

## 7.7 Rasterize

### 7.7.1 Polygons to raster

#### Functionality

The Polygons to raster operation rasterizes a polygon map. The output raster map always uses the same domain as the input polygon map. This means that the class names, IDs, or values used in the polygon map are also used in the raster map.

For the output raster map, an existing georeference has to be selected or a new one can be created. The georeference determines the number of lines and columns of the output map and the pixel size of the map. It is strongly advised that vector maps of the same area are rasterized on the same georeference: any map calculation or spatial operation performed later on a combination of raster maps will only make sense if the pixels in these maps refer to the same area on the ground.

When a polygon map has an attribute table or when the domain of the polygon map has an attribute table, the Polygons to raster operation automatically links this attribute table to the output raster map.

#### Input map requirements

No special input map requirements, but at the moment, it is not yet possible to rasterize polygon maps using a Value domain that would result raster map with a store type real. If you do have a domain Value polygon map, it is advised to set the value range in such a way that the raster map can be stored with 1, 2, or 4 bytes per pixel.

#### Domain and georeference of output map

The output raster map uses the same domain as the input polygon map. The georeference for the output map has to be selected or created; you can usually select an existing georeference corners. The georeference for the raster map must use the same coordinate system as the polygon map. Georeference None cannot be selected for the output map.

---

☞ To obtain frequency information on polygons, you can calculate the histogram of a polygon map with the Histogram operation.

---

#### Dialog box

The Polygons to raster operation rasterizes a polygon map. The class names, IDs, or values in the polygon map are also used in the raster map, i.e. the domain of the polygon map is also the domain of the raster map. The user has to select or create a georeference for the output raster map.

**Dialog box options**

Input polygon map:	Select an input polygon map. Open the list box and select the desired input map, or drag a polygon map directly from the Catalog into this box.
Output raster map:	Type a name for the output raster map that will contain the rasterized polygons.
Georeference:	Select an existing georeference for the output raster map; open the list box by clicking it. Or create a new georeference by clicking the little create button  . You can usually select an existing georeference corners.
Value range:	In case the output map uses a value domain, accept the default value range, or specify your own range of possible values in the output map.
Precision:	In case the output map uses a value domain, accept the default precision of output values, or specify your own precision.
Show:	Select this check box if you want the output map to be displayed in a map window when the operation has finished. Clear this check box if you do not want to see this map immediately: you simply define how the output map should be created.
Description:	Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed.

A dependent output map is created.

**Command line**

The Polygons to raster operation can be directly executed by typing the following expression on the command line of the Main window:

```
OUTMAP = MapRasterizePolygon(InputPolygonMapName, Georeference)
```

where:

OUTMAP	is the name of your output raster map.
MapRasterizePolygon	is the command to start the Polygons to raster operation.
InputPolygonMapName	is the name of your input polygon map.
Georeference	is the name of an existing georeference. At the moment, it is not yet possible to create a new georeference on the command line.

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated.

## Algorithm

The Polygons to raster operation rasterizes a polygon map. The output raster map always uses the same domain as the input segment map. The user has to specify the georeference for the output raster map.

### Process

For each polygon, the corresponding pixels in the raster map are found; these pixels are assigned the class name, ID or value of the polygon. Other pixels obtain the undefined value.

- The pixel size indicated in the georeference for the output map, has a large influence on the size (in bytes) of the output raster map. Mind that rasterizing on a pixel size of 10 m instead of 50 m increases the size of the raster map by a factor 25.
- If you want to rasterize only a portion of your polygon map, you can select or create a georeference that does not cover the total area of your polygon map during rasterization.

### For ILWIS 1.4 users

- During rasterization, an .INF table is not produced anymore. To know the area of polygons, length of segments, nr. of points etc., you can calculate the histogram of a polygon, segment or point map with the Histogram operation.

## 7.7.2 Segments to raster

### Functionality

The Segments to raster operation rasterizes a segment map. The output raster map always uses the same domain as the input segment map. This means that the class names, IDs, or values used in the segment map are also used in the raster map.

For the output raster map, an existing georeference has to be selected or a new one can be created. The georeference determines the number of lines and columns of the output map and the pixel size of the map. It is strongly advised that vector maps of the same area are rasterized on the same georeference: any map calculation or spatial operation performed later on a combination of raster maps will only make sense if the pixels in these maps refer to the same area on the ground.

When a segment map has an attribute table or when the domain of the segment map has an attribute table, the Segments to raster operation automatically links this attribute table to the output raster map.

---

☞ When you want to create a Digital Elevation Model, you can directly do the operation Contour interpolation. Contour Interpolation first rasterizes your segment map, then calculates interpolated values for the pixels in between the contour lines.

---

**Input map requirements**

No special input map requirements.

**Domain and georeference of output map**

The output raster map uses the same domain as the input segment map. The georeference for the output map has to be selected or created; you can usually select an existing georeference corners. The georeference for the raster map must use the same coordinate system as the segment map. Georeference None cannot be selected for the output map.

- 
- ☞ To rasterize only segments with a specific code, you can first use the Mask segments operation.
  - ☞ When you want to print vector maps for annotation purposes on top of raster maps, print quality will improve if you use the vector maps as they are, thus without rasterizing them.
  - ☞ To obtain frequency information on segments, you can calculate the histogram of segment map with the Histogram operation.
- 

**Dialog box**

The Segments to raster operation rasterizes a segment map. The class names, IDs, or values in the segment map are also used in the raster map, i.e. the domain of the segment map is also the domain of the raster map. The user has to select or create a georeference for the output raster map.

**Dialog box options**

- Input segment map: Select an input segment map. Open the list box and select the desired input map, or drag a segment map directly from the Catalog into this box.
- Output raster map: Type a name for the output raster map that will contain the rasterized segments.
- Georeference: Select an existing georeference for the output raster map; open the list box by clicking it. Or create a new georeference by clicking the little create button . You can usually select an existing georeference corners.
- Value range: In case the output map uses a value domain, accept the default value range, or specify your own range of possible values in the output map.
- Precision: In case the output map uses a value domain, accept the default precision of output values, or specify your own precision.
- Show: Select this check box if you want the output map to be displayed in a map window when the operation has finished. Clear this check box if you do not want to see this map immediately: you simply define how the output map should be created.

Description: Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed.

A dependent output map is created.

- 
- ☞ If you want to create a Digital Elevation Model, you can directly do the operation Contour Interpolation. Contour Interpolation first rasterizes your segment map, then calculates interpolated values for the pixels in between the contour lines.
- 

## Command line

The Segments to raster operation can be directly executed by typing the following expression on the command line of the Main window:

```
OUTMAP = MapRasterizeSegment(InputSegmentMapName, Georeference)
```

where:

OUTMAP	is the name of your output raster map.
MapRasterizeSegment	is the command to start the Segments to raster operation.
InputSegmentMapName	is the name of your input segment map.
Georeference	is the name of an existing georeference. At the moment, it is not yet possible to create a new georeference on the command line.

