

APÊNDICE A – Documentação do Fluxo de Trabalho

FLUXO DE HIDROGRAFIA E ALTIMETRIA

O objetivo do processo de correção de altimetria e hidrografia é identificar e corrigir erros comuns em dados geoespaciais por meio de 18 rotinas, sendo 17 de identificação e uma de manipulação para gerar as áreas com delimitadores e centróide após as correções topológicas, garantindo a execução correta.

As etapas iniciais do fluxo visam identificar geometrias inválidas, como vértices não compartilhados em interseções, sobreposições de feições e geometrias com topologia incorreta, entre outros erros que podem surgir devido a problemas na coleta, digitalização ou processamento de dados geoespaciais. A correção desses erros é essencial para evitar falhas na execução das etapas seguintes do processo de identificação de erros, tornando a etapa de identificação de geometrias inválidas uma prioridade na correção.

Os comandos em JSON funcionam como instruções de como o algoritmo provedor deve se comportar.

01- Identificação de geometrias inválidas:

Descrição: Este processo identifica geometrias inválidas nas camadas especificadas.

Arquivo: identifica_geometrias_invalidas_alt_hid_carta_orto.model3

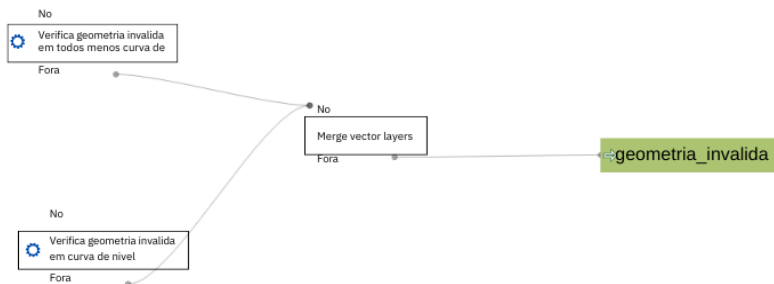
Algoritmos:

dsgtools:batchrunalgorithm,

dsgtools:identifyandfixinvalidgeometries,

native:mergevectorlayers

Composição do Modelo:



O modelo é composto por três etapas:

Verifica geometria inválida: Identifica geometrias inválidas em todas as camadas de entrada, exceto a camada de curva de nível, utilizando o algoritmo *"identifyandfixinvalidgeometries"*.

Verifica geometria inválida: Identifica geometrias inválidas somente na camada de curva de nível utilizando o algoritmo *"identifyandfixinvalidgeometries"*.

Merge Vector Layers: Realiza uma operação de unir as camadas de saída utilizando o algoritmo *mergevectorlayers*.

Parâmetros:

Execução de algoritmos em lote

Properties | Comments

Description: Verifica geometria invalida em todos menos curva de nivel

Nomes das camadas de entrada separados por vírgula

Nome do algoritmo com seu provedor

Nome da chave de entrada

Dicionário de parâmetros em JSON

Nome dos parâmetros de saída da camada [opcional]

Execução de saída em lote

Dependencies

OK Cancel Help

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Ponto	elemnat_ponto_cotado_p, centroide_massa_dagua_p, centroide_elemento_hidrografico_p, elemnat_elemento_fisiografico_p, elemnat_elemento_hidrografico_p, centroide_ilha_p
linha	delimitador_massa_dagua_l, elemnat_trecho_drenagem_l, elemnat_elemento_fisiografico_l, infra_barragem_l, elemnat_elemento_hidrografico_l elemnat_curva_nivel_l
polígono	cobter_massa_dagua_a, infra_barragem_a

Para lidar com a peculiaridade da curva de nível, em que o início e o fim se encontram, foi criada uma verificação separada para o elemento elemnat_curva_nivel_l.

Parâmetros em JSON: definem o comportamento do algoritmo provedor durante sua execução.

Verifica geometrias inválidas em todos, exceto curva de nível:

```
{
  "SELECTED": false,
  "IGNORE_CLOSED": false,
  "TYPE": false,
  "FLAGS": "TEMPORARY_OUTPUT"
}
```

Verifica geometria inválida na curva de nível:

```
{
  "SELECTED": false,
  "IGNORE_CLOSED": true,
  "TYPE": false,
  "FLAGS": "TEMPORARY_OUTPUT"
}
```

O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como **"true"**, a validação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são validadas.

O parâmetro **"IGNORE_CLOSED"** determina se a rotina deve ignorar as geometrias fechadas, como os polígonos. Quando marcado como **"true"**, as geometrias fechadas não são consideradas na validação. Caso contrário, todas as geometrias são avaliadas.

O parâmetro **"TYPE"** é usado para definir o tipo de geometria que será validada. Quando marcado como **"true"**, a validação é realizada apenas nas geometrias do tipo especificado (linha, ponto ou polígono). Caso contrário, todas as geometrias são avaliadas.

O parâmetro **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação. Neste caso, o valor **"TEMPORARY_OUTPUT"** especifica que uma camada temporária será gerada para armazenar informações sobre as geometrias inválidas encontradas durante a validação.

Para finalizar o processo, é realizada uma operação de **"MERGE VECTOR LAYERS"** para combinar todas as camadas temporárias de **FLAGS** em uma única camada.

Nome da camada de flags gerada: geometrias_invalidas.

Resultado do processo e exemplo de erros:



02-Identifica geometrias com mais de uma parte:

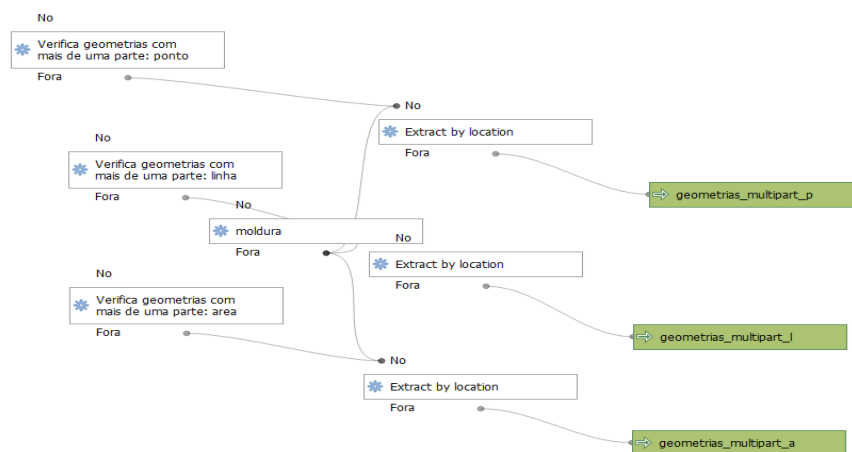
Descrição: O algoritmo em questão identifica geometrias multipartidas (geométricas composta por múltiplas partes) em diferentes tipos de feições (linha, ponto e polígono) dentro de camadas específicas.

Arquivo: identifica_multipart_alt_hid_carta_orto.model3

Algoritmos:

dsgtools:batchrunalgorithm,
dsgtools:identifymultigeometries,
native:extractbylocation
dsgtools:stringcsvtofirstlayerwithelementsalgorithm

Composição do Modelo:



O modelo é composto por três etapas:

Verifica geometrias com mais de uma parte nas camadas de ponto, linha e polígono:

Identifica geometrias com mais de uma parte nas camadas de ponto e linha usando o algoritmo *multiparttosingleparts*.

Moldura: Converte uma string csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

ExtractByLocation: Extrai as geometrias com mais de uma parte que estão dentro da moldura usando o algoritmo *extractbylocation*.

Parâmetros:

Execução de algoritmos em lote

Properties Comments

Description: Verifica geometrias com mais de uma parte: linha


Names das camadas de entrada separados por vírgula
 123 pt_elemento_fislografico_J,infra_barragem_J,elemnat_elemento_hidrografico_J,delimitador_elemento_hidrografico_J

Nome do algoritmo com seu provedor
 123 dsgeotools:identifymultigeometries

Nome da chave de entrada
 123 INPUT

Dicionário de parâmetros em JSON
 123 {
 "SELECTED": false,
 "FLAGS":"TEMPORARY_OUTPUT"
 }

Nome dos parâmetros de saída da camada [opcional]
 123 FLAGS

Execução de saída em lote
 [Digite o nome se este for um resultado final]

Dependencies
 0 dependências selecionadas

OK Cancel Help

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Ponto	elemnat_ponto_cotado_p, centroide_massa_dagua_p, centroide_elemento_hidrografico_p, elemnat_elemento_fisiografico_p, elemnat_elemento_hidrografico_p
Linha	delimitador_massa_dagua_l, elemnat_trecho_drenagem_l, elemnat_curva_nivel_l, elemnat_elemento_fisiografico_l, infra_barragem_l, elemnat_elemento_hidrografico_l, delimitador_elemento_hidrografico_l
Polígono	cobter_massa_dagua_a, infra_barragem_a

Para cada tipo de geometria, o *model* utiliza o algoritmo "dsgtools:identifymultigeometries" para identificar as geometrias multipartidas presentes na camada.

Parâmetros em JSON:

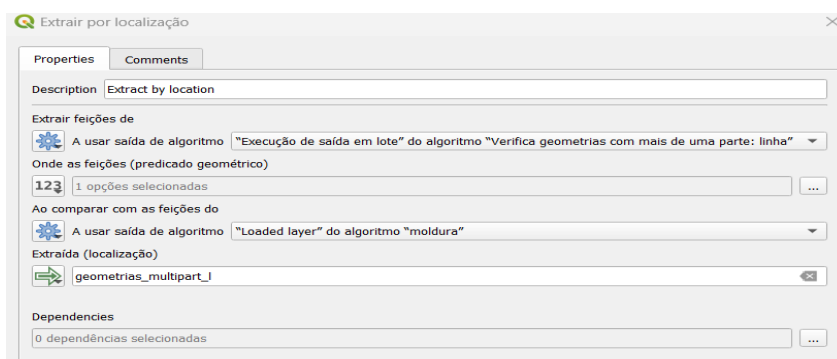
```

{"SELECTED": false,

```

```
"FLAGS": "TEMPORARY_OUTPUT"
}
```

Na sequência, o modelo usa o algoritmo do QGIS *"native:extractbylocation"* para extrair apenas as geometrias que intersectam com a camada "moldura", no modo interseção (parâmetro *"PREDICATE"* = [0]). Esse parâmetro significa que somente as geometrias que têm intersecção com a moldura serão extraídas.

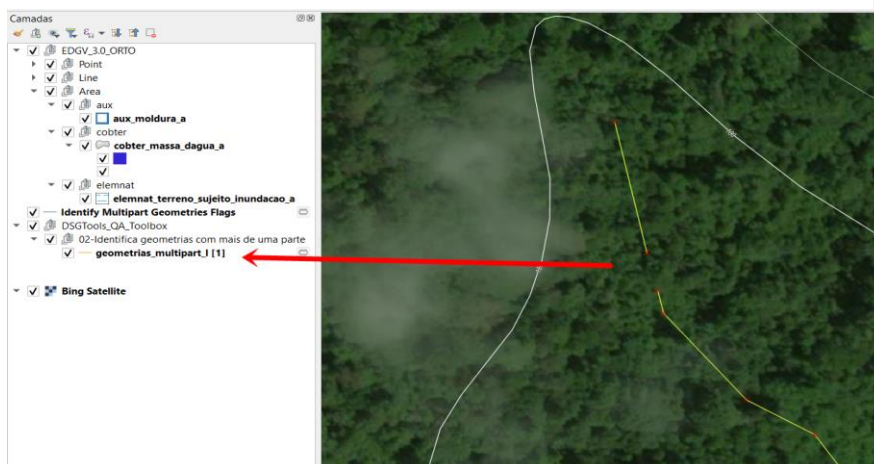


O objetivo é restringir as análises apenas às geometrias que estão dentro de uma área de interesse, no caso a moldura.

Nome da camada de flags:

geometrias_multipart_p, geometrias_multipart_l, geometrias_multipart_a

Resultado do processo e exemplo de erros:



03- Identificar feições duplicadas

Descrição: Descrição: Essa etapa tem como objetivo identificar feições duplicadas em diferentes camadas do projeto, que podem ter sido criadas acidentalmente ou durante o processo de validação. Para isso, o modelo utiliza uma lista negra de atributos (*ATTRIBUTE_BLACKLIST*) que não serão considerados na comparação das feições, a fim de evitar falsos positivos. As camadas de ponto, linha e polígono são verificadas separadamente e as feições duplicadas encontradas são sinalizadas com flags nas camadas correspondentes.

