Bay Name Matcher

Submitted By: Refractions Research Inc.
Suite 400 - 1207 Douglas Street
Victoria, BC, V8W-2E7
Phone: (250) 383-3022
Fax: (250) 383-2140

Table of Contents

Figures

Tables

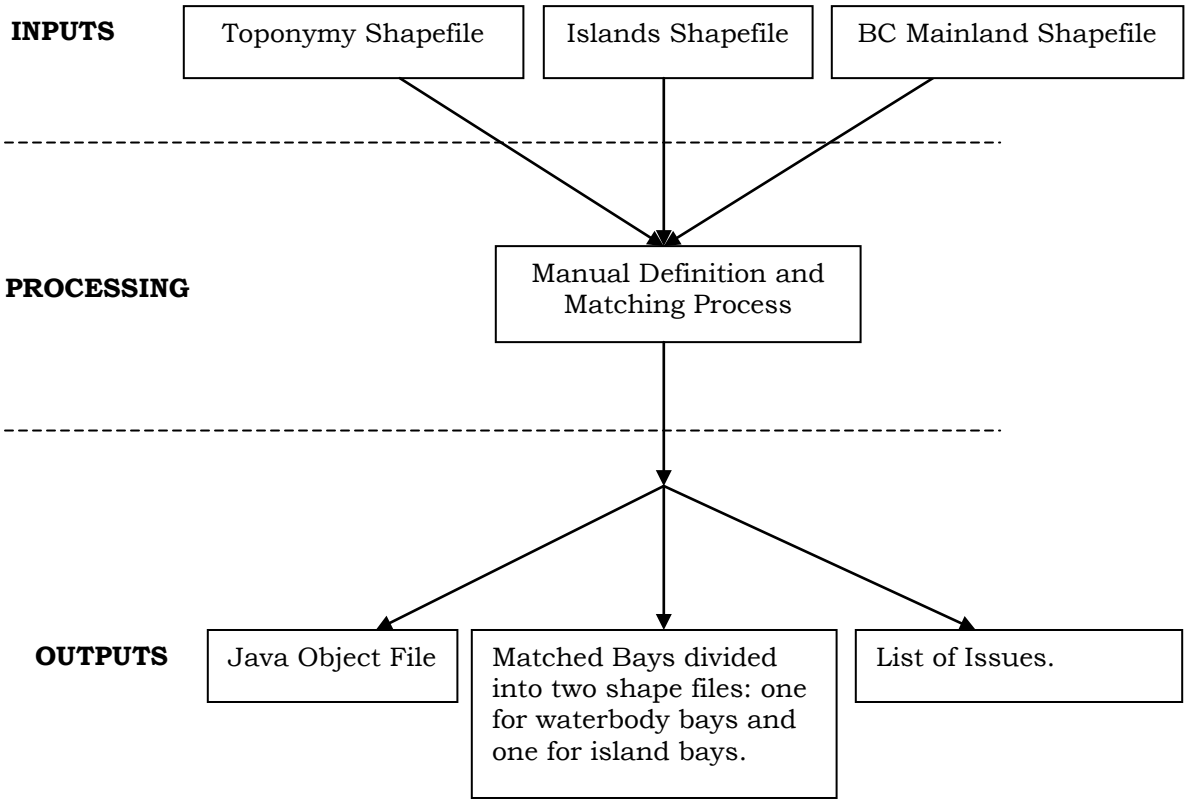
1 INTRODUCTION

The Bay Name Matcher (BNM) tool allows a user to define the extents of bays and match them to toponyms. This tool has no automated matching process; the user must define the extents of each bay.

2 BAY NAME MATCHER

The following figure shows the stages for the Bay Name Matcher process.

Figure 1: Bay Name Matcher Processes



2.1 INPUTS

The following tables describe the input files.