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated.

## Algorithm

The Segments to raster operation rasterizes a segment map. The output raster map always uses the same domain as the input segment map. The user has to specify the georeference for the output raster map.

### Process

For each segment, the corresponding pixels in the raster map are found; these pixels are assigned the class name, ID or value of the segment. Other pixels obtain the undefined value.

- The specified pixel size in the georeference for the output map, has a large influence on the size (in bytes) of the output raster map. Mind that rasterizing on a pixel size of 10 m instead of 50 m increases the size of the raster map by a factor 25.
- If you want to rasterize only a portion of your segment map, you can either select or create a georeference that does not cover the total area of your segment map during rasterization, or you can first create a SubMap of the segment map and then rasterize this segment submap.

**For ILWIS 1.4 users**

- During rasterization, an .INF table is not produced anymore. To know the area of polygons, length of segments, nr. of points etc., you can calculate the histogram of a polygon, segment or point map with the Histogram operation.

### 7.7.3 Segment density

#### Functionality

The Segment Density operation, also called Lineament Density, rasterizes a segment map. For each output pixel, the total length of all segment parts within the boundaries of the pixel is summed: this is the output value for the pixel.

During this operation, you have to select or create a georeference. It is advised to use a georeference in which the pixels are rather large, i.e. at least large enough to contain parts of several segments. When you choose a large pixel size, the segments are aggregated over a larger area. Further, by using a mask, you can specify the elements of the input map that are to be used in the calculation.

**Example**

When you have a segment map with geological faults, you can calculate the length of faults per square km, per 100 square km, etc. Similarly, you can calculate the length of roads or rivers per area.

**Mask**

By specifying a mask, you can search for segments with a certain class name, ID or value. Then only these segments will be used in the calculation of the output map. A mask is in fact a simple query to find (retrieve), display or use only those segments in a map that have certain class names, identifiers or values. A mask contains one or more search strings in which wild cards, asterisks (\*) and question marks (?), are allowed. You can enter as many search strings in a mask as you like; separate the strings by commas.

For more information on masks, see How to use masks.

**Requirements for the input map**

No special requirements for the input segment map.

**Domain and georeference of output map**

The default domain for the output raster map is the system domain Value but you can also select another domain or create a new domain. The georeference for the output map has to be selected or created. When this is the first time you aggregate pixels, you should create a new georeference corners in which the pixels are large enough to contain a number of segments. The georeference for the raster map may use a coordinate system that is different from coordinate system of the input point map but a conversion between both coordinate systems must be possible.

Georeference None cannot be selected for the output map.

- 
- ☞ The Aggregate raster map operation has similar functionality for raster maps and contains a number of different aggregation functions.
- 

## Dialog box

The Segment Density operation rasterizes a segment map. For each output pixel, the total length of all segment parts within the boundaries of the pixel is summed: this is the output value for the pixel. By using a mask, you can specify the elements of the input map that are to be used in the calculation.

### Dialog box options

- Input segment map:** Select an input segment map. Open the list box and select the desired input map, or drag a map directly from the Catalog into this box. No special input map requirements.
- Mask:** Type (multiple) search strings to find the segments in the input map that you want to use in the calculation. If all segments should be taken into account, accept the default mask \*. For more information on masks, see Mask segments : functionality or How to use masks.
- Output raster map:** Type a name for the output raster map that will contain the summed lengths of segments found in each output pixel.
- Georeference:** Select an existing georeference or create  a new georeference corners.
- Domain:** Accept the default domain Value, or select an existing value domain, or create a new value domain.
- Value range:** Accept the default value range for the output map, or specify the value range yourself.
- Precision:** Accept the default precision for the output map, or specify the precision yourself.
- Show:** Select this check box if you want the output map to be displayed in a map window when the operation has finished. Clear this check box if you do not want to see this map immediately: you simply define how the output map should be created.
- Description:** Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed. If you do not specify a description, the expression by which the output map is created will be used.

A dependent output map is created.

## Command line

The Segment density operation can be directly executed by typing the following expression on the command line of the Main window:

```
OUTMAP = MapSegmentDensity(InputSegmentMap, "mask",Georeference)
```

where:	
OUTMAP	is the name of your output raster map.
MapSegmentDensity	is the command to start the Segment Density operation.
InputSegmentMap	is the name of the input segment map.
"mask"	are the (multiple) search strings to find the classes, IDs or values in the input map that you want to use in the calculation of the output map; on the command line, the total mask needs to be surrounded by double quotes.
Georeference	is the name of an existing georeference that should be used for the output raster map.

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated.

## Algorithm

The Segment density operation rasterizes an input segment map on a pixel size which is specified in the georeference.

### Process

All pixels are initialized on value zero. For each segment, all pixels are found that are crossed by the segment. For all these pixels, the length of the segment parts within the pixel is added to the current value of the pixel.

## 7.7.4 Points to raster

### Functionality

The Points to raster operation rasterizes a point map. The output raster map always uses the same domain as the input point map. This means that the class names, IDs, or values used in the point map are also used in the raster map.

For the output raster map, an existing georeference has to be selected or a new one can be created. The georeference determines the number of lines and columns of the output map and the pixel size of the raster map. It is strongly advised that vector maps of the same area are rasterized on the same georeference: any map calculation or spatial operation performed later on a combination of raster maps will only make sense if the pixels in these maps refer to the same area on the ground.

Normally, each point will be represented by 1 pixel in the output raster map. As you may have stored information as points that are actually referring to areas, the operation has an option to indicate by how many pixels a point should be represented. When a point size of 1 is specified (default), it means that each point is represented by 1 pixel in the output raster map; when a point size of 2 is specified, it means that each point is represented by 2\*2 pixels in the output raster map, etc.

When points are rasterized with a small point size, it can be difficult to see the rasterized points when the output map is displayed. You can always display the original point map on top of the raster map.

- 
- ☞ To analyze the surroundings of points, you can rasterize points with a point size larger than one, and then perform a Cross operation.
- 

When a point map has an attribute table or when the domain of the point map has an attribute table, the Points to raster operation automatically links this attribute table to the output raster map.

#### **Input map requirements**

No special input map requirements.

#### **Domain and georeference of output map**

The output raster map uses the same domain as the input point map. The georeference for the output map has to be selected or created; you can usually select an existing georeference corners. The georeference for the raster map must use the same coordinate system as the point map. Georeference None cannot be selected for the output map.