Arquivo: identifica_feicoes_duplicadas_alt_hid_carta_orto.model3

Algoritmos:

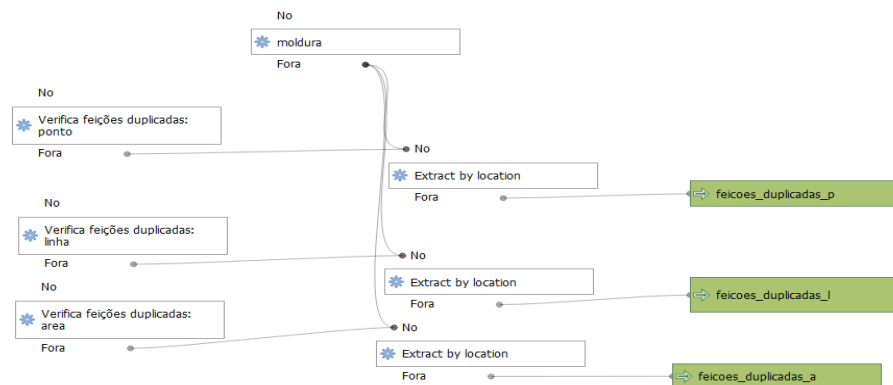
dsgtools:identifyduplicatedfeatures,

dsgtools:batchrunalgorithm,

dsgtools:identifyunsharedvertexonintersectionsalgorithm

dsgtools:stringcsvtofirstlayerwithelementsalgorithm

Composição do Modelo:



O modelo é composto por cinco etapas:

Moldura: Converte uma string csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

Verifica feições duplicadas: Identifica feições duplicadas nas camadas de ponto e linha e polígono usando o algoritmo *identifyduplicatedfeatures*.

ExtractByLocation: Extrai as feições duplicadas que estão dentro da moldura usando o algoritmo *extractbylocation*.

Parâmetros:

Execução de algoritmos em lote

Properties Comments

Description: Verifica feições duplicadas: linha

Nomes das camadas de entrada separados por vírgula

123: l_curva_nivel_l,elemnat_elemento_fisiografico_l,infra_barragem_l,elemnat_elemento_hidrografico_l,delimitador_elemento_hidrografico_l

Nome do algoritmo com seu provedor

123: dsgrtools:identifyduplicatedfeatures

Nome da chave de entrada

123: INPUT

Dicionário de parâmetros em JSON

123: {
 "SELECTED": false,
 "ATTRIBUTE_BLACK_LIST":
 ["id","texto_edicao","label_x","label_y","justificativa_txt","tamanho_txt","visivel","carta_simbolizacao","simbolizar_carta_mini","simb_rot",
 "rotular_carta_mini","espacamento","tamanho_bt","estilo_fonte","cor","cor_buffer","tamanho_buffer","observacao","length_otf"],
 "IGNORE_VIRTUAL_FIELDS": true,
 "IGNORE_PK_FIELDS": true,
 "FLAGS": "TEMPORARY_OUTPUT"
}

Nome dos parâmetros de saída da camada [opcional]

123: FLAGS

Execução de saída em lote

123: [Digite o nome se este for um resultado final]

Dependencies

0 dependências selecionadas

OK Cancel Help

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Ponto	elemnat_ponto_cotado_p, centroide_massa_dagua_p, centroide_elemento_hidrografico_p,elemnat_elemento_fisiografico_p, elemnat_elemento_hidrografico_p
Linha	delimitador_massa_dagua_l, elemnat_trecho_drenagem_l, elemnat_curva_nivel_l, elemnat_elemento_fisiografico_l, infra_barragem_l, elemnat_elemento_hidrografico_l, delimitador_elemento_hidrografico_l
Polígono	cobter_massa_dagua a, infra_barragem_a

Parâmetros em JSON:

```
{  
  "SELECTED": false,  
  "ATTRIBUTE_BLACK_LIST":  
    ["id", "texto_edicao", "label_x", "label_y", "justificativa_txt", "tamanho_txt", "visivel", "operador_criacao", "data_criacao", "operador_atualizacao", "data_atualizacao", "carta_simbolizacao", "simbolizar_carta_mini", "simb_rot", "rotular_carta_mini", "espacamento", "tamanho_txt", "estilo_fonte", "cor", "cor_buffer", "tamanho_buffer", "observacao", "length_of"]
```

```

"IGNORE_VIRTUAL_FIELDS": true,
"IGNORE_PK_FIELDS": true,
"FLAGS": "TEMPORARY_OUTPUT"
}

```

nome da camada de flags: feicoes_duplicadas_p,feicoes_duplicadas_l,feicoes_duplicadas_a

Resultado do processo e exemplo de erros:



04-Identifica vértice não compartilhado nas intersecções:

Descrição: Este algoritmo identifica vértices não compartilhados nas intersecções entre camadas de linha e polígono.

Arquivo:

identifica_vertice_nao_compartilhado_nas_interseccoes_alt_hid_carta_orto.model3

Algoritmos:

dsgtools:stringcsvtofirstlayerwithelementsalgorithm,

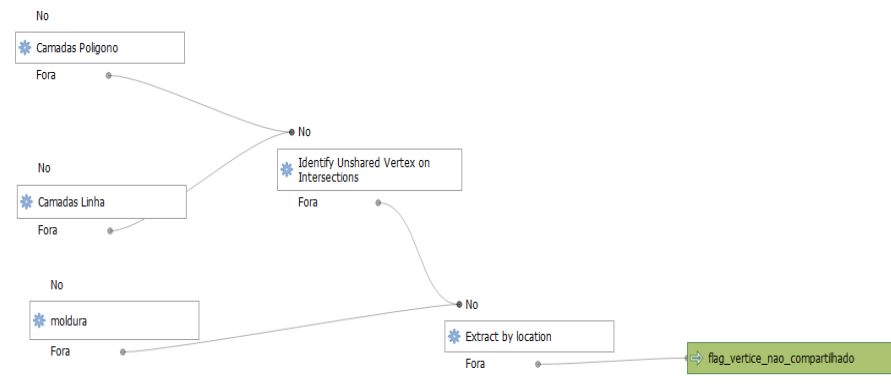
dsgtools:stringcsvtolayerlistalgorithm,

dsgtools:stringcsvtolayerlistalgorithm

dsgtools:identifyunsharedvertexonintersectionsalgorithm.

native:extractbylocation

Composição do Modelo:



O modelo é composto por cinco etapas:

Camadas Polígono: Converte uma string csv em uma lista de camadas de polígono usando o algoritmo *stringcsvtolayerlistalgorithm*.

Camadas Linha: Converte uma string csv em uma lista de camadas de linha usando o algoritmo *stringcsvtolayerlistalgorithm*.

Moldura: Converte uma string csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

IdentifyUnsharedVertexOnIntersections: Identifica vértices não compartilhados nas intersecções entre as camadas de linha e polígono usando o algoritmo *identifyunsharedvertexonintersectionsalgorithm*.

ExtractByLocation: Extrai os vértices não compartilhados que estão dentro da moldura usando o algoritmo *extractbylocation*.

Parâmetros:

Camadas de entrada: Uma lista de camadas de linha e polígono.

Processar apenas feições selecionadas: Um booleano que indica se apenas as entidades selecionadas na camada de entrada serão processadas.

Flags: Uma camada de saída temporária que armazena as intersecções identificadas com vértices não compartilhados.

Extrair feições de: Utiliza os dados de saída do procedimento anterior.

Predicado geométrico: Uma lista que especifica o tipo de relação espacial entre as geometrias que devem ser consideradas na extração. Neste caso, o valor [0] significa que apenas as geometrias que se intersectam serão extraídas.

Ao comparar com as feições do: A camada de polígono que será usada para extrair as intersecções.

Flags: A camada de saída final que armazenará os vértices não compartilhados identificados.

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	delimitador_massa_dagua_l, elemnat_trecho_drenagem_l, infra_barragem_l, elemnat_elemento_hidrografico_l
Polígono	cobter_massa_dagua_a,infra_barragem_a

nome da camada de flags: Flag_vertice_nao_compartilhado

Resultado do processo e exemplo de erros:



05-Identificar vértice não compartilhado nos segmentos compartilhados:

Descrição: Este algoritmo identifica vértices não compartilhados em segmentos compartilhados entre camadas de linha e polígono.

Arquivo:

identificar_vertice_nao_compartilhado_nos_segmentos_compartilhados.model3

Algoritmos:

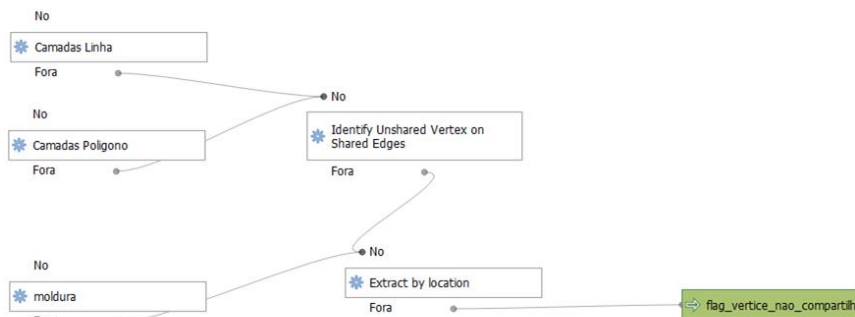
dsgtools:stringcsvtolayerlistalgorithm,

dsgtools:stringcsvtofirstlayerwithelementsalgorithm,

dsgtools:identifyunsharedvertexonsharededgesalgorithm,

native:extractbylocation

Composição do Modelo:



O modelo é composto por cinco etapas:

Camadas Linha: Converte uma string csv em uma lista de camadas de linha usando o algoritmo *stringcsvtolayerlistalgorithm*.

Camadas Polígono: Converte uma string csv em uma lista de camadas de polígono usando o algoritmo *stringcsvtolayerlistalgorithm*.

Moldura: Converte uma string csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

IdentifyUnsharedVertexOnSharedEdges: Identifica vértices não compartilhados nos segmentos compartilhados entre as camadas de linha e polígono usando o algoritmo *identifyunsharedvertexonsharededgesalgorithm*.

ExtractByLocation: Extrai os vértices não compartilhados que estão dentro da moldura usando o algoritmo *extractbylocation*.

Parâmetros:

Camadas de entrada:

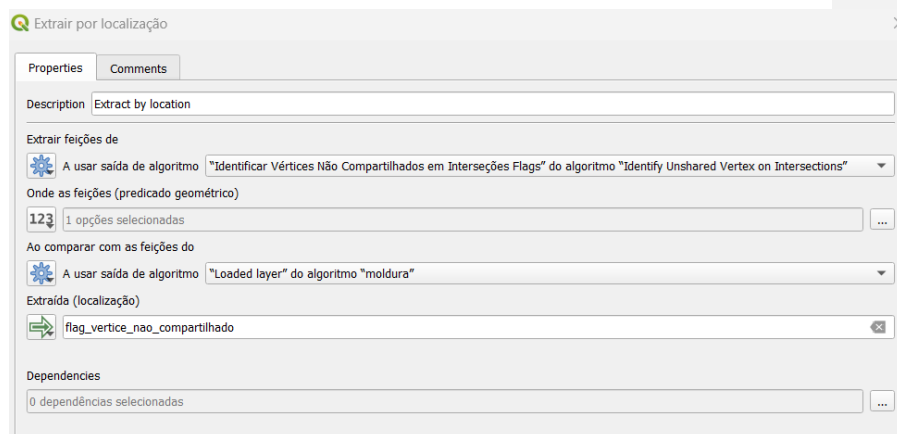
Tipo de geometria	Camadas de entrada
Linha	delimitador_massa_dagua_l, elemnat_trecho_drenagem_l, infra_barragem_l, elemnat_elemento_hidrografico_l
Polígono	cobter_massa_dagua_a,infra_barragem_a

Processar apenas feições selecionadas: Um booleano que indica se apenas as entidades selecionadas na camada de entrada serão processadas.

O Parâmetro **Raio de busca**: O raio de busca do algoritmo determina a distância máxima que será percorrida para encontrar vértices não compartilhados em relação aos segmentos de linha e polígonos que se cruzam nas intersecções. No modelo, o valor padrão do raio de busca é definido como 0,00001 unidades do sistema de coordenadas utilizado no processamento. No entanto, é importante lembrar que esse valor pode ser ajustado de acordo com as necessidades específicas, considerando a escala e precisão dos dados utilizados no processamento. Outro ponto importante que precisa ser ressaltado é que deve ser levado em consideração que se houver vértices muito próximos em uma linha, (dentro do raio de busca), será gerado uma flag apontando o erro como vértice não compartilhado, nesse caso a geometria precisa ser simplificada, reduzindo os vértices muito próximos.

Comentado [GPG1]: Isso precisa ser verificado, é importante.

Flags: Uma camada de saída temporária que armazena as intersecções identificadas com vértices não compartilhados.



