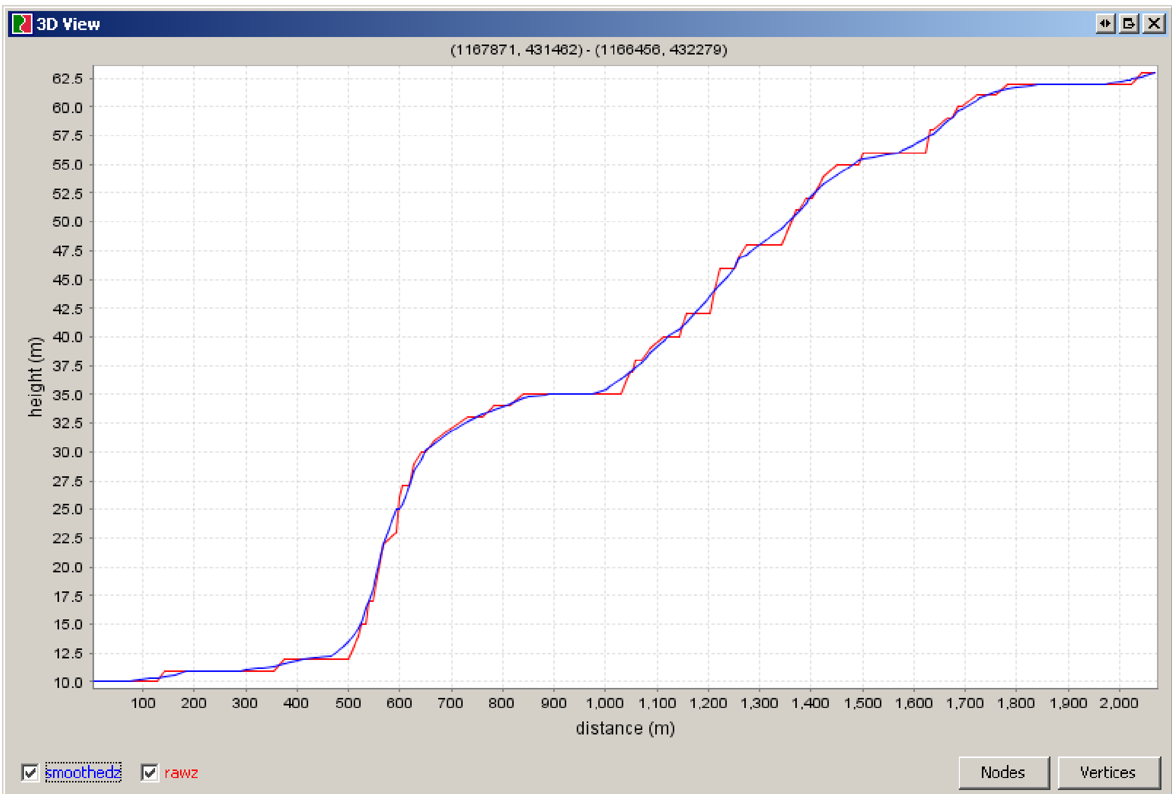
At this stage the refined stream network is monotonically decreasing in a downhill direction. However, the 1m TRIM precision model used for the raw data results in 'stair stepping’. This stair-stepping must be removed by applying a smoothing algorithm to the data.