Table 1: Islands & BC Mainland Shapefiles

Field	Description
Shape	Polygon (in CW order).
Island Id	Unique id attached to each island polygon for identification purposes.
Island Type	0 – Saltwater Island. 1 – Freshwater Island. 2 – Saltwater & Freshwater Island (Example: Swishwashi Island).
Contained	ID of the island polygon that contains this polygon. Contained will be –1 if this island polygon is not contained within another island polygon. [This does not include the BC mainland polygon].

Table 2: Toponymy Shapefile

Field	Description
Point	Point (location of a letter).
TopoId	Toponym ID (all letters with the same toponym id form a toponym).
Elem_id	Position of letter in toponym name.
Letter	Letter associated with point.
Size	Size of the text.
Rotation	Rotation of the text.
GnisId	ID to link toponyms belonging to the same feature together.
Code	Code of the toponym: 1400 – Bay 1405 – Harbour 1407 – Cove 1411 – Inlet
Code_Cat	Category classification of toponym.

2.2 OUTPUTS

The following tables describe the output files.

Table 3: Named Bays Shapefile

Field	Description
Shape	'Polygon'
Type	'Island Bay'
Name	Name matched to point.
Gnisid	GnisId matched to point.
QAQC	QAQC flags associated with the match.

Table 4: Waterbody Bays Shapefile

Field	Description
Shape	'LineString
Type	'Waterbody Bay'
Name	Name matched to point.
Gnisid	GnisId matched to point.
QAQC	QAQC flags associated with the match.

A comma-delimited file (.csv) will be produced of all the toponyms flagged with an issue by the user.

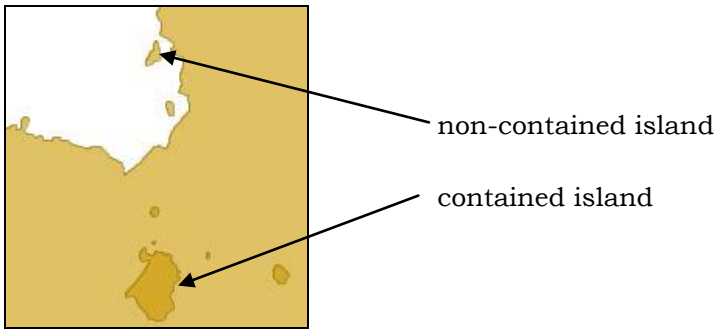
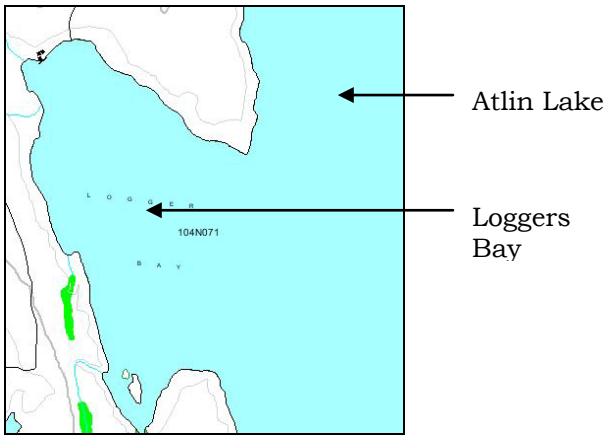
Table 5: List of Issues

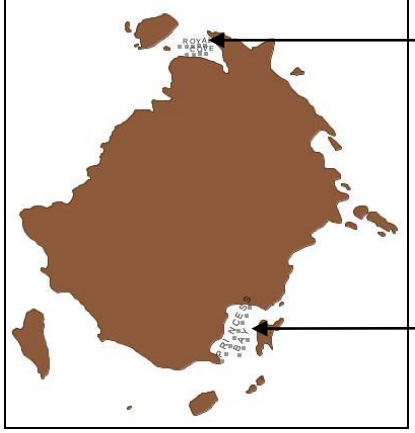
Field	Description
Toponym (Name)	Name of toponym to be split.
GNIS ID	Gnis ID for toponym to be split.
TOPO ID	Topo ID of toponym to be split.
COMMENTS	Comments about the issue entered by the user during the verification stage.