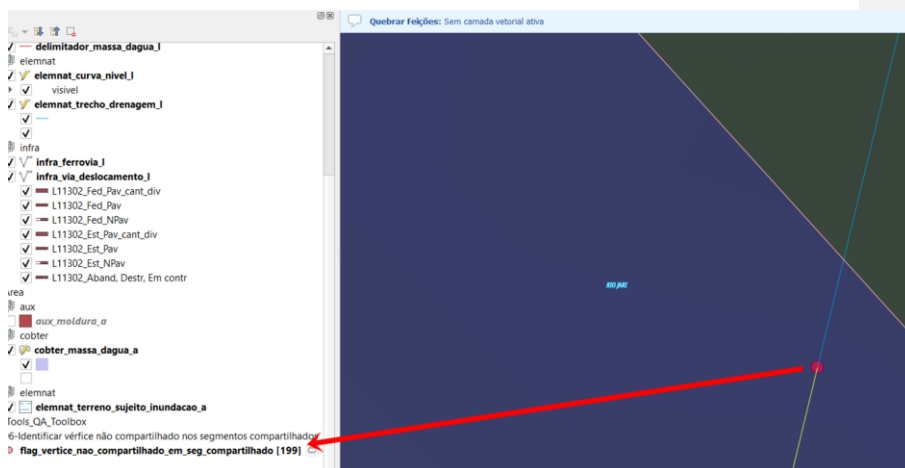
Extrair feições de: Utiliza os dados de saída do procedimento anterior.
Predicado geométrico: Uma lista que especifica o tipo de relação espacial entre as geometrias que devem ser consideradas na extração. Neste caso, o valor [0] significa que apenas as geometrias que se intersectam serão extraídas.

Ao comparar com as feições do: A camada de polígono que será usada para extrair as intersecções.

Flags: A camada de saída final que armazenará os vértices não compartilhados identificados.

nome da camada de flags: *Flag_vertice_nao_compartilhado_em_seg_compartilhado*

Resultado do processo e exemplo de erros:



06-Identificar Vértice Próximo de Aresta:

Descrição: Este algoritmo identifica vértices muito próximos de arestas em uma lista de camadas.

Arquivo: identificar_vertice_proximo_de_aresta.model3

Algoritmo:

dsgtools:batchrunalgorithm

dsgtools:stringcsvtofirstlayerwithelementsalgorithm

native:extractbylocation

Composição do Modelo:



O modelo é composto por três etapas:

BatchRunAlgorithm: Executa o algoritmo *identifyvertexnearedges* em uma lista de camadas de linha.

Moldura: Converte uma string csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

ExtractByLocation: Extrai os vértices que estão muito próximos de arestas, utilizando um raio pequeno de busca, e que estão dentro da moldura usando o algoritmo *extractbylocation*.

Parâmetros:

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	delimitador_massa_dagua_l, elemnat_trecho_drenagem_l, elemnat_curva_nivel_l, infra_barragem_l, elemnat_elemento_hidrografico_l
Polígono	cobter_massa_dagua_a,infra_barragem_a

Parâmetros em JSON: {

"SELECTED": false,

"SEARCH_RADIUS": 1e-05,

"FLAGS": "TEMPORARY_OUTPUT"

}

O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como "true", a validação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são validadas.

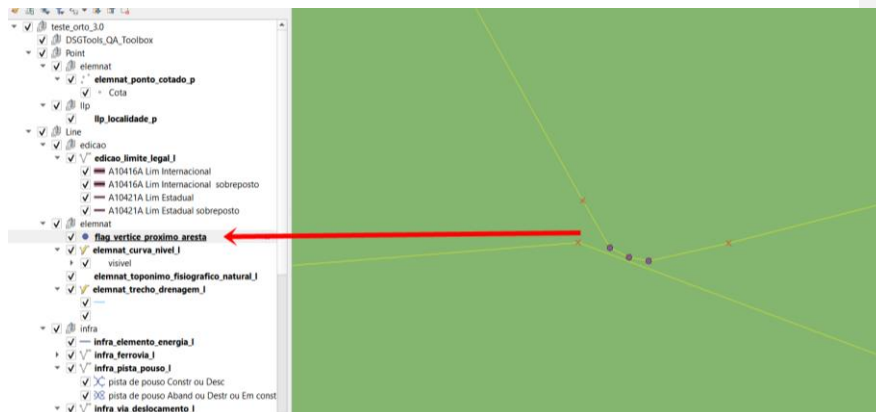
O parâmetro **"SEARCH_RADIUS"**: O raio de busca de 0,00001 do algoritmo determina a distância máxima que será percorrida, nesse caso com o objetivo de buscar distâncias muito pequenas entre os vértices que poderiam vir a colapsar a geometria.

Comentado [GPG2]: Seria interessante descobrir o que no Sistema de referência altera nesse raio de busca

O parâmetro **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação. Neste caso, o valor **"TEMPORARY_OUTPUT"** especifica que uma camada temporária será gerada para armazenar informações sobre os vértices próximo a aresta encontradas durante a validação.

nome da camada de flags: Flag_vertice_proximo_aresta

Resultado do processo e exemplo de erros:



07-Identifica geometrias com densidade incorreta de vértices:

08-Identificar ângulos pequenos:

Descrição: Este algoritmo identifica ângulos pequenos em uma lista de camadas de linha e polígono.

Arquivo: identifica_angulos_pequenos_alt_hid_carta_orto.model3

Algoritmo:

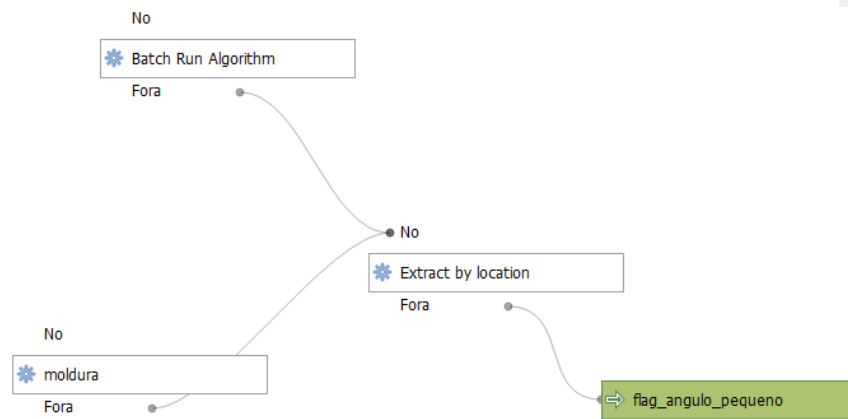
dsgtools:batchrunalgorithm

dsgtools:stringcsvtofirstlayerwithelementsalgorithm

native:extractbylocation

dsgtools:identifyoutofboundsangles

Composição do Modelo:



O modelo é composto por três etapas:

BatchRunAlgorithm: Executa o algoritmo *identifyoutofboundsangles* com os parâmetros selecionados abaixo.

Moldura: Converte uma *string* csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

ExtractByLocation: Extrai os ângulos pequenos que estão dentro da moldura usando o algoritmo *extractbylocation*.

Parâmetros:

Execução de algoritmos em lote

Properties Comments

Description Batch Run Algorithm

Nomes das camadas de entrada separados por vírgula

123 elemnat_curva_nivel_l,cobter_massa_dagua_a,infra_barragem_a,infra_barragem_l,elemnat_elemento_hidrografico_a

Nome do algoritmo com seu provedor

123 dsuptools:identifyoutofboundsangles

Nome da chave de entrada

123 INPUT

Dicionário de parâmetros em JSON

123 {
 "SELECTED": false,
 "TOLERANCE": 10,
 "FLAGS": "TEMPORARY_OUTPUT"
}

Nome dos parâmetros de saída da camada [opcional]

123 FLAGS

Execução de saída em lote

➡ [Digite o nome se este for um resultado final]

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	delimitador_massa_dagua_l, elemnat_trecho_drenagem_l, elemnat_curva_nivel_l, infra_barragem_l, elemnat_elemento_hidrografico_l
Polígono	cobter_massa_dagua_a,infra_barragem_a

Parâmetros em JSON: {

"SELECTED": false,

"TOLERANCE": 10,

```
"FLAGS": "TEMPORARY_OUTPUT"
```

}

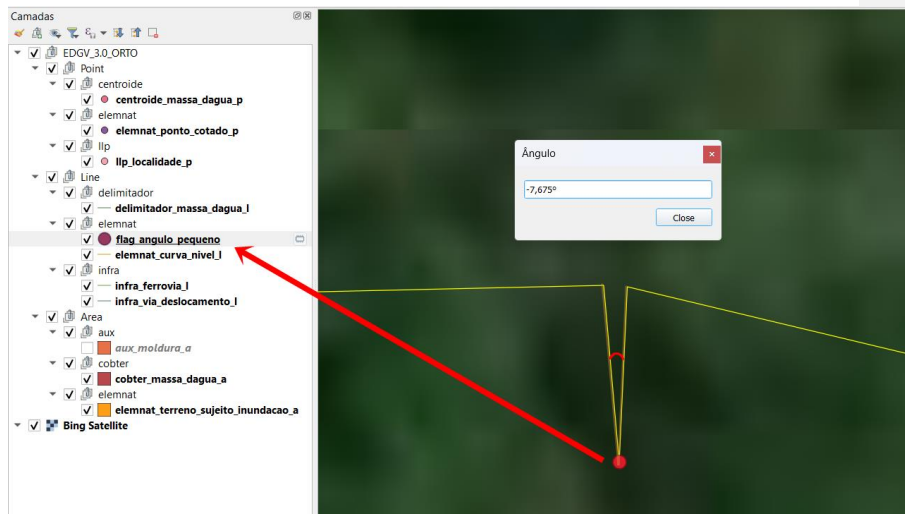
O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como "true", a validação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são validadas.

O parâmetro **"TOLERANCE"**: Um ângulo de tolerância de 10 graus indica que qualquer ângulo abaixo desse valor será considerado pequeno. Essa medida é usada para identificar irregularidades na geometria, tais como curvas de nível muito acentuadas ou dobras em trechos de drenagem. Entretanto, é importante mencionar que para outras aplicações, uma tolerância maior ou menor pode ser mais apropriada.

O parâmetro **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação. Neste caso, o valor **"TEMPORARY_OUTPUT"** especifica que uma camada temporária será gerada para armazenar informações sobre os ângulos pequenos encontradas durante a validação.

nome da camada de flags: Flag_angulo_pequeno

Resultado do processo e exemplo de erros:



09-Identifica ângulos pequenos entre camadas:

Descrição: Este algoritmo funciona de maneira similar à rotina anterior, identificando os ângulos pequenos em uma lista de camadas de linha. A diferença é que as camadas são unificadas, a fim de se comportarem como uma única camada.

Arquivo: identificar_angulos_pequenos_entre_camadas.model3

Algoritmo:

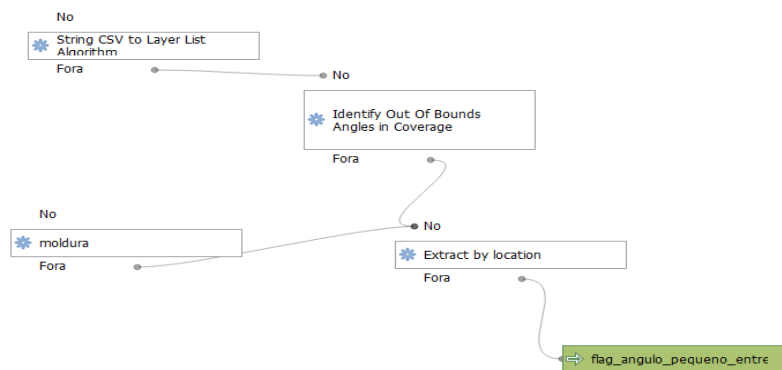
dsgtools:stringcsvtolayerlistalgorithm

native:extractbylocation

dsgtools:identifyoutofboundsanglesincoverage

dsgtools:stringcsvtofirstlayerwithelementsalgorithm

Composição do Modelo:



O modelo é composto por quatro etapas:

StringCsvToLayerListAlgorithm: Converte uma *string* CSV em uma lista de camadas.

IdentifyOutOfBoundsAnglesInCoverage: Identifica ângulos pequenos em cada camada de entrada, com base no parâmetro *TOLERANCE*.

Moldura: Converte uma string csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

ExtractByLocation: Extrai os ângulos pequenos que estão dentro da área delimitada pela moldura usando o algoritmo *extractbylocation*.