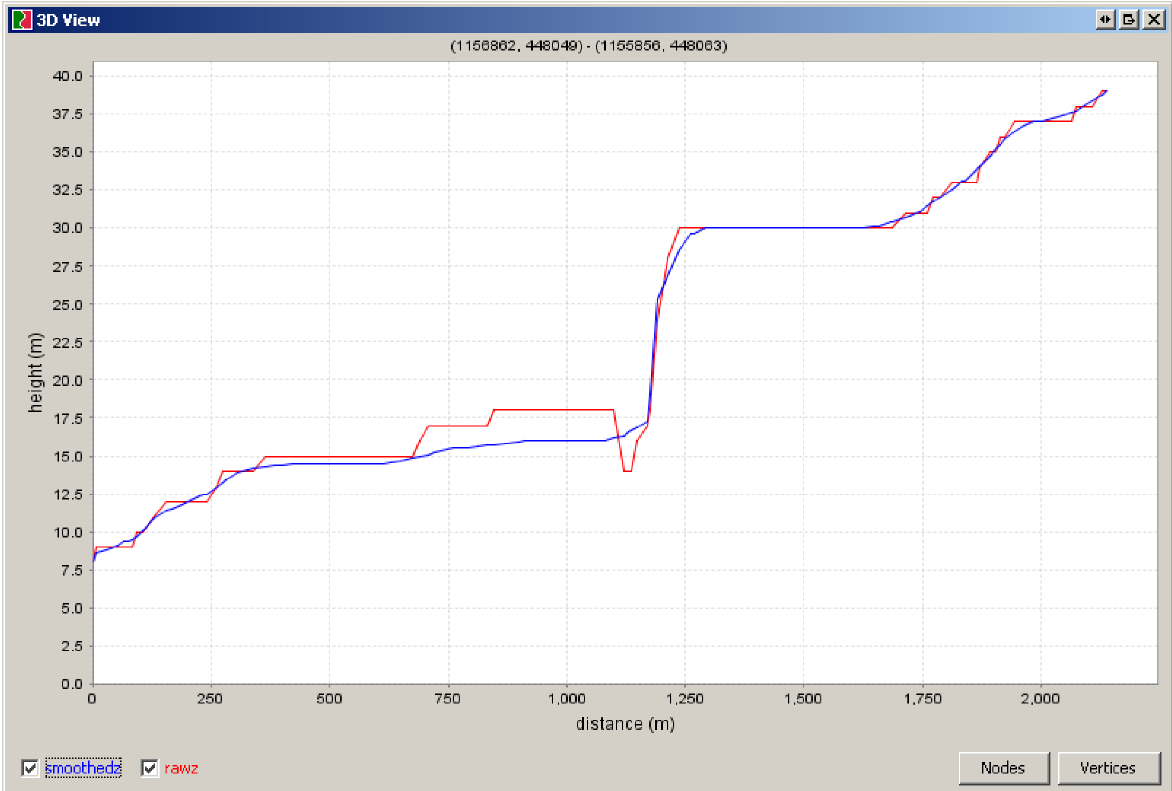
See *Section 1.7 Stream Edge Smoothing* in *Issues* document.

The smoothing algorithm used is the chi-squared algorithm described in the “*TRIM Watershed Atlas Elevation Preprocessing and Stream Gradient Calculation*” document.

After this algorithm has been run, as a final QA step the vertices on each edge are checked to ensure that they are still monotonically decreasing.

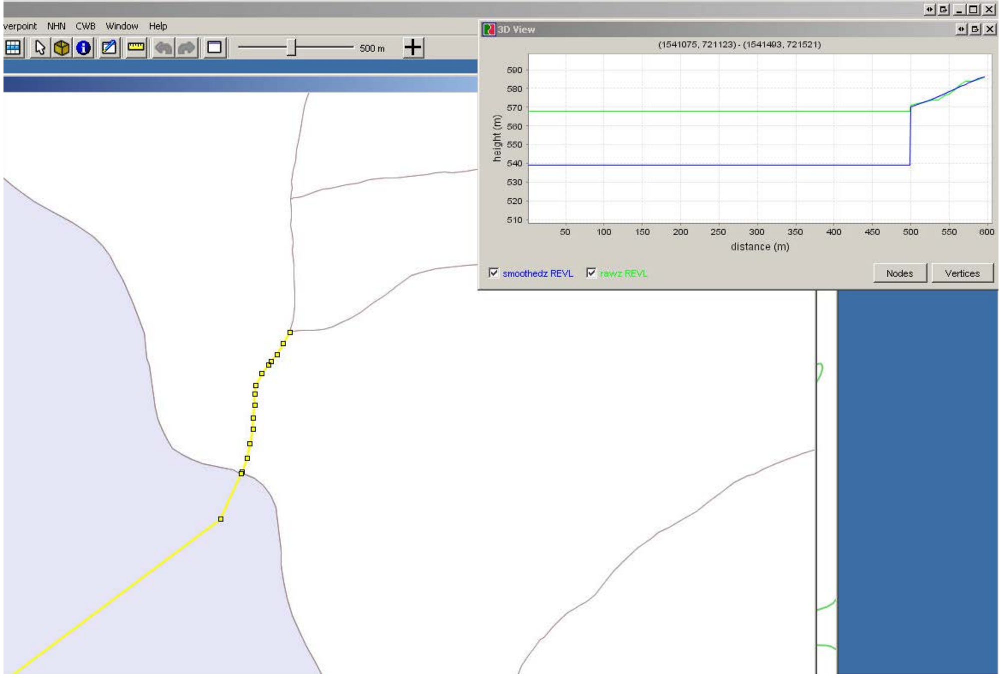
The images below show the effect of smoothing on some stair-stepped edges.