3 MATCHING ALGORITHM

3.1 Definitions

Table 6: Definitions

Toponym	A name on a map representing a feature. *Example*: "Vancouver Island"
TopoID	An island feature can have more than one toponym. Each toponym is assigned a unique ID. *Example*: "Vancouver Island" may appear on a map 10 times. Each of these toponyms would be assigned a separate, unique TopoID.
GnisID	All toponyms that represent the same feature are assigned the same GnisID. *Example*: The 10 "Vancouver Island" toponyms would all be assigned the same GnisID.
Contained Island	An island is contained if it exists inside another island (within a river, lake, or other water feature). *Example*: An island that is inside a lake on Vancouver Island would be a contained island. An island inside of a lake on the bc mainland would not be a contained island. 
Waterbody Bay	A bay feature that is defined by a lake, river, or other waterbody.  'Logger Bay' is a Waterbody Bay as it is defined by a waterbody (Atlin Lake) and not the island.
Island Bay	A bay feature defined by an island.



3.2 Bay Generation

3.2.1 Waterbody Bays

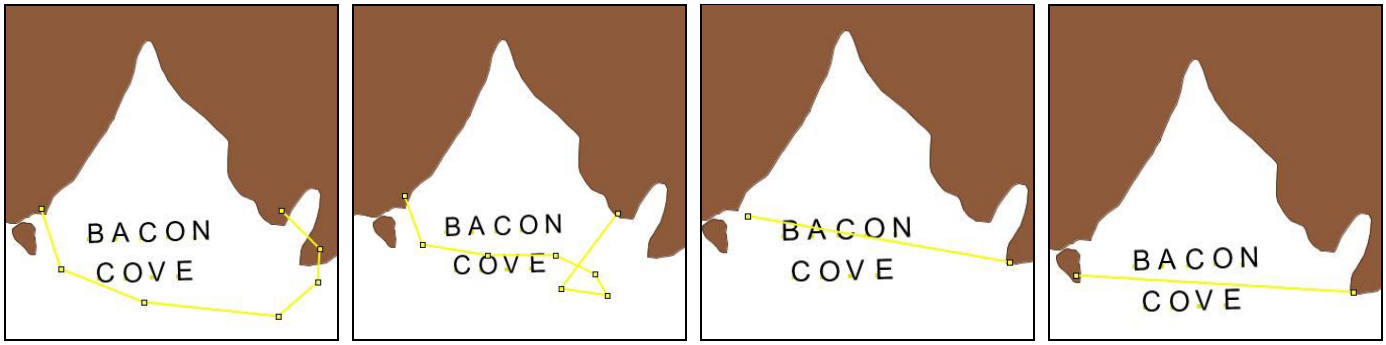
Waterbody bays are treated differently from island bays because the stream and lake datasets are not available (due to the size of the datasets) during the matching process to determine the bay polygon. Thus, the boundary line of the bay is drawn by the user using the WMS layers, but the actual bay polygon is not computed. A record is made of the boundary line and associated toponym and the bay polygons will be calculated at a later time.

3.2.2 Island Bays

For island bays the actual bay polygon is computed. The user draws the bay boundary and then using the island dataset the bay polygon is automatically generated. The bay boundary line must conform to the following five rules:

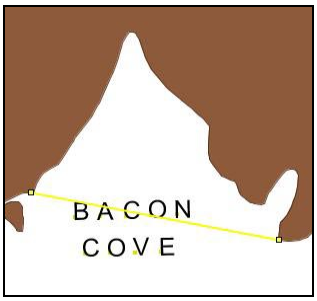
1. The start and end points of the line must intersect the same island ('main' island).
2. The line cannot cross itself.
3. The line must intersect the 'main' island exactly twice.
4. The start and end points must intersect only one island (the exception is the large polygon representing the BC mainland as shown in the two figures below).