Parâmetros:

Identificar Ângulos Fora de Limites na Cobertura Terrestre

Properties Comments

Description: Identify Out Of Bounds Angles in Coverage

Camada de entrada

A usar saída de algoritmo: "Lista múltipla de camadas" do algoritmo "String CSV to Layer List Algorithm"

Processar apenas feições selecionadas

123 Não

Minimum angle (in degrees)

123 10

Identificar Ângulos Fora de Limites na Cobertura Terrestre Flags

[Digite o nome se este for um resultado final]

Dependencies

0 dependências selecionadas

OK Cancel Help

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	delimitador_massa_dagua_l, elemnat_trecho_drenagem_l, elemnat_curva_nivel_l, infra_barragem_l, elemnat_elemento_hidrografico_l

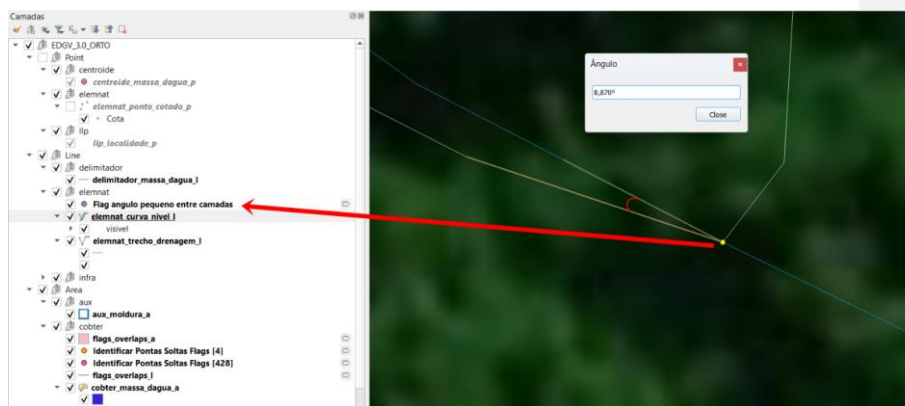
O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como "true", a validação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são validadas.

O parâmetro **"TOLERANCE"**: Um ângulo de tolerância de 10 graus indica que qualquer ângulo abaixo desse valor será considerado pequeno. Essa medida é usada para identificar irregularidades na geometria, tais como curvas de nível muito acentuadas ou dobras em trechos de drenagem. Entretanto, é importante mencionar que para outras aplicações, uma tolerância maior ou menor pode ser mais apropriada.

O parâmetro "**FLAGS**" é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação. Neste caso, o valor "**TEMPORARY_OUTPUT**" especifica que uma camada temporária será gerada para armazenar informações sobre os ângulos pequenos encontradas durante a validação.

nome da camada de flags: Flag_angulo_pequeno

Resultado do processo e exemplo de erros:



10-Identifica ângulos em Z:

Descrição: O ângulo Z é a diferença entre ângulos formados por três pontos consecutivos.

O modelo tem por objetivo encontrar esses ângulos em linhas e polígonos.

Arquivo: identifica_z_alt_hid_carta_orto.model3

Algoritmo:

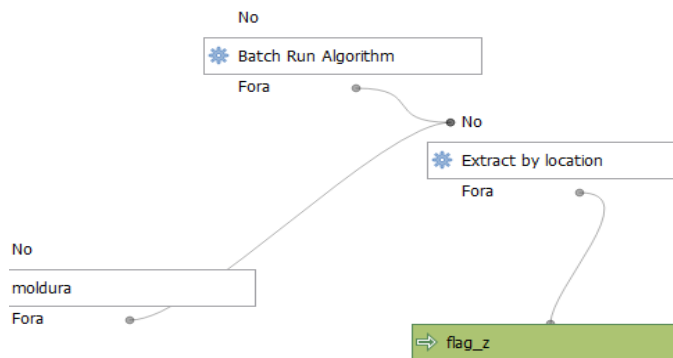
dsgtools:stringsvtolayerlistalgorithm

dsgtools:batchrunalgorithm

native:extractbylocation

dsgtools:identifyanglesbetweenfeatures

Composição do Modelo:



O modelo é composto por três etapas:

Batchrunalgorithm executa o algoritmo *identifyoutofboundsangles* com os parâmetros selecionados abaixo.

Moldura: Converte uma string csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

ExtractByLocation: Extrai os ângulos pequenos que estão dentro da área delimitada pela moldura usando o algoritmo *extractbylocation*.

Parâmetros:

Properties	Comments
Description	Batch Run Algorithm
Nomes das camadas de entrada separados por vírgula	prmat_curva_nivel_l,cobter_massa_dagua_a,infra_barragem_a,infra_barragem_l,elemnat_elemento_hidrografico_l
Nome do algoritmo com seu provedor	dsgtools:identifyanglesbetweenfeatures
Nome da chave de entrada	INPUT
Dicionário de parâmetros em JSON	{ "ANGLE": 300, "OUTPUT": "TEMPORARY_OUTPUT" }
Nome dos parâmetros de saída da camada [opcional]	OUTPUT
Execução de saída em lote	[Digite o nome se este for um resultado final]

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	delimitador_massa_dagua_l, elemnat_trecho_drenagem_l, elemnat_curva_nivel_l, infra_barragem_l, elemnat_elemento_hidrografico_l
Polígono	cobter_massa_dagua_a,infra_barragem_

Parâmetros em JSON: {

"ANGLE": 300,

"OUTPUT": "TEMPORARY_OUTPUT"

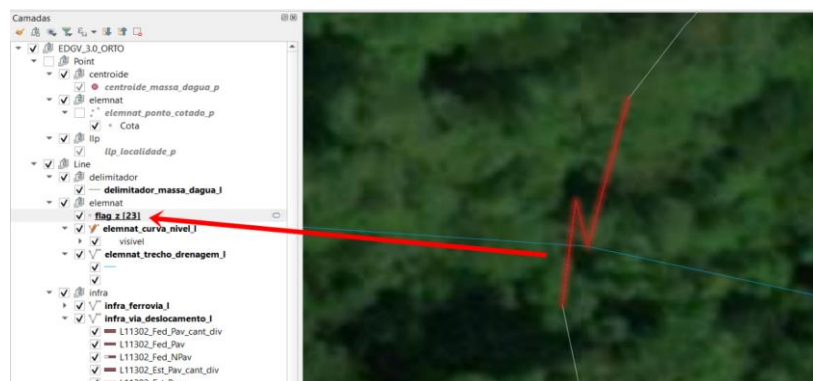
}

O Parâmetro **"ANGLE"** representa a medida de ângulo em graus, onde qualquer ângulo menor do que 300 graus será considerado pequeno. Essa medida é usada para identificar irregularidades na geometria, entretanto, é importante mencionar que para outras aplicações, uma tolerância maior ou menor pode ser mais apropriada.

O Parâmetro **"OUTPUT"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação. Neste caso, o valor **"TEMPORARY_OUTPUT"** especifica que uma camada temporária será gerada para armazenar informações sobre os ângulos inválidos encontradas durante a validação.

nome da camada de flags: Flag_z

Resultado do processo e exemplo de erros:



11- Identificar *overlaps* dentro da mesma camada:

Descrição: Este algoritmo identifica sobreposições dentro da mesma camada em uma lista de camadas de linha e polígono.

Arquivo: identifica_overlaps_dentro_da_mesma_camada.model3

Algoritmo:

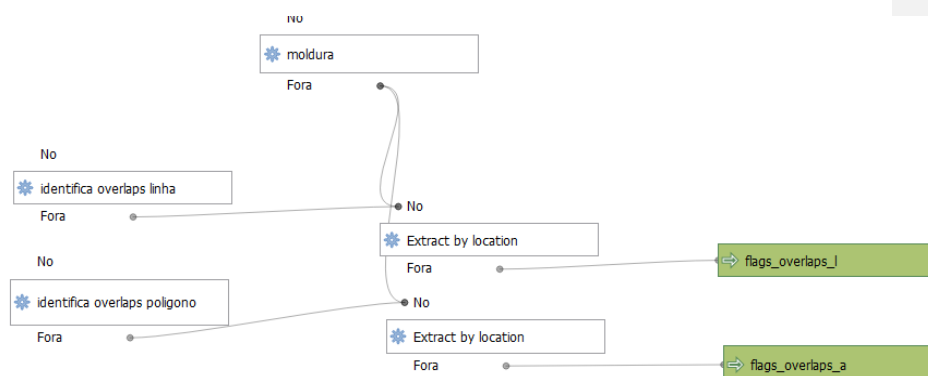
dsgtools:stringcsvtolayerlistalgorithm

dsgtools:batchrunalgorithm

native:extractbylocation

dsgtools:identifyanglesbetweenfeatures

Composição do Modelo:



O modelo é composto por três etapas:

Batchrunalgorithm Executa o algoritmo *identifyoverlaps* com os parâmetros selecionados abaixo para as camadas de linha e separadamente executa para polígonos.

Moldura: Converte uma *string* csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

ExtractByLocation: Extrai os ângulos pequenos que estão dentro da área delimitada pela moldura usando o algoritmo *extractbylocation*.

Parâmetros:

Properties

Comments

Description

identifica overlaps linha

Nomes das camadas de entrada separados por vírgula

123

sa_dagua_l,elemnat_trecho_drenagem_l,elemnat_curva_nivel_l,infra_barragem_l,elemnat_elemento_hidrografico_l

Nome do algoritmo com seu provedor

123

dsgtools:identifyoverlaps

Nome da chave de entrada

123

INPUT

Dicionário de parâmetros em JSON

123

{
"SELECTED": false,
"FLAGS": "TEMPORARY_OUTPUT"
}

Nome dos parâmetros de saída da camada [opcional]

123

FLAGS

Execução de saída em lote

[Digite o nome se este for um resultado final]

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	delimitador_massa_dagua_l, elemnat_trecho_drenagem_l, elemnat_curva_nivel_l, infra_barragem_l, elemnat_elemento_hidrografico_l
Polígono	cobter_massa_dagua_a,infra_barragem_

Parâmetros em JSON: {

"SELECTED": false,

"FLAGS": "TEMPORARY_OUTPUT"

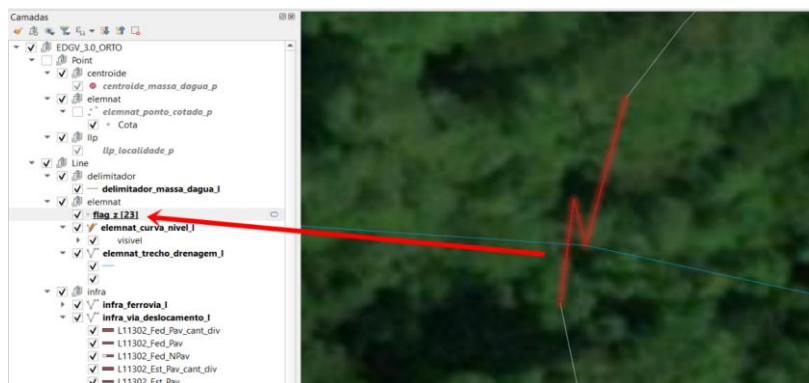
}

O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como **"true"**, a validação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são validadas.

"**FLAGS**" é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação. Neste caso, o valor "**TEMPORARY_OUTPUT**" especifica que uma camada temporária será gerada para armazenar informações sobre os overlaps encontradas durante a validação.

nome da camada de flags: flags_overlaps_1, flags_overlaps_a

Resultado do processo e exemplo de erros:



12- Identifica *undershoot* com moldura e conexão de linhas:

Descrição: Este algoritmo identifica pontas soltas dentro da mesma camada em uma lista de camadas de linha e polígono.

Arquivo: identifica_undershoot_alt_hid_carta_orto.model3

Algoritmo:

dsgtools:batchrunalgorithm

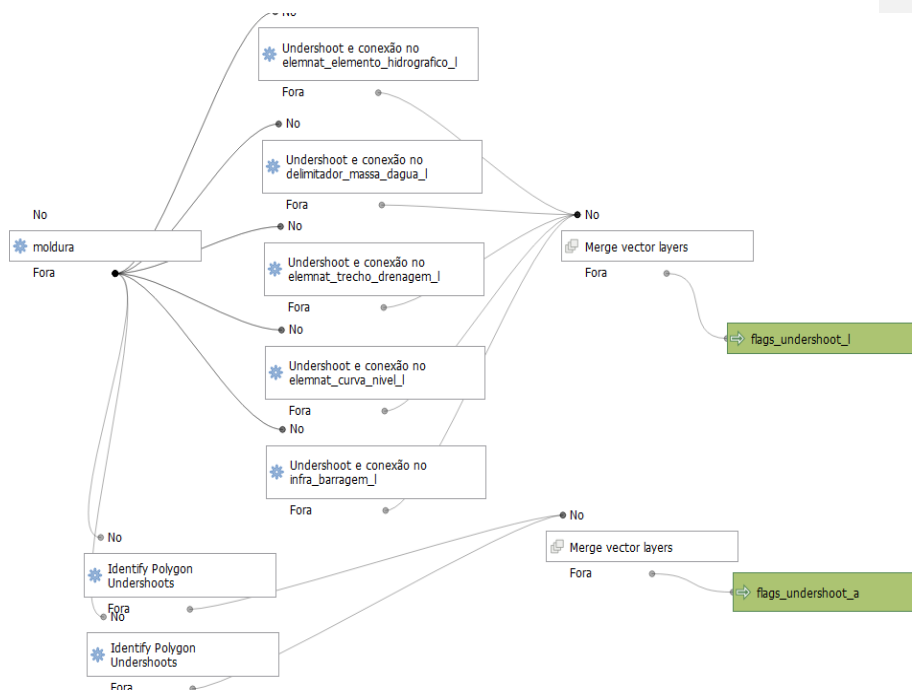
dsgtools:stringcsvtofirstlayerwithelementsalgorithm

dsgtools:identifypolygonundershootsalgorithm

native:mergevectorlayers

dsgtools:identifydanglesalgorithm

Composição do Modelo:



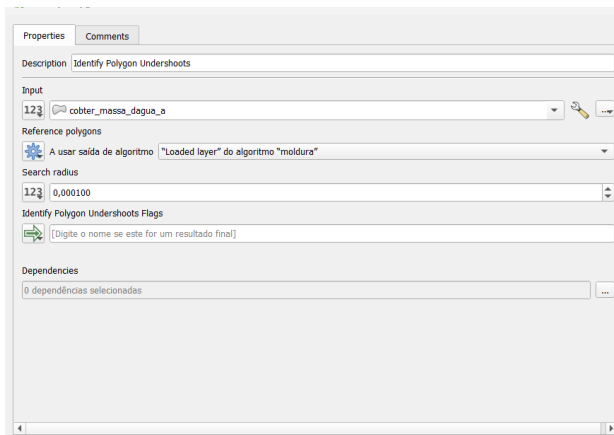
O modelo é composto por três etapas:

Moldura: Converte uma *string* csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

Batchrunalgorithm Executa os algoritmos *identifydangles* para as camadas de linha e *identifypolygonundershoots* para as camadas de polígonos com os parâmetros descritos abaixo.