5.7 SNAP WETLAND CONSTRUCTION LINES

Wetland Construction lines (1200, 1300, 1400) and Inferred lines (1410, 1425) are not monotonized/smoothed, since they cannot be assumed to flow downhill. Instead, their raw elevation values (determined via the conflation/draping process) are used as (copied to) the refined elevation. In order to ensure vertical connectivity in the network, where necessary their node elevations are snapped to adjacent previously computed refined elevations (e.g. on observed edges).



6 QUALITY ASSURANCE

This section describes Quality Assurance (QA) processing performed after Stream Elevation processing is complete.

6.1 ARTIFICIAL WATERFALLS

Artificial waterfalls is the term given to situations where a steep drop occurs in the refined elevation which is not present in the original raw elevation. They are caused by various situations, including:

- Errors in the source elevation data at edge nodes. If these errors are not removed by the elevation cleaning processes (spike removal), they can affect the elevations computed during Node Elevation Monotonizing.   Incorrectly low nodes may cause upstream nodes to also be lowered, which produces artificial waterfalls.

- Lake leveling of lakes with large variations in their raw elevation. Artificial waterfalls will occur at the downstream point of incoming stream edges when a lake has been lowered, and at the upstream point of outgoing stream edges if a lake has been raised.

- Fixing the coastline elevation. This can cause artificial waterfalls to be reported in situations where the raw TRIM coastline elevation was digitized to lie at the top of a cliff, rather than at the bottom.

Artificial waterfalls can occur at both lake banks and stream confluences.

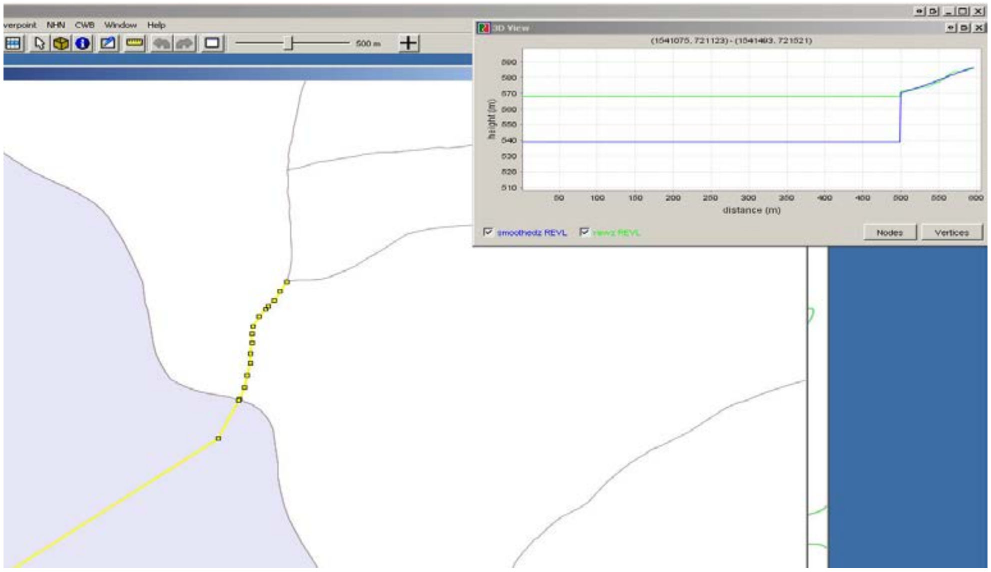


Figure 3: An example of an Artificial Waterfall

6.1.1 Detection Algorithm

The QA process involves running a detection algorithm over the entire stream network. The detection algorithm inspects the stream network elevations at every inflow/outflow of a lake and at every stream confluence. At each location, it calculates the delta between the slopes of the raw and refined elevation datasets. The raw slope is computed along a 5 meter section up or down the stream (depending on whether it is an inflow or outflow). For detection purposes, only locations with a raw slope of 40 % or less are considered (i.e. these are considered to be relatively flat stream reaches). The refined slope is computed on the first segment of the edge (since this is where the artificial waterfall will be introduced). Artificial waterfalls are considered to exist where the difference in refined and raw slope is 1.6 or greater. (This delta between the slopes eliminates flagging minor discrepancies.)