Figure 2: Incorrect Bay Polygon



The line intersects (crosses) the 'main' island at four points.	The line intersects itself.	Both the start and end points do not intersect an island	The start and end points do not intersect the same island

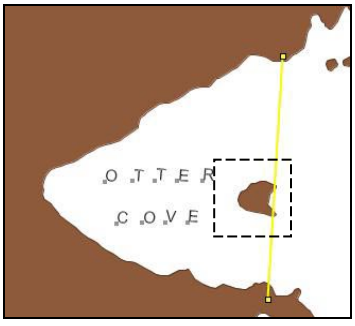
Figure 3: Correct Bay Polygon

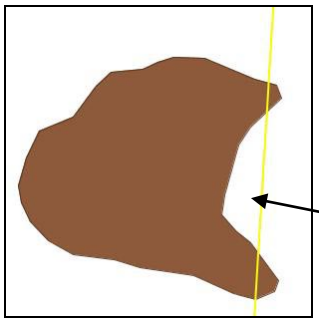


This line has both the start and end point intersecting the same island. The island is intersected by the line exactly twice and the line does not intersect itself.

5. The boundary line cannot intersect an island that results in multi polygons being created when that island is subtracted from the area of the bay.

Figure 4: Incorrect Boundary Line



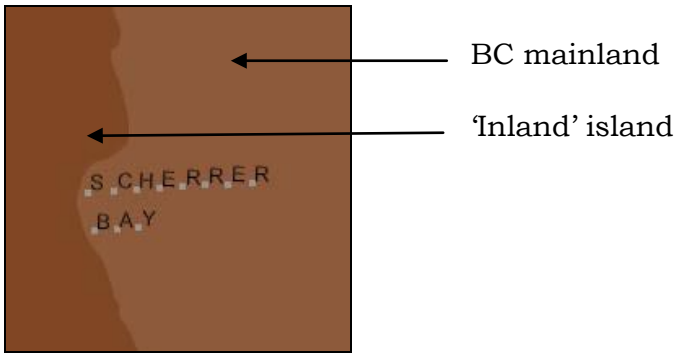


This small island is getting cut in such a way that two polygons are produced when the area of this little island is subtracted from the area of the bay. The bigger bay polygon will produced, but so will a small polygon defined by the area between the small island and the boundary line. This will produce an error.

3.2.3 Process for Determining Bay Boundary

1. The island intersecting the start point of the boundary line is found. A check is done to ensure the island only intersects one island polygon (see exception below). The same is done for the end point of the boundary line.
   - The *exception* to the intersection rule is the large polygon representing BC mainland. To allow inland bays defined by an island to be generated automatically, if a start/end point intersects two islands and one is the BC mainland polygon then the start/end point is allowed to intersect two islands. When determining which island is being intersected the island polygon which is NOT the BC mainland polygon is used.

Figure 5: Bay Boundary Determination

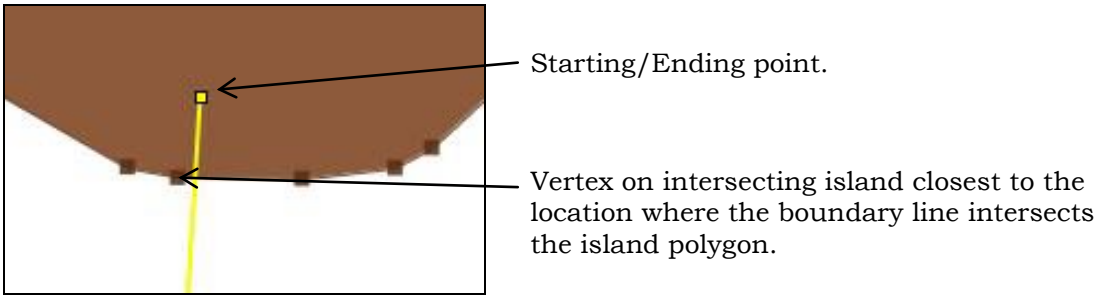


2. Verify that both the start and end intersecting islands are the same.

3. Verify the boundary line intersects this island at EXACTLY two points, and that the boundary line does not intersect itself.