- 
- ☞ The Point density operation counts the number of points that are located in each pixel.
  - ☞ Interpolations on point values can be performed with the Moving average, Trend surface and Moving surface operations.
  - ☞ To rasterize only points with a specific code, you can first use the Mask points operation.
  - ☞ When you want to print vector maps for annotation purposes on top of raster maps, print quality will improve if you use the vector maps as they are, thus without rasterizing them.
  - ☞ To obtain frequency information on points, you can calculate the histogram of a point map with the Histogram operation.
- 

## **Dialog box**

The Points to raster operation rasterizes a point map. The class names, IDs, or values in the point map are also used in the raster map; i.e. the domain of the point map is also the domain of the raster map. The user has to select or create a georeference for the output raster map.

#### **Dialog box options**

Input point map:	Select an input point map. Open the list box and select the desired input map, or drag a point map directly from the Catalog into this box. No special input map requirements.
Point size:	Type a value (integer > 0) for the size of the points in pixels in the output map. Point size 1 means each point is

	represented by 1 pixel in the output map. Point size 2 means that each point is represented by 2*2 pixels in the output map.
Output raster map:	Type a name for the output raster map that will contain the rasterized points.
Georeference:	Select an existing georeference for the output raster map; open the list box by clicking it. Or create a new georeference by clicking the little create button  . You can usually select an existing georeference corners.
Show:	Select this check box if you want the output map to be displayed in a map window when the operation has finished. Clear this check box if you do not want to see this map immediately: you simply define how the output map should be created.
Description:	Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed.

A dependent output map is created.

## Command line

The Points to raster operation can be directly executed by typing the following expression on the command line of the Main window:

```
OUTMAP = MapRasterizePoint(InputPointMapName, Georeference, Point size)
```

where:

OUTMAP	is the name of your output raster map.
MapRasterizePoint	is the command to start the Points to raster operation.
InputPointMapName	is the name of your input point map.
Georeference	is the name of an existing georeference that should be used for the output map.
<i>Point size</i>	is the size of the points in pixels in the output map (integer > 0). Point size 1 means each point is represented by 1 pixel in the output map. Point size 2 means that each point is represented by 2*2 pixels in the output map.

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated.

## Algorithm

The Points to raster operation rasterizes a point map. The output raster map always uses the same domain as the input point map. The user has to specify the georeference for the output raster map.

**Process**

If the chosen point size is 1, then for each point, the corresponding pixel in the raster map is found; this pixel is assigned the class name, ID or value of the point. Other pixels obtain the undefined value. If the chosen point size is larger than 1, then a square is drawn around each point with the specified size in pixels. All pixels inside this box are assigned the class name, ID or value of the point. Other pixels obtain the undefined value.

- The pixel size indicated in the georeference for the output map, has a large influence on the size (in bytes) of the output raster map. Mind that rasterizing on a pixel size of 10 m instead of 50 m increases the size of the raster map by a factor 25.
- If you want to rasterize only a portion of your point map, you can either select or create a georeference that does not cover the total area of your point map during rasterization, or you can first create a SubMap of the point map and then rasterize this submap.

**For ILWIS 1.4 users**

- During rasterization, an .INF table is not produced anymore. To know the area of polygons, length of segments, nr. of points etc., you can calculate the histogram of a polygon, segment or point map with the Histogram operation.
- To rasterize attribute values of a point map, first create an attribute map of the point map with the Attribute map of point map operation, then rasterize this attribute point map.
- When rasterizing a value point map, it is not yet possible to sum the values of points that would fall in one pixel.

### 7.7.5 Point density

**Functionality**

The Point Density operation rasterizes a point map. For each output pixel, the number of points located in that pixel is counted: this is the output value for the pixel. This operation can be used to examine the regional distribution of points.

In the dialog box of the operation, you have to select or create a georeference. It is advised to use a georeference in which the pixels are rather large, i.e. at least large enough to contain several points. When you choose a large pixel size, the points are aggregated over a larger area.

Further, you have to specify the point size for the output map. Point size 1 means that each point is counted in 1 pixel in the output map. Point size 2 means that each point is counted in 2\*2 pixels in the output map, etc. When you use a point size larger than 1, the output raster map will have a smoother appearance.

**Example**

Suppose you have a point map in which each point represents one incidence of a forest-fire or an illness, etc. You can use the Point density operation to aggregate these incidences on square kilometers, on hectares, etc. This gives you quantitative information on the incidences per area.

**Input map requirements**

No special requirements for the input point map.

**Domain and georeference of output map**

The default domain for the output raster map is the system domain Count but you can also select another domain or create a new domain. The georeference for the output map has to be selected or created. When this is the first time you aggregate pixels, you should create a new georeference corners in which the pixels are large enough to contain a number of points. The georeference for the raster map may use a coordinate system that is different from coordinate system of the input point map but a conversion between both coordinate systems must be possible. Georeference None cannot be selected for the output map.

---

☞ The Aggregate map operation has similar functionality for raster maps and contains a number of different aggregation functions.

---

**Dialog box**

The Point Density operation rasterizes a point map. For each output pixel, the number of points located in that pixel is counted: this is the output value for the pixel. This operation can be used to examine the regional distribution of points.

**Dialog box options**

Input point map:	Select an input point map. Open the list box and select the desired input map, or drag a map directly from the Catalog into this box. No special input map requirements.
Point size:	Type a value (integer > 0) for the size of the points in pixels in the output map. Point size 1 means each point is counted in 1 pixel in the output map. Point size 2 means that each point is counted in 2 x 2 pixels in the output map.
Output raster map:	Type a name for the output raster map that will contain the counted number of points found in each output pixel.
Georeference:	Select an existing georeference or create  a new georeference corners.
Domain:	Accept the default domain Count, or select an existing value domain, or create a new value domain.
Value range:	Accept the default value range for the output map, or specify the value range yourself.
Precision:	Accept the default precision for the output map, or specify the precision yourself.

- Show: Select this check box if you want the output map to be displayed in a map window when the operation has finished. Clear this check box if you do not want to see this map immediately: you simply define how the output map should be created.
- Description: Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed. If you do not specify a description, the expression by which the output map is created will be used.

A dependent output map is created.

## Command line

The Point density operation can be directly executed by typing the following expression on the command line of the Main window:

```
OUTMAP = MapRasterizePointCount(InputPointMap, Georeference,
                                PointSize)
```

where:

- |                        |  |
|------------------------|--|
| OUTMAP                 | is the name of your output raster map.   |
| MapRasterizePointCount | is the command to start the Point Density operation.   |
| InputPointMap          | is the name of the input point map.  |
| Georeference           | is the name of an existing georeference that should be used for the output raster map.   |
| PointSize              | is the point size (integer value > 0) of the output map. Point size 1 means each point is counted in 1 pixel in the output map. Point size 2 means that each point is counted in 2*2 pixels in the output map. |

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated.

## Algorithm

The Point density operation rasterizes a point map on a pixel size which is specified in the georeference.

### Process

All pixels are initialized on value zero. For every point in the input point map, it is determined in which output pixel it is located. The value of an output pixel is incremented by 1 for each point that is found in that pixel. If no points are found in a pixel and you used the default domain Count, the pixel obtains value 0.