6.1.2 Manual Changes

Detected artificial waterfalls are manually investigated to determine the cause of the introduced waterfall. If necessary, manual fixes are performed. This may include rectifying errors in the original elevation input and/or adjusting some of the refined elevations. The expected result of this manual update is that most artificial waterfalls which are due to data errors will be eliminated (as opposed to artificial waterfalls introduced by lake leveling, which are an expected result of the chosen process).

6.1.3 Recording as Obstructions

Any remaining artificial waterfall locations are added as point locations to the Obstructions layer, with feature code (FCODE) of GA10450200.

6.2 DELETED WATERFALLS

Deleted Waterfalls is the term given to situations where a waterfall occurs in the raw elevation which does not occur in the refined elevation. Typically this occurs due to lake leveling and network monotonization. This process can result in a lake elevation being raised, which in turn may force inflows to be flattened so that there are no upstream vertices lower than the lake. This is in effect a "flooding" which can "submerge" existing waterfalls.

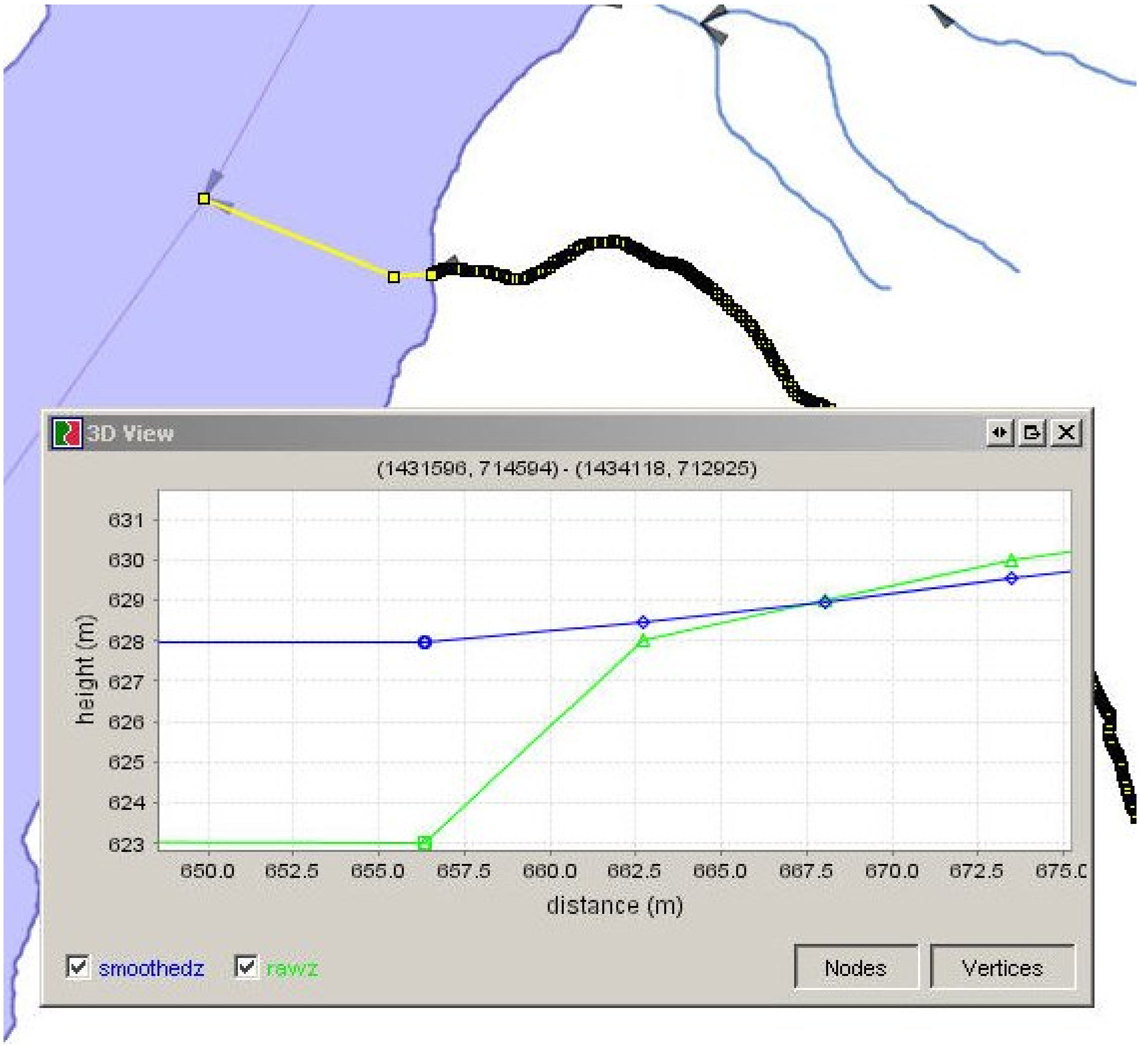


Figure 4: An example of a Deleted Waterfall

6.2.1 Detection Algorithm

The detection algorithm inspects the stream network elevations at every inflow/outflow of a lake and at every stream confluence. At each location, it calculates the delta between the slopes of the raw and refined elevation datasets. The raw slope is computed along a 5 meter section up or down the stream (depending on whether it is an inflow or outflow). Locations with a raw slope of 40 % or more are considered to be waterfalls. The refined slope is computed on the same section. A waterfall deletion is considered to exist where the refined slope is less than 0.1 (i.e. is relatively flat).