Merge Vector Layers: Realiza uma operação de unir as camadas de saída utilizando o algoritmo *mergevectorlayers*

Parâmetros:



Camadas de entrada:

Tipo de geometria	Camadas de entrada
Polígono	cobter_massa_dagua_a, infra_barragem_

O parâmetro **"INPUT"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário.

O parâmetro **"REFERENCE POLYGONS"** utiliza a saída do algoritmo anterior que basicamente converte uma lista csv em camadas, no caso em específico utiliza a moldura para delimitar a área de interesse.

O Parâmetro: **"SEARCH RADIUS"** do algoritmo determina a distância máxima que será percorrida para encontrar pontas soltas em relação aos segmentos de linha e polígonos que se cruzam nas intersecções. No modelo, o valor padrão do raio de busca é definido como 0,00001 unidades do sistema de coordenadas utilizado no processamento. No entanto, é importante lembrar que esse valor pode ser ajustado de acordo com as necessidades específicas, considerando a escala e precisão dos dados utilizados no processamento.

O parâmetro: **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação.

nome da camada de flags: *flags_undershoot_a*

Parâmetros:

The screenshot shows the 'Properties' tab of a QGIS processing tool. The description is 'Undershoot e conexão no elemnat_curva_nivel_1'. The input layer is 'elemnat_curva_nivel_1'. The 'Processar apenas feições selecionadas' (Process only selected features) option is set to 'Não' (No). The 'Input is a boundary layer' option is also set to 'Não'. The search radius is '0,000010'. The 'Camada de filtro do tipo linha' (Line filter layer) is set to '0 selecionar entradas' (0 select inputs). The 'Camadas de filtro do tipo polígono' (Polygon filter layers) are set to 'A user saída de algoritmo "Loaded layer" do algoritmo "moldura"' (A user output of algorithm "Loaded layer" of the algorithm "moldura"). The 'Ignorar pontas soltas em linhas não segmentadas' (Ignore dangling lines in non-segmented lines) option is set to 'Sim' (Yes). The 'Geographic Boundary' option is also set to 'A user saída de algoritmo "Loaded layer" do algoritmo "moldura"'. The 'Identificar Pontas Soltas Flags' (Identify Dangling Line Flags) option is set to 'Sim'. The 'Dependencies' section is empty.

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	delimitador_massa_dagua_l, elemnat_trecho_drenagem_l, elemnat_curva_nivel_l, infra_barragem_l, elemnat_elemento_hidrografico_l

O parâmetro **"INPUT"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário.

O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como **"True"**, a validação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são validadas.

O parâmetro **"INPUT IS A BOUNDARY LAYER"** Se a opção **"False"** estiver definida no algoritmo, isso significa que a moldura não precisa estar conectada a elementos da camada de entrada ou dos filtros. É importante ressaltar que o campo elemnat_curva_nivel a opção precisa ser **"True"**, por se tratar de um contorno e ter um comportamento particular.

O Parâmetro: "**SEARCH RADIUS**" do algoritmo determina a distância máxima que será percorrida para encontrar pontas soltas em relação aos segmentos de linha e polígonos que se cruzam nas intersecções. No modelo, o valor padrão do raio de busca é definido como 0,00001 unidades do sistema de coordenadas utilizado no processamento. No entanto, é importante lembrar que esse valor pode ser ajustado de acordo com as necessidades específicas, considerando a escala e precisão dos dados utilizados no processamento

O parâmetro "**CAMADA DE FILTRO**" tem como proposta identificar pontas soltas próximas às camadas especificadas. Essas pontas soltas serão adicionadas à regra de verificação de proximidade durante a busca por feições próximas às analisadas, a fim de minimizar a ocorrência de falsos positivos.

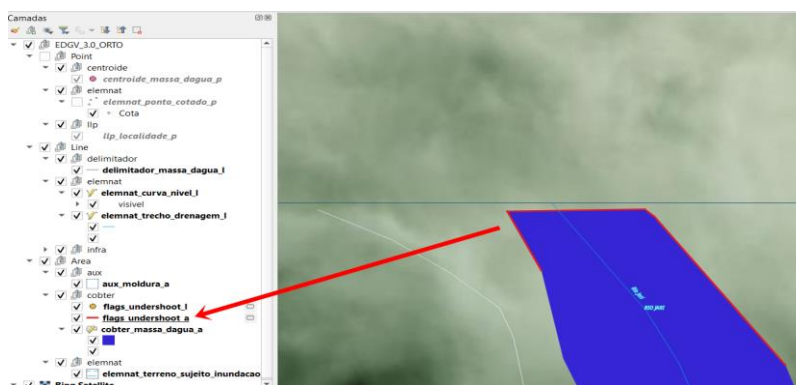
O parâmetro "**IGNORAR PONTAS SOLTAS EM LINHAS NÃO SEGMENTADAS**" quando marcado como "*True*" permite que o algoritmo ignore pontas soltas em linhas não segmentadas. Ou seja, ele não considera como *undershoots* aquelas pontas soltas que não estão conectadas a outros segmentos de linha. Ao ignorar essas pontas, o algoritmo pode concentrar-se apenas nas linhas que realmente importam e fornecer resultados mais precisos.

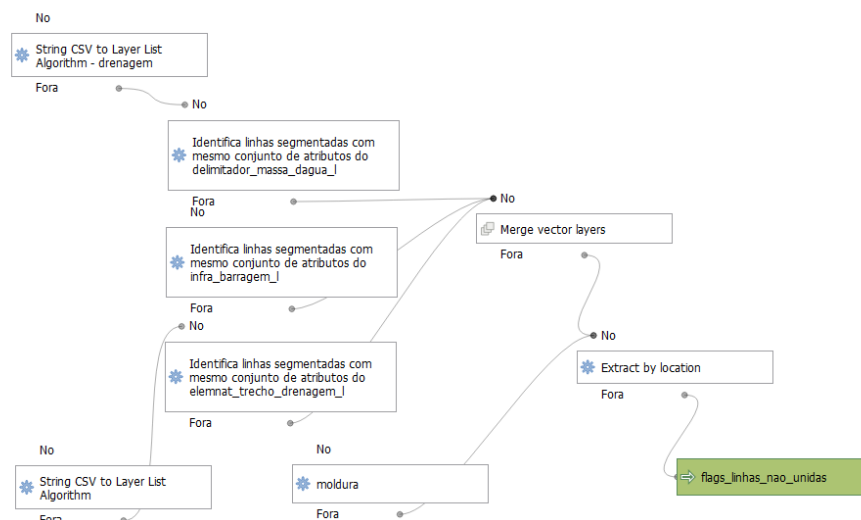
O parâmetro "**GEOGRAPHIC BOUNDARY**" se refere à delimitação geográfica para a execução do processamento em uma região determinada. A "moldura" representa a área delimitada em questão, isso evita que sejam incluídas informações irrelevantes ou imprecisas de outras áreas que estão fora da delimitação.

O parâmetro: "**FLAGS**" é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação.

nome da camada de flags: *flags_undershoot_1*

Resultado do processo e exemplo de erros:





O modelo é composto por cinco etapas:

String CSV to Layer List Algorithm: Converte uma *string* csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

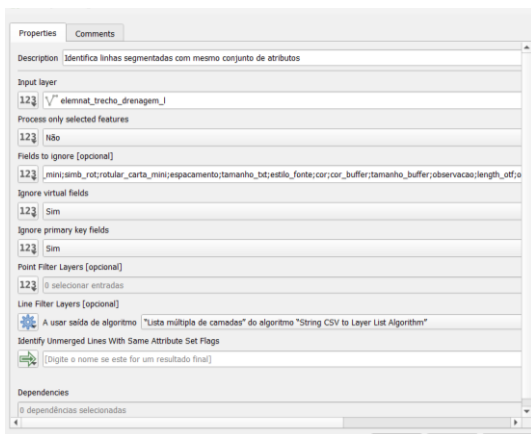
Identificar linhas Segmentadas: Identifica as linhas segmentadas com os parâmetros definidos abaixo utilizando o algoritmo *identifysplitlines*.

Merge Vector Layers: Realiza uma operação de unir as camadas de saída utilizando o algoritmo *mergevectorlayers*.

Moldura: Converte uma *string* csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*

ExtractByLocation: Extrai os ângulos pequenos que estão dentro da área delimitada pela moldura usando o algoritmo *extractbylocation*.

Parâmetros:



Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	delimitador_massa_dagua_1, elemnat_trecho_drenagem_1, infra_barragem_1,

O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como **"true"**, a validação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são validadas.

No parâmetro **"FIELDS TO IGNORE"** é definido uma lista negra de atributos (**ATTRIBUTE_BLACKLIST**) que não serão considerados na comparação das feições, a fim de evitar falsos positivos.

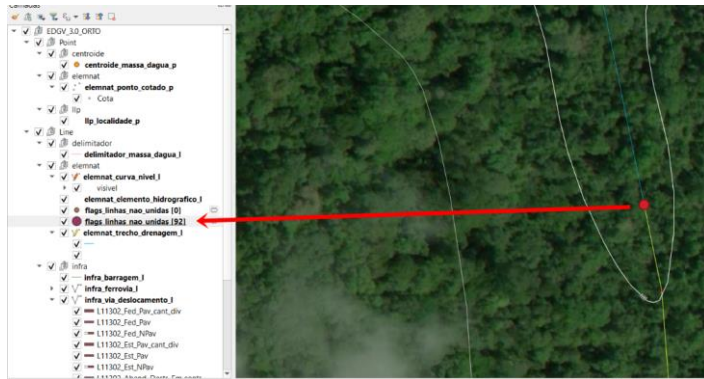
O parâmetro **"IGNORE_PK_FIELDS"** é um booleano que indica se os campos da chave primária das camadas de entrada devem ser ignorados na comparação. (apesar de estar marcado como **"true"**, o campo de chave primária **"id"** já está sendo ignorado na lista negra de atributos).

O parâmetro **"IGNORE_VIRTUAL_FIELDS"** é um booleano que indica se os campos virtuais (calculados dinamicamente) devem ser ignorados na comparação.

O parâmetro: **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação.

nome da camada de flags: flags_linhas_nao_unidas

Resultado do processo e exemplo de erros:



14- Identifica linhas não segmentadas nas intersecções:

Descrição: O algoritmo em questão tem como finalidade detectar linhas não segmentadas nas intersecções de camadas. Essa tarefa é realizada através da identificação de geometrias não conectadas que não compartilham vértices. A partir dessa análise, é possível determinar se as linhas com vértices não conectados em sua extensão devem ser classificadas como "dangles" (pontas soltas) e, portanto, descartadas, ou se devem ser consideradas como não segmentadas dentro das camadas especificadas em uma lista de camadas de linha.