## 7.8 Vectorize

### 7.8.1 Raster to polygons

#### Functionality

The Raster to Polygons operation extracts polygons from units in a raster map. The output polygon map uses the same domain as the input raster map, i.e. the class names, IDs, or values in the input raster map will also be used for the polygons in the output polygon map. No polygons are created for pixels with the undefined value.

The polygons in the output map are derived from areas of pixels that have the same value, class name or ID in the input raster map. You can choose to create polygons:

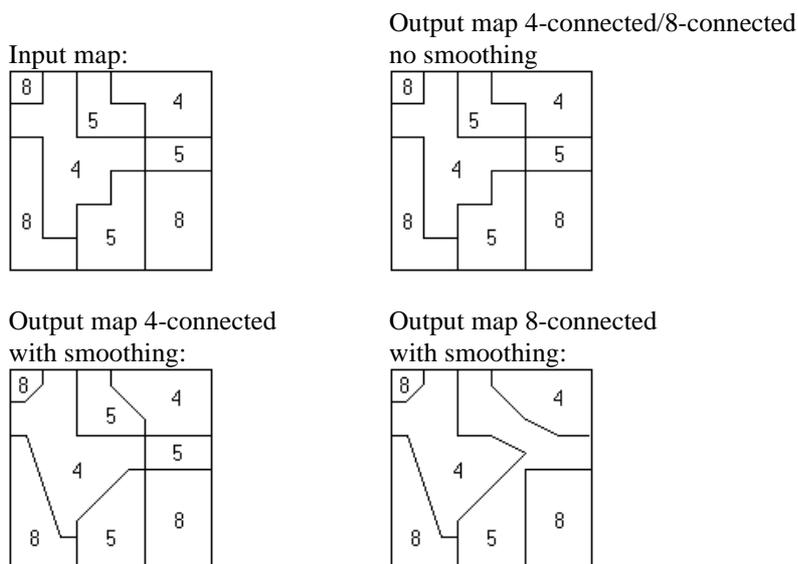
- from 4-connected pixels: areas of pixels are found where pixels with the same value, class name or ID are horizontally or vertically connected;
- from 8-connected pixels: areas of pixels are found where pixels with the same value, class name or ID are horizontally, vertically or diagonally connected.

For more information on constructing areas of 4 or 8-connected pixels, see Area numbering : functionality.

Furthermore, you can specify whether or not to smooth the boundaries of the output polygons.

#### Examples

The effect of using 4 or 8-connected pixels and smoothing or no smoothing is illustrated in the figures below.



The result of this operation depends on the homogeneity of the raster map and the pixel size compared to the size of the mapping units. It is advised to only use this operation on rather homogeneous raster maps that consist of areas with a considerable number of pixels. When you want to extract for instance polygons from a raster map which is the result of the Classify operation, it may be better to first run the majority filter on that raster map to homogenize the classification results and then perform the Raster to polygons operation.

When the input raster map has an attribute table or when the domain of the input raster map has an attribute table, the Raster to polygons operation automatically links this attribute table to the output polygon map.

#### **Input map requirements**

No special requirements regarding the domain of the input map.

The input raster map must have a georeference which is not georeference None.

#### **Domain and coordinate system of output map**

The output polygon map uses the same domain as the input raster map.

The output polygon map uses the same coordinate system as the georeference of the input raster map. The coordinate boundaries for the polygon map are the boundaries of this georeference.

## **Dialog box**

The Raster to Polygons operation extracts polygons from units in a raster map. The output polygon map uses the same domain as the input raster map, i.e. the class names, IDs, or values in the input raster map will also be used for the polygons in the output polygon map. No polygons are created for pixels with the undefined value.

#### **Dialog box options**

- |   |   |
|---|---|
| Input raster map:                             | Select an input raster map. Open the list box and select the desired input map, or drag a raster map directly from the Catalog into this box. The raster map must have a georeference which is not georeference None. |
| Connect:                                      | Specify to obtain polygons from 4-connected pixels or from 8-connected pixels.  |
| <input checked="" type="radio"/> 4-connected: | Find areas of pixels with the same value/name which are horizontally or vertically connected, then create polygons from these areas.  |
| <input checked="" type="radio"/> 8-connected: | Find areas of pixels with the same value/name which are horizontally, vertically or diagonally connected, then create polygons from these areas.  |
| Smooth lines:                                 | Select this check box if you want the boundaries of the output polygons to be smoothed. Clear this check box if the polygon boundaries should follow the exact boundaries of the pixel areas.                         |

Output polygon map: Type a name for the output polygon map that will contain the polygons extracted from units in the raster map.

Show: Select this box if you want the output map to be displayed in a map window when the operation has finished. Clear this box if you do not want to see this map immediately: you simply define how the output map should be created.

Description: Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed.

A dependent output map is created.

## Command line

The Raster to polygons operation can be directly executed by typing the following expression on the command line of the Main window:

```
OUTMAP = PolygonMapFromRas(InputRasterMapName)
```

```
OUTMAP = PolygonMapFromRas(InputRasterMapName, 8 | 4)
```

```
OUTMAP = PolygonMapFromRas(InputRasterMapName, 8 | 4,
    smooth | nosmooth)
```

where:

OUTMAP	is the name of your output polygon map.
PolygonMapFromRas	is the command to start the Raster to Polygons operation.
InputRasterMapName	is the name of your input raster map.
8   4	is an optional parameter to indicate whether to construct 8-connected 4-connected areas (either 8 or 4) from which polygons should be extracted; when this parameter is not specified, 8-connected areas will be used.
smooth   nosmooth	is an optional parameter to indicate whether the boundaries of the output polygons should be smoothed or not (type either <code>smooth</code> or <code>nosmooth</code> ); when this parameter is not specified, boundaries will be smoothed.

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated.

## Algorithm

The Raster to Polygons operation extracts polygons from units in a raster map. The output polygon map uses the same domain as the input raster map, i.e. the class names, IDs, or values in the input raster map will also be used for the polygons in the output polygon map.

First Area Numbering is applied on the input raster map. In case of 4-connectivity, the program compares the values of the 4 horizontal and vertical neighbouring pixels with the value of the center pixel, and in case of 8-connectivity, the program compares the values of all 8 neighbouring pixels (including the diagonal neighbours) with the value of the center pixel.

Then, polygons are initialized; a topological structure is built and stored; and the areas of the polygons are calculated.

No polygons are created for pixels with the undefined value.

## 7.8.2 Raster to segments

### Functionality

The Raster to segments operation extracts segments from the boundaries of mapping units in a raster map. The segments in the output map either obtain the code Segments or a special code which is a combination of the class names or IDs of the two mapping units found on either side of the segment.