6.2.2 Recording as Obstructions

Deleted waterfall locations are added as point locations to the Obstructions layer, with feature code (FCODE) of GA10450300.

7 DATA MODEL

The SE Generation process produces 2 values for (almost) every vertex in the Freshwater Atlas database.

7.1 DATABASE

The Freshwater Atlas database stores the original (x,y) geometries as a PostGIS LINESTRING geometry object in the existing *cwb_edges* table. Additional dimensions are stored in a separate *cwb_edge_dimensions* table as arrays. These tables are linked by the *edge_id*.

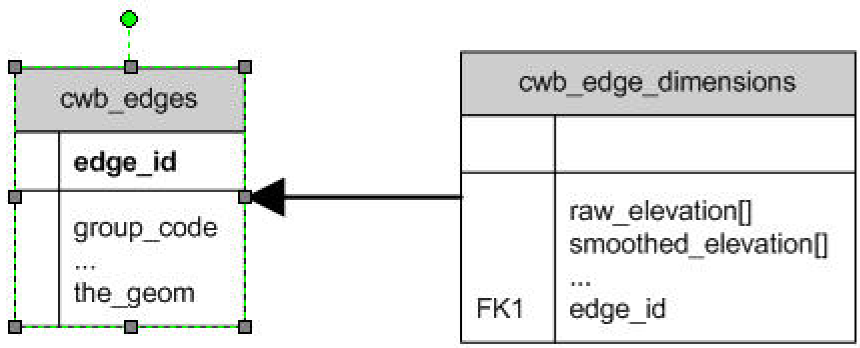


Figure 5: Database Storage Model

The following additional database functions are generated to aid in the analysis and viewing of multi-dimensional data:

Function	Description
adddimension(geometry, double[])	Create a 3-D geometry using a CWB 2D and either the raw or smoothed elevation arrays.
array_delta(double[])	Returns the delta between all adjacent values in an array.
array_fill(double[], double)	Initializes an array with the supplied value.
array_max(double[])	Finds the maximum value in an array.
array_min(double[])	Finds the minimum value in an array.
array_round(double[])	Rounds a supplied array to 5 decimal places.
array_set_min(double[], double)	Sets the minimum value in an array.

7.2 DELIVERY

The resultant SE Freshwater Atlas dataset is delivered as one complete 3D dataset:

1. Refined elevation (X, Y, Z’) on the stream network edges