Arquivo: identificar_linhas_nao_segmentadas_nas_intersecoes_alt_hid.model3

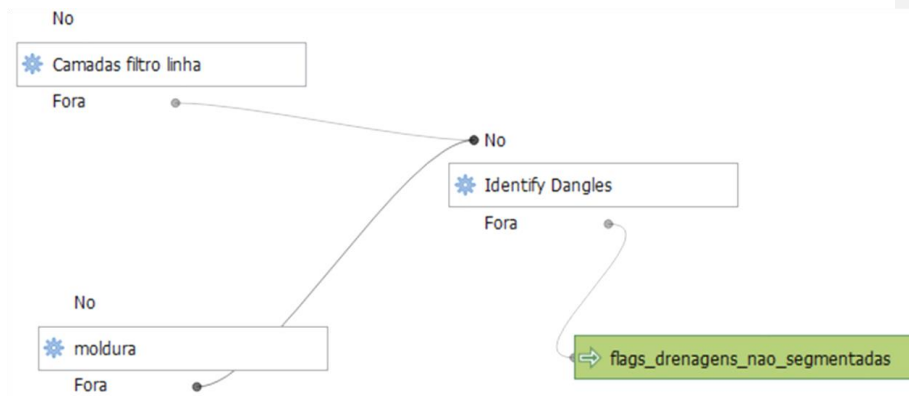
Algoritmo:

dsgtools:stringcsvtolayerlistalgorithm

dsgtools:stringcsvtofirstlayerwiththelementsalgorithm

dsgtools:identifydanglesalgorithm

Composição do Modelo:



O modelo é composto por três etapas:

Camadas filtro linha: Converte uma string csv em uma lista de camadas usando o algoritmo *stringcsvtolayerlistalgorithm*.

Moldura: Converte uma string csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwiththelementsalgorithm*.

Identify Dangles: Identifica pontas soltas na camada de elemnat_trecho_drenagem_l algoritmo *identifydanglesalgorithm*

Parâmetros:

Properties	Comments
Description: Identify Dangles	
Camada de entrada	elemnat_trecho_drenagem_l
Processar apenas feições selecionadas	Não
Input is a boundary layer (every line must be connected to an element of either the input layer or the filters)	
Input is a boundary layer	Não
Raio de busca	0,000100
Camada de filtro do tipo linha [opcional]	
A user saída de algoritmo "Lista múltipla de camadas" do algoritmo "Camadas filtro linha"	
Camadas de filtro do tipo polígono [opcional]	
A user saída de algoritmo "Loaded layer" do algoritmo "moldura"	
Ignorar pontas soltas em linhas não segmentadas	
Não	
Geographic Boundary (this layer only filters the output dangles) [opcional]	
A user saída de algoritmo "Loaded layer" do algoritmo "moldura"	
Identificar Pontas Soltas Flags	
flags_drenagens_nao_segmentadas	
Dependencies	
0 dependências selecionadas	

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	elemnat_trecho_drenagem_l delimitador_massa_dagua_l, infra_barragem_l

O parâmetro **"INPUT"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário.

O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como **"true"**, a validação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são validadas.

O parâmetro **"INPUT IS A BOUNDARY LAYER"** Se a opção **"False"** estiver definida no algoritmo, isso significa que a moldura não precisa estar conectada a elementos da camada de entrada ou dos filtros.

O Parâmetro: **"SEARCH RADIUS"** do algoritmo determina a distância máxima que será percorrida para encontrar pontas soltas em relação aos segmentos de linha que se cruzam nas intersecções. No modelo, o valor padrão do raio de busca é definido como 0,00001 unidades do sistema de coordenadas utilizado no processamento. No entanto, é importante lembrar que esse valor pode ser ajustado de acordo com as necessidades específicas, considerando a escala e precisão dos dados utilizados no processamento.

O parâmetro **"CAMADA DE FILTRO"** usados para limitar as camadas que serão analisadas no algoritmo.

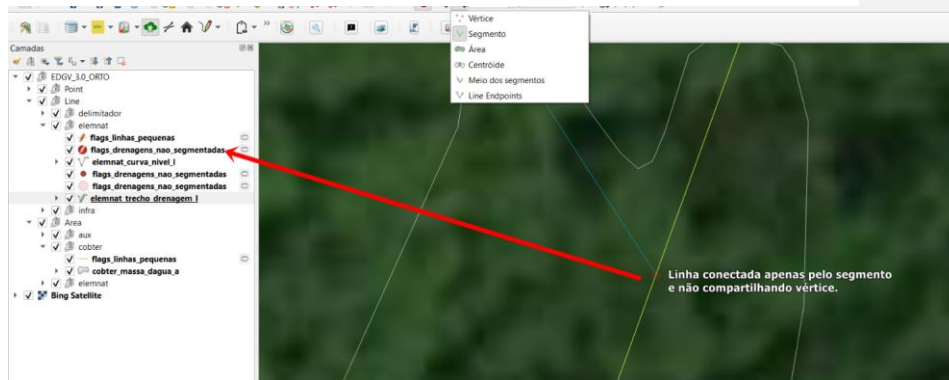
O parâmetro **"IGNORAR PONTAS SOLTAS EM LINHAS NÃO SEGMENTADAS"** quando marcado como **"False"** permite que o algoritmo encontre pontas soltas em linhas não segmentadas. Ou seja, considera como *undershoots* aquelas pontas soltas que não estão conectadas a outros segmentos de linha.

O parâmetro **"GEOGRAPHIC BOUNDARY"** se refere à delimitação geográfica para a execução do processamento em uma região determinada. A "moldura" representa a área delimitada em questão, isso evita que sejam incluídas informações irrelevantes ou imprecisas de outras áreas que estão fora da delimitação.

O parâmetro: **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação.

nome da camada de flags: *flags_undershoot_l*

Resultado do processo e exemplo de erros:



15- Identifica elementos pequenos na Rede:

Descrição: Este algoritmo realiza a identificação de elementos de pequena dimensão em uma camada pré-determinada, ao mesmo tempo em que detecta segmentos de drenagem que se encontram desconectados da rede.

Arquivo: identificar_elementos_pequenos_na_rede.model3

Algoritmo:

dsgtools:stringcsvtolayerlistalgorithm

dsgtools:batchrunalgorithm

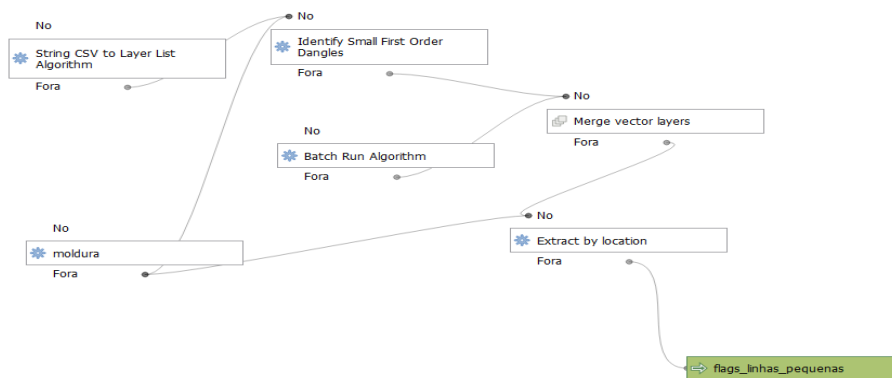
dsgtools:stringcsvtofirstlayerwiththeelementsalgorithm

dsgtools:identifysmallfirstorderdangles

native:mergevectorlayers

native:extractbylocation

Composição do Modelo:



O modelo é composto por seis etapas:

String CSV to Layer List Algorithm: Converte uma *string* csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

Moldura: Converte uma *string* csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

Batchrunalgorithm: Executa o algoritmo *identifysmalllines* com os parâmetros selecionados abaixo para as camadas de linha e separadamente executa para polígonos.

Identify Small First Order Dangles: Procura por pequenos segmentos de linha que não se conectam a outros segmentos ou estão muito próximos de uma interseção, executando o algoritmo *identifysmallfirstorderdangles*.

Merge Vector Layers: Realiza uma operação de unir as camadas de saída utilizando o algoritmo *mergevectorlayers*.

ExtractByLocation: Extrai os elementos pequenos que estão dentro da área delimitada pela moldura usando o algoritmo *extractbylocation*.

Parâmetros:

Properties

Comments

Description

Batch Run Algorithm

Names das camadas de entrada separados por vírgula

123 delimitador_massa_dagua_l,infra_barragem_l,elemnat_elemento_hidrografico_l,elemnat_trecho_drenagem_l

Nome do algoritmo com seu provedor

123 dsgrtools:identifysmalllines

Nome da chave de entrada

123 INPUT


Dicionário de parâmetros em JSON

123 {
 "SELECTED": false,
 "TOLERANCE": 0.00001,
 "FLAGS": "TEMPORARY_OUTPUT"
}

Nome dos parâmetros de saída da camada [opcional]

123 FLAGS

Execução de saída em lote

 [Digite o nome se este for um resultado final]

Dependencies

0 dependências selecionadas

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	delimitador_massa_dagua_l, infra_barragem_l, elemnat_elemento_hidrografico_l elemnat_Trecho_drenagem_l

Parâmetros em JSON: {

"SELECTED": false,

"TOLERANCE": 0.00001,

"FLAGS": "TEMPORARY_OUTPUT"

}

O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como **"true"**, a validação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são validadas.

O parâmetro **"TOLERANCE"** em questão determina a tolerância em relação à distância e ao tamanho das linhas detectadas pelo algoritmo. Quanto menor o valor, mais sensível o algoritmo será na detecção de linhas pequenas. O valor padrão é 0,00001, indicando alta sensibilidade. Valores maiores tornam o algoritmo menos sensível, enquanto valores menores o tornam mais sensível.

Comentado [GPG3]: Verificar em todas as rotinas cado o elemnat_elemento_hidrografico_l

O parâmetro **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação. Neste caso, o valor **"TEMPORARY_OUTPUT"** especifica que uma camada temporária será gerada para armazenar informações sobre as linhas pequenas encontradas durante a validação.

nome da camada de flags: flags_linhas_pequenas.

O algoritmo **"Identify Small First Order Dangles"** é uma ferramenta utilizada para identificar trechos de drenagem que não estão conectados ao restante da rede, também conhecidos como **"dangles"** ou pontas soltas. Ele busca por trechos de linha que possuem um único vértice que não é conectado a nenhum outro trecho. Para evitar a identificação de **dangles** não relevantes, o algoritmo se concentra em identificar apenas aqueles que têm um comprimento mínimo especificado:

Parâmetros:

Description

Identify Small First Order Dangles

Input layer

123 ☒ elemnat_trecho_drenagem_l

Process only selected features

123 Não

Minimum size

123 0,001000

Search radius

123 0,000100

Linestring Filter Layers [opcional]

A user saída de algoritmo "Lista múltipla de camadas" do algoritmo "String CSV to Layer List Algorithm"


Polygon Filter Layers [opcional]

123 0 selecionar entradas

Geographic Boundary (this layer only filters the output dangles) [opcional]

A user saída de algoritmo "Loaded layer" do algoritmo "moldura"

Identify Small First Order Dangles Flags

 [Digite o nome se este for um resultado final]

Dependencies

0 dependências selecionadas

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	elemnat_Trecho_drenagem_l

Comentado [GPG4]: Verificar em todas as rotinas cade o elemnat_elemento_hidrografico_l

O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como **"true"**, a validação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são validadas.

O parâmetro **"MINIMUM_SIZE"** se concentra em identificar apenas *dangles* pequenos, ou seja, aqueles que têm um comprimento mínimo de 0,001 e estão localizados dentro de um raio de pesquisa definido pelo parâmetro **"SEARCH_RADIUS"** de 0,0001. permitindo identificar possíveis problemas na rede.

Comentado [GPG5]: Preciso definir metros ou graus ou unidades do sistema de referência

O parâmetro **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação. Neste caso, o valor **"TEMPORARY_OUTPUT"** especifica que uma camada temporária será gerada para armazenar informações sobre os elementos pequenos e possíveis problemas encontrados na rede.

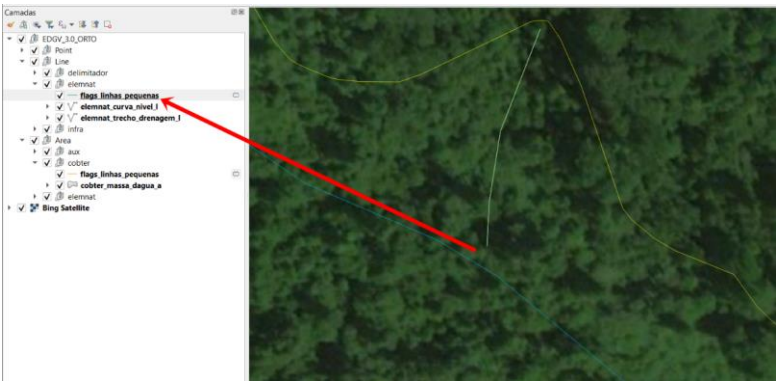
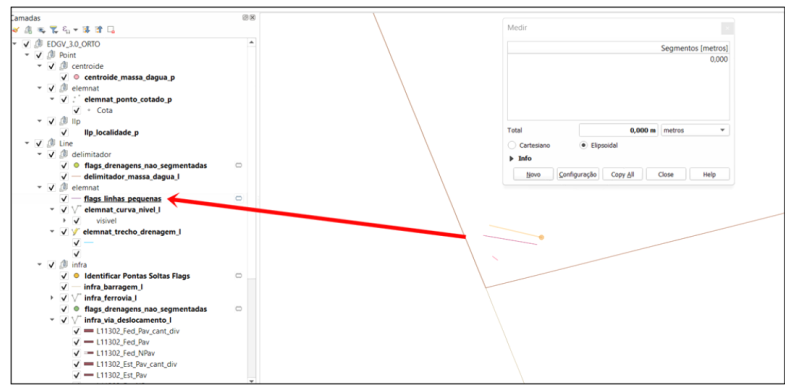
Por fim é utilizado o algoritmo de extrair pela localização para filtrar os elementos que estão dentro da moldura.