The segments in the output map are derived from the boundaries of areas of pixels that have the same class name or ID in the input raster map. You can choose to create segments:

- from 4-connected pixels: areas of pixels are found where pixels with the same class name or ID are horizontally or vertically connected; or
- from 8-connected pixels: areas of pixels are found where pixels with the same class name or ID are horizontally, vertically or diagonally connected.

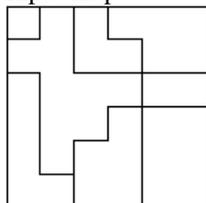
For more information on constructing areas of 4 or 8-connected pixels, see Area numbering : functionality.

Furthermore, you can specify whether or not to smooth the output segments.

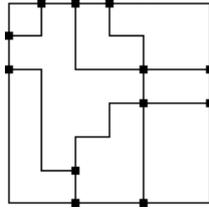
### Examples

The effect of using 4 or 8-connected pixels and smoothing or no smoothing is illustrated in the figures below.

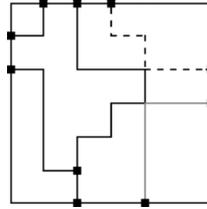
Input map:



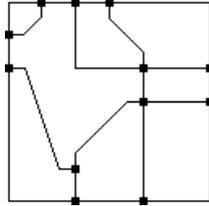
Output map 4-connected  
no smoothing:



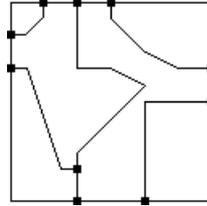
Output map 8-connected  
no smoothing:



Output map 4-connected  
with smoothing:



Output map 8-connected  
with smoothing:



Using 4- and 8-connectivity, the appearance of the output map looks the same, however in the case of 4-connectivity the output map has more segments. This is because only horizontal and vertical neighbours were evaluated to construct segments from the boundaries of the mapping units. In the case of 8-connected, horizontal, vertical and diagonal neighbours were used to construct segments. Note that in the output map the two highlighted segments (dark gray and dashed) 'touch' another segment (black) in the central part of the map without a node.

The result of this operation depends on the homogeneity of the raster map and the pixel size compared to the size of the mapping units. It is advised to only use this operation on rather homogeneous raster maps that consist of areas with a considerable number of pixels. When you want to extract for instance segments from a raster map which is the result of the Classify operation, it may be better to first run the majority filter on that raster map to homogenize the classification results and then perform the Raster to segments operation.

#### Input map requirements

When you use the Raster to Segments operation through the dialog box, you can use for the input raster map a map with a class, ID or Bool domain. On the command line, you can use any type of input map.

The input raster map must have a georeference which is not georeference None.

#### Domain and coordinate system of output map

The Raster to segments operation creates a new class domain for the output map when the input map also had a class domain; in other cases an Identifier domain is created. This output domain is always stored by the output map (internal domain).

When you select the option 'Single Name', all segments are simply named Segments.

When you do not select the option 'Single Name', each segment obtains a name which is a combination of the names of the neighbouring mapping units like neighbour1 | neighbour2 , neighbour1 | neighbour3, etc.

The output segment map uses the same coordinate system as the georeference of the input raster map. The coordinate boundaries for the segment map are the boundaries of this georeference.

- 
- ☞ To obtain frequency information on segments, you can calculate the histogram of the resulting segment map with the Histogram operation.
  - ☞ If you only want to extract segments from a small part of your raster map, you can first create a Submap of the raster map and then extract segments from the raster submap.
  - ☞ For more information on internal domains and representations, refer to How to open internal domains/representations.
- 

## Dialog box

The Raster to segments operation extracts segments from the boundaries of mapping units in a raster map. The segments in the output map either obtain the code Segments or a special code which is a combination of the class names or IDs of the two mapping units found on either side of the segment.

### Dialog box options

- |   |   |
|---|---|
| Input raster map:                             | Select an input raster map (map with a class, ID or a Bool domain). Open the list box and select the desired input map, or drag a raster map directly from the Catalog into this box. The raster map must have a georeference which is not georeference None. |
| Connect:                                      | Specify to obtain segments from 4-connected pixels or from 8-connected pixels.  |
| <input checked="" type="radio"/> 4-connected: | Find areas of pixels with the same name which are horizontally or vertically connected, then create segments from the boundaries of these areas.  |
| <input checked="" type="radio"/> 8-connected: | Find areas of pixels with the same name which are horizontally, vertically or diagonally connected, then create segments from the boundaries of these areas.  |
| Smooth lines:                                 | Select this check box if you want the output segments to be smoothed. Clear this check box if the segments should follow the exact boundaries of the pixel areas.   |
| Single name:                                  | Select this check box if you want that all segments obtain the same code Segments.<br>Clear this check box if each segment should obtain the code of the two mapping units on either side of each segment found.  |

- Output segment map: Type a name for the output segment map that will contain the segments which were extracted from the boundaries of mapping units in the raster map.
- Show: Select this box if you want the output map to be displayed in a map window when the operation has finished. Clear this box if you do not want to see this map immediately: you simply define how the output map should be created.
- Description: Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed.

A dependent output map is created.

## Command line

The Raster to segments operation can be directly executed by typing the following expression on the command line of the Main window:

```
OUTMAP = SegmentMapFromRasAreaBnd(InputRasterMapName, 8 | 4,
    smooth | nosmooth, single | composite)
```

where:

- |                          |  |
|--------------------------|--|
| OUTMAP                   | is the name of your output segment map.  |
| SegmentMapFromRasAreaBnd | is the command to start the Raster to segments operation.  |
| InputRasterMapName       | is the name of your input raster map.  |
| 8   4                    | is the parameter to indicate whether to construct 8-connected 4-connected areas (either 8 or 4) from which segments should be extracted.   |
| smooth   nosmooth        | is the parameter to indicate whether the output segments should be smoothed or not.  |
| single   composite       | is the parameter to indicate whether the output segments should obtain one name Segments or a name which is a combination of the names of the mapping units on either side of the segment. |

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated. Further, the codes for the output segments are stored by the output segment map (internal domain).

## Algorithm

The Raster to segments operation first finds areas of pixels in the input map that have the same class name, ID or value. Areas can be constructed from pixels that are either horizontally or vertically connected (4-connected) or horizontally,

vertically or diagonally connected (8-connected). For more information on constructing areas of 4 or 8-connected pixels, see Area numbering : functionality.

Then, from the boundaries of these areas, segments are extracted. The segments in the output map either obtain the code Segments or a special code which is a combination of the class names or IDs of the two mapping units found on either side of the segment like neighbour1|neighbour2, neighbour1|neighbour3, etc.

Finally, if specified, the segments in the output map will be smoothed.

### 7.8.3 Raster to points

#### Functionality

The Raster to points operation extracts a point from each pixel in the raster map. Each point gets the value, class name or ID of the corresponding pixel. No points are created for pixels with the undefined value.