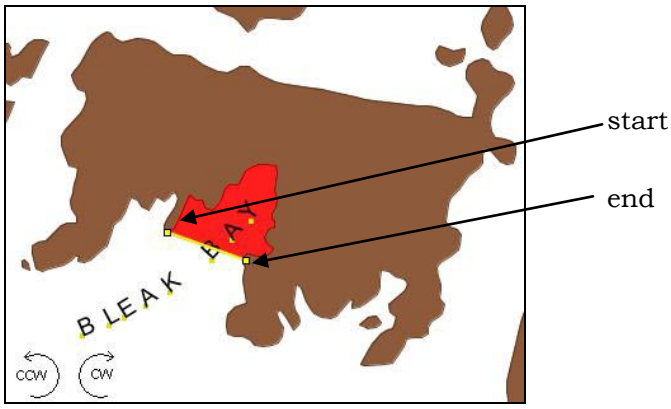
4. Find the index of the island vertex closest to the location where the boundary line intersects the island polygon. This result in two indexes, one for the start point and one for the end point of the boundary line.

Figure 6: Vertex on Intersecting Island



5. The outlining path for the bay polygon can be defined by either walking counter-clockwise or clockwise around the polygon. The actual algorithm used depends on the number of vertices on the island [two different algorithms are used as walking around each edge of a large island can be slow].
   - *For islands with a large number of vertices*: the bay outline is determined by finding the path from the start index to the end index with the least number of vertices;
     - Assuming all islands are generated in a clockwise order this is easy to compute. If the start index is 1200, the end index is 500, and the total number of vertices is 3000, then the shortest path would be in the counter-clockwise direction from vertex 1200 to vertex 500 (total of 700 vertices).
   - *For island with a small number of vertices*: the bay outline is determined by walking around the island in both CW and CCW direction and finding the shortest path (based on distance).
     - For example, the island in the figure below is considered a small island. The algorithm begins at the start point and walks around the vertex in both the CW and CCW direction. When one path reaches the end point and the other path has a greater total distance traveled, the shortest path is found. In this example the shortest path is in the clockwise direction.

Figure 7: Outlining Path for Bay Polygon

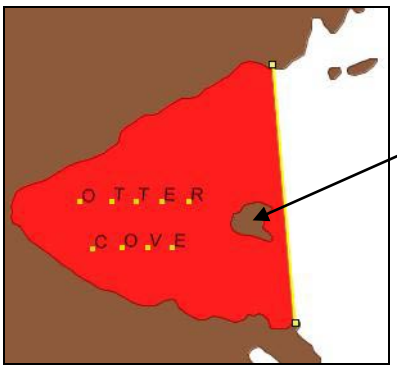


6. The difference between the boundary line and the island is calculated to produce a second boundary line.

7. A polygon is created from the path around the island (step 5) and the new boundary line (step 6).

8. All islands that intersect or are within in bay created in the previous step are subtracted from the bay area (so the bay will have holes where the island exists). If a multi-polygon is produced as a result of this process (see example above), then an error message is produced, otherwise the bay polygon has been generated and validated.

Figure 8: Island Intersect in Bay



This island is contained within the bay polygon, thus it is removed to produce a bay polygon with a hole in it where the island exists (right).

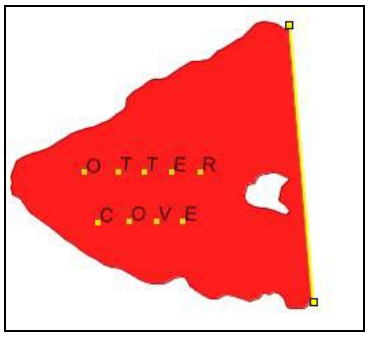


Figure 9: Process Flow