Extrair feições de: Utiliza os dados de saída do procedimento anterior.

Predicado geométrico: Uma lista que especifica o tipo de relação espacial entre as geometrias que devem ser consideradas na extração. Neste caso, o valor [0] significa que apenas as geometrias que se intersectam serão extraídas.

nome da camada de flags: flags_linhas_pequenas.

Resultado do processo e exemplo de erros:



16- Identifica erros na construção da rede de drenagem:

Descrição: Este algoritmo tem como objetivo identificar erros na construção da rede de drenagem

Arquivo: identificar_erros_rede_drenagem.model3

Algoritmos:

dsgtools:batchrunalgorithm

dsgtools:stringcsvtolayerlistalgorithm

dsgtools:stringcsvtofirstlayerwithelementsalgorithm

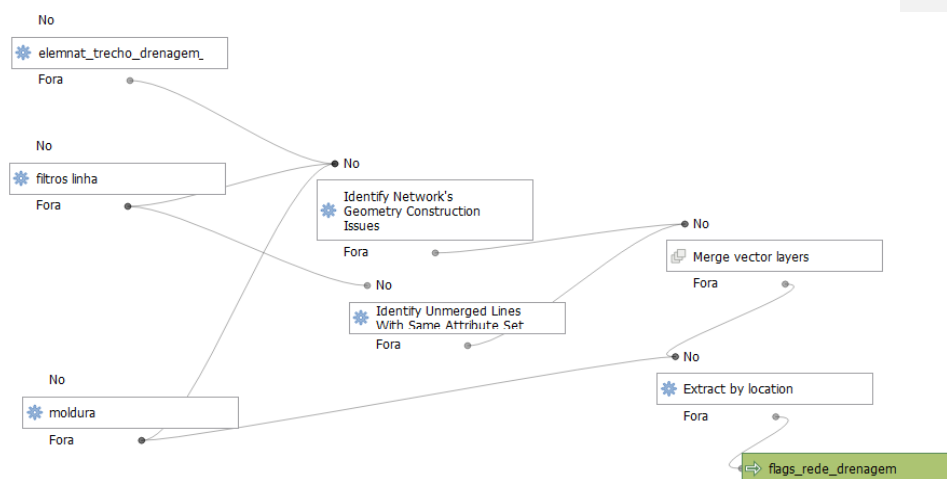
dsgtools:identifyunmergedlineswithsameattributeset

dsgtools:identifynetworkconstructionissues

native:mergevectorlayers

native:extractbylocation

Composição do Modelo:



O modelo é composto por cinco etapas:

String CSV to Layer List Algorithm: Converte uma *string* csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

Identify Network's Geometry Construction Issues: Tem por finalidade a identificação de problemas na construção da geometria da rede, em especial na detecção de potenciais erros na rede de drenagem. Seu objetivo consiste em encontrar, de maneira automatizada, ângulos agudos, pontas soltas, linhas não segmentadas desconectadas da rede utilizando o algoritmo *dsgtools:identifynetworkconstructionissues*.

Identify Unmerged Lines With Same Attribute Set: Responsável por identificar linhas não mescladas (*unmerged*) com o mesmo conjunto de atributos. Ele compara os atributos de todas as linhas dentro de uma camada especificada e verifica se duas ou mais linhas estão com o mesmo conjunto de atributos utilizando o algoritmo *dsgtools:identifyunmergedlineswithsameattributeset*.

Merge Vector Layers: Realiza uma operação de unir as camadas de saída utilizando o algoritmo *mergevectorlayers*.

ExtractByLocation: Extrai os erros que estão dentro da área delimitada pela moldura usando o algoritmo *extractbylocation*.

Parâmetros:

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

311

312

313

314

315

316

317

318

319

320

321

322

323

324

325

326

327

328

329

330

331

332

333

334

335

336

337

338

339

340

341

342

343

344

345

346

347

348

349

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

370

371

372

373

374

375

376

377

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

414

415

416

417

418

419

420

421

422

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437

438

439

440

441

442

443

444

445

446

447

448

449

450

451

452

453

454

455

456

457

458

459

460

461

462

463

464

465

466

467

468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

483

484

485

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

501

502

503

504

505

506

507

508

509

510

511

512

513

514

515

516

517

518

519

520

521

522

523

524

525

526

527

528

529

530

531

532

533

534

535

536

537

538

539

540

541

542

543

544

545

546

547

548

549

550

551

552

553

554

555

556

557

558

559

560

561

562

563

564

565

566

567

568

569

570

571

572

573

574

575

576

577

578

579

580

581

582

583

584

585

586

587

588

589

590

591

592

593

594

595

596

597

598

599

600

601

602

603

604

605

606

607

608

609

610

611

612

613

614

615

616

617

618

619

620

621

622

623

624

625

626

627

628

629

630

631

632

633

634

635

636

637

638

639

640

641

642

643

644

645

646

647

648

649

650

651

652

653

654

655

656

657

658

659

660

661

662

663

664

665

666

667

668

669

670

671

672

673

674

675

676

677

678

679

680

681

682

683

684

685

686

687

688

689

690

691

692

693

694

695

696

697

698

699

700

701

702

703

704

705

706

707

708

709

710

711

712

713

714

715

716

717

718

719

720

721

722

723

724

725

726

727

728

729

730

731

732

733

734

735

736

737

738

739

740

741

742

743

744

745

746

747

748

749

750

751

752

753

754

755

756

757

758

759

760

761

762

763

764

765

766

767

768

769

770

771

772

773

774

775

776

777

778

779

780

781

782

783

784

785

786

787

788

789

790

791

792

793

794

795

796

797

798

799

800

801

802

803

804

805

806

807

808

809

810

811

812

813

814

815

816

817

818

819

820

821

822

823

824

825

826

827

828

829

830

831

832

833

834

835

836

837

838

839

840

841

842

843

844

845

846

847

848

849

850

851

852

853

854

855

856

857

858

859

860

861

862

863

864

865

866

867

868

869

870

871

872

873

874

875

876

877

878

879

880

881

882

883

884

885

886

887

888

889

890

891

892

893

894

895

896

897

898

899

900

901

902

903

904

905

906

907

908

909

910

911

912

913

914

915

916

917

918

919

920

921

922

923

924

925

926

927

928

929

930

931

932

933

934

935

936

937

938

939

940

941

942

943

944

945

946

947

948

949

950

951

952

953

954

955

956

957

958

959

960

961

962

963

964

965

966

967

968

969

970

971

972

973

974

975

976

977

978

979

980

981

982

983

984

985

986

987

988

989

990

991

992

993

994

995

996

997

998

999

1000

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

62

63

64

65

66

67

68

69

70

71

72

73

74

75

76

77

78

79

80

81

82

83

84

85

86

87

88

89

90

91

92

93

94

95

96

97

98

99

100

101

102

103

104

105

106

107

108

109

110

111

112

113

114

115

116

117

118

119

120

121

122

123

124

125

126

127

128

129

130

131

132

133

134

135

136

137

138

139

140

141

142

143

144

145

146

147

148

149

150

151

152

153

154

155

156

157

158

159

160

161

162

163

164

165

166

167

168

169

170

171

172

173

174

175

176

177

178

179

180

181

182

183

184

185

186

187

188

189

190

191

192

193

194

195

196

197

198

199

200

201

202

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

250

251

252

253

254

255

256

257

258

259

260

261

262

263

264

265

266

267

268

269

270

271

272

273

274

275

276

277

278

279

280

281

282

283

284

285

286

287

288

289

290

291

292

293

294

295

296

297

298

299

300

301

302

303

304

305

306

307

308

309

310

311

312

313

314

315

316

317

318

319

320

321

322

323

324

325

326

327

328

329

330

331

332

333

334

335

336

337

338

339

340

341

342

343

344

345

346

347

348

349

350

351

352

353

354

355

356

357

358

359

360

361

362

363

364

365

366

367

368

369

370

371

372

373

374

375

376

377

378

379

380

381

382

383

384

385

386

387

388

389

390

391

392

393

394

395

396

397

398

399

400

401

402

403

404

405

406

407

408

409

410

411

412

413

414

415

416

417

418

419

420

421

422

423

424

425

426

427

428

429

430

431

432

433

434

435

436

437

438

439

440

441

442

443

444

445

446

447

448

449

450

451

452

453

454

455

456

457

458

459

460

461

462

463

464

465

466

467

468

469

470

471

472

473

474

475

476

477

478

479

480

481

482

483

484

485

486

487

488

489

490

491

492

493

494

495

496

497

498

499

500

501

502

503

504

505

506

507

508

509

510

511

512

513

514

515

516

517

518

519

520

521

522

523

524

525

526

527

528

529

530

531

532

533

534

5

O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como **"true"**, a verificação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são verificadas.

Comentado [GPG6]: Trocar a palavra Validada para Verificada em todos os campos SELECTED

O parâmetro **"INPUT IS A BOUNDARY LAYER"** Se a opção **"False"** estiver definida no algoritmo, isso significa que a moldura não precisa estar conectada a elementos da camada de entrada ou dos filtros.

O Parâmetro: **"SEARCH RADIUS"** do algoritmo determina a distância máxima que será percorrida para encontrar pontas soltas em relação aos segmentos de linha e polígonos que se cruzam nas intersecções. No modelo, o valor padrão do raio de busca é definido como 0,00001 unidades do sistema de coordenadas utilizado no processamento. No entanto, é importante lembrar que esse valor pode ser ajustado de acordo com as necessidades específicas, considerando a escala e precisão dos dados utilizados no processamento.

Comentado [GPG7]: Definir esse sistema de coordenada .-

O parâmetro **"FILTER LAYER"** As camadas de filtro presentes são utilizadas com o propósito de especificar as camadas que serão empregadas para filtrar a identificação de linhas que apresentem problemas de construção na rede de drenagem. As camadas filtradas são, então, empregadas para a identificação de problemas geométricos na construção da rede.

O parâmetro **"GEOGRAPHIC BOUNDARY"** se refere à delimitação geográfica para a execução do processamento em uma região determinada. A "moldura" representa a área delimitada em questão, isso evita que sejam incluídas informações irrelevantes ou imprecisas de outras áreas que estão fora da delimitação.

O parâmetro **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações de validação. Neste caso, o valor **"TEMPORARY_OUTPUT"** especifica que uma camada temporária será gerada para armazenar informações sobre os problemas de construção na rede de drenagem.

Parâmetros:

Properties	Comments
Description Identify Unmerged Lines With Same Attribute Set	
Input layer	
123	<input checked="" type="checkbox"/> elemnat_trecho_drenagem_l
Process only selected features	
123	<input type="checkbox"/> Não
Fields to ignore [opcional]	
123	amanho_bt;estilo_fonte;cor;cor_buffer;tamanho_buffer;observacao;length_off;operador_criacao;data_criacao;operador_at
Ignore virtual fields	
123	<input type="checkbox"/> Sim
Ignore primary key fields	
123	<input type="checkbox"/> Sim
Point Filter Layers [opcional]	
123	0 selecionar entradas
Line Filter Layers [opcional]	
<input checked="" type="checkbox"/> A usar saída de algoritmo "Lista múltipla de camadas" do algoritmo "filtros linha"	
Identify Unmerged Lines With Same Attribute Set Flags	
<input checked="" type="checkbox"/> [Digite o nome se este for um resultado final]	
Dependencies	
0 dependências selecionadas	

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	elemnat_trecho_drenagem_l

O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como **"true"**, a validação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são validadas.

No parâmetro **"FIELDS TO IGNORE"** é definido uma lista negra de atributos (**ATTRIBUTE_BLACKLIST**) que não serão considerados na comparação das feições, a fim de evitar falsos positivos.