---

☞ To temporarily view pixel values as text in a map window, you can select the Text check box in the Display Options dialog box of a raster map.

---

#### Input map requirements

The input raster map must have a georeference and this georeference cannot be georeference None.

#### Domain and coordinate system of output map

The output point map uses the same domain as the input raster map.

The output point map uses the same coordinate system as the input raster map; coordinate boundaries for the output point map are constructed from the georeference of the input raster map.

#### Dialog box

The Raster to points operation extracts a point from each pixel in the raster map. Each point gets the value, class name or ID of the corresponding pixel. No points are created for pixels with the undefined value.

#### Dialog box options

Input raster map:	Select an input raster map. Open the list box and select the desired input map, or drag a raster map directly from the Catalog into this box. The raster map must have a georeference which is not georeference None.
Output point map:	Type a name for the output point map that will contain the values of pixels as points.
Show:	Select this check box if you want the output map to be displayed in a map window when the operation has finished. Clear this check box if you do not want to see this map

immediately: you simply define how the output map should be created.

Description: Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed.

A dependent output map is created.

## Command line

The Raster to points operation can be directly executed by typing the following expression on the command line of the Main window:

```
OUTMAP = PointMapFromRas(InputMapName)
```

where:

OUTMAP is the name of your output point map.

PointMapFromRas is the command to start the Raster to points operation.

InputMapName is the name of the input raster map; this map cannot have georeference None.

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated.

## Algorithm

The Raster to points operation extracts a point from each pixel in the raster map. Each point gets the value, class name or ID of the corresponding pixel. The points are located at the centers of the pixels. No points are created for pixels with the undefined value.

## 7.8.4 Polygons to segments

### Functionality / Algorithm

The Polygons to segments operation extracts polygon boundaries and creates a segment map out of them. By default, the operation extracts segments from all polygon boundaries of the input map. You can also specify a mask to extract segments of specific polygons. The operation is useful when you want to edit/update boundaries of the polygons.

There are two possibilities to assign names to the extracted segments:

- single name: all extracted segments are named 'Segments';
- composite names: each segments is named according to its neighbouring polygons, e.g. if neighbouring polygons are 'Forest' and 'Urban', the output segment name will be 'Forest | Urban'.

#### Input map requirements

No special input map requirements.

#### Domain and coordinate system of output map

The operation creates a new domain which is stored by the output map (internal domain). When the input map has class domain, the output also has a class domain, otherwise the output map will use an ID domain. The output segment map uses the same coordinate system as the input polygon map.

- 
- 🔗 You can also extract segments in the polygon editor.
  - 🔗 For more information on internal domains and representations, refer to [How to open internal domains/representations](#).
- 

### Dialog box

The Polygons to segments operation extracts polygon boundaries and creates a segment map out of them. A mask can be specified to extract segments of specific polygons.

#### Dialog box options

Input polygon map:	Select an input polygon map. Open the list box and select the desired input map, or drag a polygon map directly from the Catalog into this box. No special input map requirements.
Mask:	Type (multiple) search strings to extract segments of specific polygons. For more information, see <a href="#">Mask Polygons : functionality and How to use masks</a> .
Single name:	Select this check box if you want to assign the single name 'Segments' to all extracted segments. Clear this check box to assign composite names.

Output segment map:	Type a name for the output segment map that will contain the extracted segments.
Show:	Select this check box if you want the output map to be displayed in a map window when the operation has finished. Clear this check box if you do not want to see this map immediately: you simply define how the output map should be calculated.
Description:	Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed.

A dependent output map is created.

## Command line

The Polygons to segments operation can be directly executed by typing one of the following expressions on the command line of the Main Window:

```
OUTMAP = SegmentMapPolBoundaries(InputPolygonMap, "*", single)
```

```
OUTMAP = SegmentMapPolBoundaries(InputPolygonMap, "mask",
    single)
```

```
OUTMAP = SegmentMapPolBoundaries(InputPolygonMap, "*", composite)
```

```
OUTMAP = SegmentMapPolBoundaries(InputPolygonMap, "mask",
    composite)
```

where:

OUTMAP	is the name of your output segment map.
SegmentMapPolBoundaries	is the command to start the Polygons to segments operation.
InputPolygonMap	is the name of your input polygon map.
"mask"	are the (multiple) search strings to extract segments of specific polygons; on the command line, the total mask needs to be surrounded by double quotes.
single	is the parameter to create a domain with only one item 'Segments'.
composite	is the parameter to create a domain with composite names.

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated.

## 7.8.5 Polygons to points

### Functionality

The Polygons to points operation creates a point for each polygon in the polygon map. Each point obtains the class name, ID, or value of the corresponding polygon. In this way, polygon label points are created.

- 
- ☞ By creating polygon label points, you can easily display polygon names on top of the polygon map. First display the polygon map, then display the label point map in the same map window; select the Text check box in the Display Options dialog box of the point map.
  - ☞ When you are not satisfied with the position of a label point, display the polygon boundaries and the label points in the same map window, break the dependency link of the point map, and use the Point editor to move a label point to another position.
  - ☞ In the Display Options dialog box of the point map, you can also clear the Symbol check box and select the Text check box to have the labels appear centered at the exact point locations.
- 

#### Input map requirements

No special input map requirements.

#### Domain and coordinate system of output map

The output point map uses the same domain as the input polygon map.

The output point map uses the same coordinate system and coordinate boundaries as the input polygon map.

### Dialog box

The Polygons to points operation creates a point for each polygon in the polygon map. Each point obtains the class name, ID, or value of the corresponding polygon. In this way, polygon label points are created.

#### Dialog box options

- Input polygon map: Select an input polygon map. Open the list box and select the desired input map, or drag a polygon map directly from the Catalog into this box. No special input map requirements.
- Label points: For each polygon, a point is created in the output map. The points are located inside the corresponding polygons.
- Output point map: Type a name for the output point map that will contain the polygon label points.
- Show: Select this check box if you want the output map to be displayed in a map window when the operation has finished. Clear this check box if you do not want to see this map

immediately: you simply define how the output map should be created.

Description: Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed.

A dependent output map is created.

## Command line

The Polygons to points operation can be directly executed by typing the following expression on the command line of the Main window:

```
OUTMAP = PointMapPolLabels(InputPolygonMap)
```

where:

OUTMAP is the name of your output point map.  
PointMapPolLabels is the command to start the Polygons to points operation.  
InputPolygonMap is the name of the input polygon map.

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated.

## Algorithm

The Polygons to points operation creates a point for each polygon in the polygon map. Each point obtains the class name, ID, or value of the corresponding polygon. In this way, polygon label points are created.

### Label points

For each polygon:

- The Y-coordinate of a label point is determined as the mean value of the minimum and maximum Y-coordinate used by the polygon. In other words, the Y-coordinate of the label point will be positioned on the horizontal middle line of the polygon.
- In case the polygon is convex, the X-coordinate of a label point is determined as the mean of the minimum and maximum X-coordinate used by the polygon.
- In case the polygon is concave, it is possible that the horizontal middle line is cut into several pieces by the polygon boundary. Then, the X-coordinate of the label point is positioned at the middle of the longest piece of the horizontal middle line inside the polygon.

This means that:

- for circular polygons, the label point lies more or less at the center of the polygon;
- for vertical half-moon shaped polygons, the label point lies in the middle of the broader part of the polygon.
- for horizontal half-moon shaped polygons, the label point lies in one of the tips of the half-moon.
- for hour-glass shaped polygons, the label point may lie at the narrowest part of the polygon.
- for polygons containing one or more islands, the label point may lie at a narrow part close to an island.

When you are not satisfied with the position of a label point, display the polygon boundaries and the label points in the same map window, break the dependency link of the point map, and use the Point editor to move a label point to another position.

## 7.8.6 Segments to polygons

### Functionality

The Segments to Polygons operation polygonizes a segment map. To polygonize segments, all segments must be closed, i.e. connected to other segments or to themselves (islands) by nodes and no dead ends should exist.

A polygon is a list of a linked segments representing the (complete) boundary of an area. Provinces, soil units, cadastral lots and other mapping units can be stored as polygons. Polygon maps are generally used as a stepping stone to raster maps.

A segment map may contain different classes or IDs representing different 'layers'. From each segment class name or set of class names, you can create different polygon maps. By selecting a mask the user specifies the segment class names/IDs that are to be used to create a polygon map.

Polygons can be named using two different methods:

- Label Points: the user specifies a point map which contains a label for each polygon, i.e. a class name or an Identifier for each polygon.
- Automatic Identifiers: the polygons are sequentially named as Pol 1, Pol 2, etc. With this method an internal domain will be created for the output polygon map.

Prior to the actual polygon creation, segments are checked for possible topological errors. Segment checking is carried out in three steps:

- Self overlap check: to check whether a segment crosses itself (finding false polygons);
- Dead ends check: to check whether begin and end nodes of segments are properly snapped to other nodes;
- Intersections check: to check whether different segments which cross each other have a node at the intersection.

Optionally, when the option AutoCorrect is selected and errors are encountered during the Check Segment phase, errors are automatically removed:

- false polygons are automatically deleted;
- segments which exactly overlay another segment are deleted;
- dead end segments are deleted;
- crossing segments are split and snapped to each other.

Erroneous segments are only deleted in a temporary copy of the segment map, not from the input segment map itself. If AutoCorrection is not selected, and a topological error is encountered, the program stops with an error message.

- 
- ☞ As the user has no control over the AutoCorrect procedure, it is advised to polygonize segments within the Segment editor where you can correct errors yourself.
- 

### **Input map requirements**

No special requirements for the input segment map.

If the Label Point option is selected, the point map with the labels and the segment map should use the same or a compatible coordinate system. When the coordinate systems of the input maps are different (e.g. different projections) but compatible, the point coordinates will be transformed to the coordinate system of the segment map.

### **Domain and coordinate system of output map**

When the Label Point option is selected, the output polygon map will use the same domain as the point map with the labels.

When the Label Point option is not selected, a new ID domain is created which is stored by the output polygon map (internal domain). For more information on internal domains, see How to open and edit internal domains.

## **Dialog box**

The Segments to Polygons operation polygonizes a segment map. All segments in the segment map must be closed, i.e. connected to other segments or to themselves (islands) by nodes; dead ends are not allowed. A mask can be specified to polygonize specific segments.

### **Dialog box options**

- |                    |   |
|--------------------|---|
| Input segment map: | Select an input segment map. Open the list box and select the desired input map, or drag a segment map directly from the Catalog into this box.   |
| Mask:              | Select this check box when you only want to polygonize specific segments. Type (multiple) search strings to find the segments that you want to polygonize. For more information, see Mask Segments : functionality and How to use Masks. Clear this check box to use all segments.  |
| Label points:      | Select this check box to use a point map which contains labels , i.e. a class name or ID, for each polygon. Subsequently, select a point map from the list. Clear this check box to obtain polygons with 'auto-identifiers': the polygons will be automatically and sequentially named as Pol 1, Pol 2, etc. These 'auto-identifiers' will be stored by the output map (internal domain). |
| Auto correction:   | Select this check box if you want the program to automatically correct erroneous segments. This means that  |

segments which are not properly snapped/connected to each other or to themselves by nodes, might not appear as polygon boundaries in the output map. Clear this check box to stop polygonization when an error is encountered in the input segment map.

Output polygon map: Type a name for the output polygon map.  
 Show: Select this check box if you want the output map to be displayed in a map window when the operation has finished. Clear this check box if you do not want to see this map immediately: you simply define how the output map should be created.  
 Description: Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed.

A dependent output map is created.

---

☞ To polygonize segments and have the possibility to interactively correct segment errors, use the **Polygonize** option within the **Segment** editor.

---

## Command line

The Segment to polygons operation can be directly executed by typing one of the following expressions on the command line of the Main window:

```
OUTMAP = PolygonMapFromSegment(InputSegmentMap, "")
```

```
OUTMAP = PolygonMapFromSegment(InputSegmentMap, "", Auto)
```

```
OUTMAP = PolygonMapFromSegment(InputSegmentMap, "",
LabelPointMap)
```

```
OUTMAP = PolygonMapFromSegment(InputSegmentMap, "",
LabelPointMap, Auto)
```

```
OUTMAP = PolygonMapFromSegment(InputSegmentMap, "mask")
```

```
OUTMAP = PolygonMapFromSegment(InputSegmentMap, "mask", Auto)
```

```
OUTMAP = PolygonMapFromSegment(InputSegmentMap, "mask",
LabelPointMap)
```

```
OUTMAP = PolygonMapFromSegment(InputSegmentMap, "mask",
LabelPointMap, Auto)
```

where:

OUTMAP is the name of your output polygon map.

PolygonMapFromSegment is the command to start the Segments to Polygons operation.

InputSegmentMapName is the name of your input segment map.

"mask"	are the (multiple) search strings to find the classes, IDs or values in the input map that you want to use for polygonization; on the command line, the total mask needs to be surrounded by double quotes.
LabelPointMap	optional parameter to specify a point map with labels to name polygons according to label points; if this parameter is not specified, the polygons automatically obtain IDs as Pol 1, Pol 2, etc.
Auto	optional parameter to automatically correct erroneous segments in the input map; if this parameter is not specified, polygonization will stop when an error is encountered in the input segment map.

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated.

## Algorithm

The Segments to Polygons operation polygonizes a segment map. All segments in the segment map must be closed, i.e. connected to other segments or to itself (islands) by nodes; dead ends are not allowed. A mask can be specified to polygonize specific segments.

### Process

- All segments as specified in the mask are copied to the output map.
- Segments are checked for topological errors.
- If AutoCorrect is selected:
  - false polygons are automatically deleted;
  - segments which exactly overlay another segment are deleted;
  - dead end segments are deleted;
  - crossing segments are split and snapped to each other.

Erroneous segments are only deleted in the temporary copy of the segment map, not from the input segment map itself.

- Polygon topology is created. Every polygon boundary gets a forward and backward link, and a left and right polygon. For every polygon, the area and perimeter is calculated.
- If the Label Point option is used: the operation determines which point is surrounded by which polygon. This is checked in the sequence of the point map file. Each polygon is named according to the label point found. If there is no point in a polygon, the polygon will keep the undefined value.

Otherwise each polygon obtains an identifier by the system.

## 7.8.7 Segments to points

### Functionality

The Segments to points operation creates a point map from a segment map. The output point map can contain either:

- a point for each node in the segment map, or
- points at a regular distance along the segments in the segment map; nodes are included regardless of the specified distance, or
- points for all stored coordinate pairs in the segment map.

#### Input map requirements

No special input map requirements.

#### Domain and coordinate system of output map

The output point map uses the same domain as the input segment map.

The output point map uses the same coordinate system and coordinate boundaries as the input segment map.

### Dialog box

The Segments to points operation creates a point map from a segment map. The output point map can contain either:

- a point for each node in the segment map, or
- points at a regular distance along the segments in the segment map, or
- points for all stored coordinate pairs in the segment map.

#### Dialog box options

- Input segment map: Select an input segment map. Open the list box and select the desired input map, or drag a segment map directly from the Catalog into this box. No special input map requirements.
- Only nodes: Select this option button if you want to obtain all the segment nodes as points.
- Distance: Select this option button if you want to obtain coordinates at a regular distance along segments, and type a value for the distance interval (in meters); real value > 0.
- All points: Select this option button if you want to obtain the segment nodes and intermediate coordinates as points.
- Output point map: Type a name for the output point map that will contain the segment label points.
- Show: Select this check box if you want the output map to be displayed in a map window when the operation has finished. Clear this check box if you do not want to see this map immediately: you simply define how the output map should be created.

Description: Optionally, type a description for the output map. The description appears in the title bar when the output map is displayed.

A dependent output map is created.

## Command line

The Segments to points operation can be directly executed by typing one of the following expression on the command line of the Main window:

```
OUTMAP = PointMapSegDist(InputSegmentMap, distance)
```

```
OUTMAP = PointMapSegNodes(InputSegmentMap)
```

```
OUTMAP = PointMapSegCoords(InputSegmentMap)
```

where:

OUTMAP	is the name of your output point map.
PointMapSegDist	is the command to start extracting points from segments at a regular distance interval.
PointMapSegNodes	is the command to start extracting all nodes from a segment map into a point map.
PointMapSegCoords	is the command to start extracting all nodes and intermediate coordinates from a segment map into a point map.
InputSegmentMap	is the name of the input segment map.
distance	is the distance interval (in meters) at which points should be extracted from segments; real value > 0.

When the definition symbol = is used, a dependent output map is created; when the assignment symbol := is used, the dependency link is immediately broken after the output map has been calculated.

## Algorithm

The Segments to points operation simply copies either:

- all nodes of a segment map into a point map,
- finds coordinates along segments at a user-defined distance interval and writes these to a point map, or
- all nodes and intermediate coordinates of a segment map into a point map.

Segments to points by distance:

Segment nodes are always copied to the point map. For each segment,

- the program starts at the begin node,
- then the segment is followed along its intermediate coordinates using straight lines, and
- after each user-defined distance interval along this path, a point is created in the point map.

## 7.9 Table Operations

### 7.9.1 Transpose table

#### Functionality

The Transpose table operation interchanges the rows and columns of a table. Each row of the input table becomes a column in the output table; while column names of the input table become output domain records.

#### Example

In this example, a Rainfall table which contains rainfall figures of January, February and March for all rainfall stations in the Cochabamba area is transposed.

Input table:

	january	february	march
Aro Cagua	85	83	76
Cerro MachaMach	170	165	155
Colca Pithua	99	87	79
Laguna Santa Rosa	175	165	160
Laguna Totura	180	170	150
LagunaWaraWara	175	170	160
PROMIC	93	89	81
taquina	99	95	87
UMSS	100	96	87

Output table:

	Aro_Cagua	Cerro_MachaMach	Colca_Pithua	Lad
january	85.0	170.0	99.0	
february	83.0	165.0	87.0	
march	76.0	155.0	79.0	

#### Input table requirements

The input table should have a class or ID domain or domain None.

The input table should not have more than 1000 records. The columns in the input table are generally value columns.

#### Domain of output table and of output columns

The Transpose table operation creates a class domain out of all column names of the input table. The output table thus uses a class domain.

In the output table, all columns will use the same domain; you can select the domain for the output columns during the operation.

#### Dialog box

The Transpose table operation interchanges the rows and columns of a table. Each row of the input table becomes a column in the output table; while column names of the input table become output domain records.

**Dialog box options**

Input table:	Select an input table. Open the list box and select the desired input table, or drag a table directly from the Catalog into this box.
Output table:	Type the name of the output table.
Domain for columns:	Select the domain that will be used for all output columns in the output table; or create a new domain (click the create button).
Value range:	Accept the default value range, or specify your own range of possible values in the output columns.
Precision:	Accept the default precision for the output values, or specify your own precision.
Show:	Select this check box if you want the output table to be displayed in a table window when the operation has finished. Clear this check box if you do not want to see this table immediately: you simply define how the output table should be created.
Description:	Optionally, type a description for the output table. The description appears in the title bar when the output table is displayed.

A dependent output table is created.

**Command line**

The Transpose table operation can be directly executed by typing one of the following expressions on the command line of the Main Window:

```
OUTTABLE = TableTranspose(InputTableName, ColumnDomain)
```

```
OUTTABLE = TableTranspose(InputTableName, ColumnDomain,
                           ValueRange)
```

where:

OUTTABLE	is the name of your output table.
TableTranspose	is the command to start the Transpose table operation.
InputTableName	is the name of your input table.
ColumnDomain	is the domain that will be used for all output columns.
ValueRange	is the value range that will be used for all output columns.

When the definition symbol = is used, a dependent output table is created; when the assignment symbol := is used, the dependency link is immediately broken after the output table has been calculated.

## Algorithm

The Transpose table operation interchanges the rows and columns of a table. Each row of the input table becomes a column in the output table; while column names of the input table become output domain records.

The Transpose table operation creates a class domain out of all column names of the input table. In the output table, all columns will use the same domain; this is defined by the user. The input table should not have more than 1000 records.

In case the input table uses domain None, the names of the output columns will be Col 1, Col 2, etc.