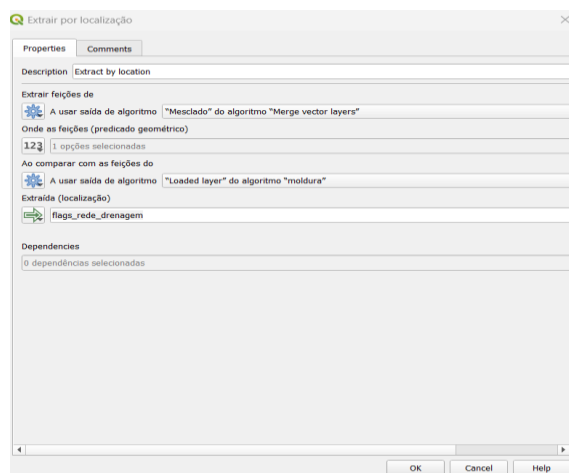
O parâmetro **"IGNORE_PK_FIELDS"** é um booleano que indica se os campos da chave primária das camadas de entrada devem ser ignorados na comparação. (apesar de estar marcado como **"true"**, o campo de chave primária **"id"** já está sendo ignorado na lista negra de atributos).

O parâmetro **"IGNORE_VIRTUAL_FIELDS"** é um booleano que indica se os campos virtuais (calculados dinamicamente) devem ser ignorados na comparação.

O parâmetro **"FILTER LAYER"** As camadas de filtro presentes são usadas para especificar camadas que serão usadas para identificar linhas que não foram unidas corretamente.

O parâmetro **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações que necessitam de correção. Neste caso, o valor **"TEMPORARY_OUTPUT"** especifica que uma camada temporária será gerada para armazenar informações sobre as linhas que não foram unidas corretamente.

Por fim é utilizado o algoritmo de extrair pela localização para filtrar os elementos que estão dentro da moldura.



Extrair feições de: Utiliza os dados de saída do procedimento anterior.

Predicado geométrico: Uma lista que especifica o tipo de relação espacial entre as geometrias que devem ser consideradas na extração. Neste caso, o valor [0] significa que apenas as geometrias que se intersectam serão extraídas.

nome da camada de flags: flags_rede_drenagem.

Resultado do processo e exemplo de erros:



17- Identifica erros na construção das curvas de nível:

Descrição: Este modelo tem por objetivo identificar falhas na construção de curvas de nível, as quais são caracterizadas por apresentarem equidistâncias específicas entre seus pontos. Para tanto, torna-se necessária a adaptação do modelo a fim de que seja inserido o arquivo de acordo com a escala utilizada pelo usuário. Tendo em vista que somente a equidistância varia entre diferentes escalas e para fins didáticos, será apresentado um exemplo de modelo para a escala de 1:50.000.

Arquivos: identificar_erros_na_construcao_das_curvas_de_nivel_25k.model3
identificar_erros_na_construcao_das_curvas_de_nivel_50k.model3
identificar_erros_na_construcao_das_curvas_de_nivel_100k.model3
identificar_erros_na_construcao_das_curvas_de_nivel_250k.model3

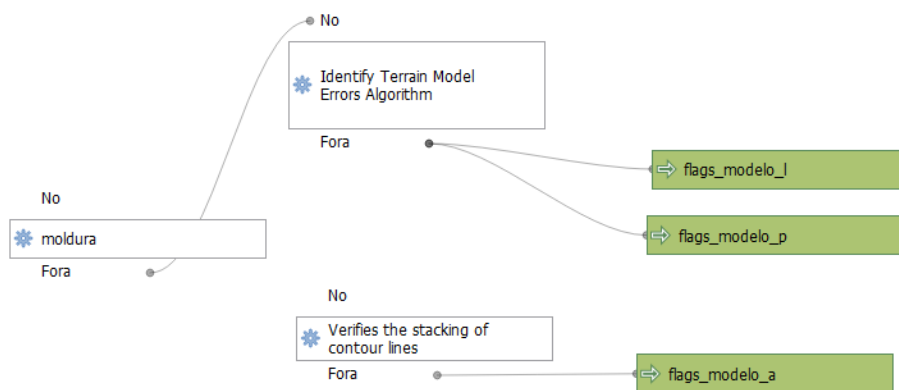
Algoritmos:

dsgtools:stringcsvtofirstlayerwithelementsalgorithm

dsgtools:identifyterrainmodelerrorsalgorithm

dsgtools:verifycountourstacking

Composição do Modelo:



O modelo é composto por três etapas:

String CSV to Layer List Algorithm: Converte uma *string* csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

Identify Terrain Model Errors: Tem como função identificar erros na modelagem do terreno, como por exemplo erros na construção de curvas de nível. Ele utiliza a camada específica de curvas de nível como entrada e verifica se as equidistâncias entre as curvas estão corretas de acordo com os valores definidos para cada escala utilizando o algoritmo *identifyterrainmodelerrorsalgorithm*.

Verify Countour Stacking: (Verificar Empilhamento de Curvas de Nível) tem como objetivo verificar se as curvas de nível estão empilhadas corretamente, ou seja, se não há sobreposição utilizando o algoritmo *verifycountourstacking*.

Parâmetros:

Properties

Comments

Description

Identify Terrain Model Errors Algorithm

Input contour layer

123 elemnat_curva_nivel_1

Processar apenas feições selecionadas

123 Não

Campo de valor da cota

123 cota

Tolerância

@equidistancia

Camada de limite geográfico

A usar saída de algoritmo "Loaded layer" do algoritmo "moldura"

Algoritmo de Identificação de Erros na Modelagem do Terreno Flags do tipo ponto

flags_modelo_p

Algoritmo de Identificação de Erros na Modelagem do Terreno Flags do tipo linha

flags_modelo_l

Dependencies

0 dependências selecionadas

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	elemnat_curva_nivel_1

O parâmetro **"SELECTED"** determina se a validação deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como **"true"**, a verificação é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são verificadas.

O parâmetro **"CAMPO DE VALOR DA COTA"** requer que seja fornecida a coluna contendo os valores de cota. Tal entrada é utilizada para detecção de erros na construção das curvas de nível, por meio da comparação dos valores de cota.

Comentado [GPG8]: Trocar a palavra Validada para Verificada em todos os campos SELECTED



O Parâmetro: **"TOLERÂNCIA"** o algoritmo utiliza a variável @equidistancia, que representa a distância entre as curvas de nível no modelo e tem um valor pré-definido de 20 metros para a escala 1:50000. Esse valor pode variar de acordo com o arquivo do modelo utilizado. Se a diferença entre as curvas de nível adjacentes não estiver de acordo com a tolerância, determinada por @equidistancia, o algoritmo identifica esses pontos como erros e gera *"flags"* para indicá-los.

O parâmetro **"CAMADA DE LIMITE GEOGRÁFICO"** se refere à delimitação geográfica para a execução do processamento em uma região determinada. A "moldura" representa a área delimitada em questão, isso evita que sejam incluídas informações irrelevantes ou imprecisas de outras áreas que estão fora da delimitação.

O parâmetro **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações dos lugares onde necessitam de correção.

nome da camada de flags: flag_modelo_p, flag_modelo_l

Parâmetros:

Properties	Comments
Description	
Verifies the stacking of contour lines	
Selecinar	
123	✓ elemnat_curva_nivel_l
Layer de curva de nível	
123	cota
Atributo que informa a depressão	
123	depressao
Equidistância	
	@equidistancia
Process is running inside model	
123	Sim
Flags	
	flag_modelo_a
Dependencies	
0 dependências selecionadas	

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Linha	elemnat_curva_nivel_1

O parâmetro **"LAYER DE CURVA DE NIVEL"** requer que seja fornecida a coluna contendo os valores de cota. Tal entrada é utilizada para detecção de erros na construção das curvas de nível, por meio da comparação dos valores de cota.

O Parâmetro: **"ATRIBUTO QUE INFORMA A DEPRESSÃO"** requer que seja fornecida a coluna contendo os valores de depressão.

O Parâmetro: **"EQUIDISTÂNCIA"** o algoritmo utiliza a variável @equidistancia, que representa a distância entre as curvas de nível no modelo e tem um valor pré-definido de 20 metros para a escala 1:50000. Esse valor pode variar de acordo com o arquivo do modelo utilizado.

O parâmetro **"CAMADA DE LIMITE GEOGRÁFICO"** se refere à delimitação geográfica para a execução do processamento em uma região determinada. A "moldura" representa a área delimitada em questão, isso evita que sejam incluídas informações irrelevantes ou imprecisas de outras áreas que estão fora da delimitação.

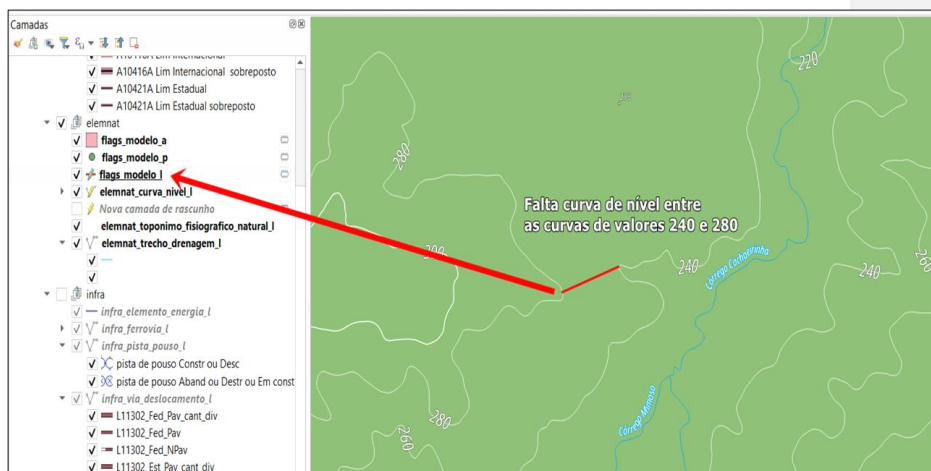
O parâmetro **"RUNNING INSIDE MODEL"** Quando definido como **"True"**, o algoritmo ajusta seu comportamento para lidar com os parâmetros de entrada que foram fornecidos por outros algoritmos dentro do mesmo modelo de processamento.

O parâmetro **"FLAGS"** é utilizado para definir o nome da camada que será criada pela rotina para armazenar as informações dos lugares onde necessitam de correção.

nome da camada de flags: flag_modelo_a.

Comentado [GPG9]: A Rotina de Verify stacking não ta gerando output, verificar como ela funciona e como gerar a Flag de area presente no modelo

Resultado do processo e exemplo de erros:



18- Fechar polígonos de corpos de água e Ilha:

Descrição: Este modelo tem como objetivo a geração de polígonos de massa d'água e ilha por meio da manipulação vetorial, utilizando a camada delimitadora de linha como entrada. Destaca-se que este algoritmo é o único responsável pela manipulação dos vetores.

Arquivos: fechar_poligonos_massa_dagua_e_ilha.model3

Algoritmos:

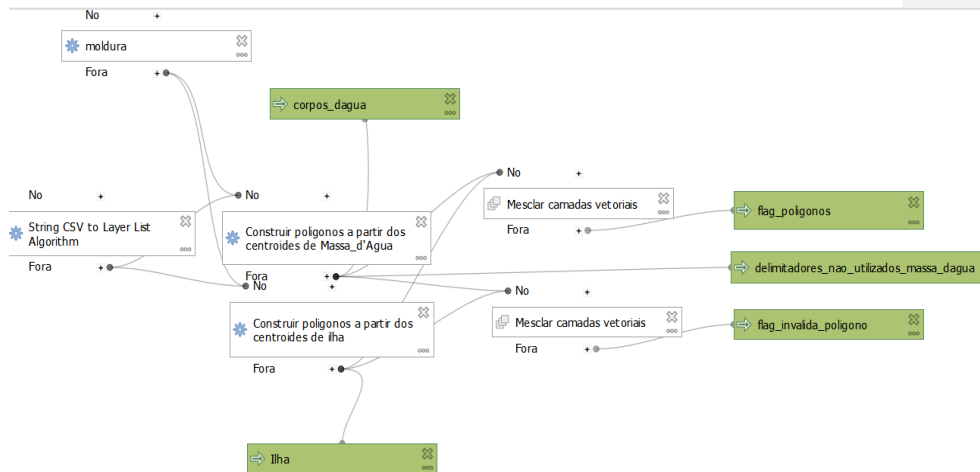
dsgtools:stringcsvtofirstlayerwithelementsalgorithm

dsgtools:stringcsvtolayerlistalgorithm

dsgtools:buildpolygonsfromcenterpointsandboundariesalgorithm

native:mergevectorlayers

Composição do Modelo:



O modelo é composto por quatro etapas:

String CSV to Layer List Algorithm: Converte uma *string* csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

Moldura: Converte uma *string* csv em uma camada especificada usando o algoritmo *stringcsvtofirstlayerwithelementsalgorithm*.

Construir polígonos a partir dos centroides: O algoritmo em questão tem como objetivo criar polígonos que representem as massas d'água e de forma separada as ilhas a partir de linhas de restrição (delimitadores), as quais delimitam os limites do polígono a ser construído. Para tanto, utiliza-se um ponto central que contém as informações principais acerca da geometria a ser gerada. Ademais, o polígono recém-construído herda os atributos do ponto de centroide, este procedimento é realizado utilizando o algoritmo:

buildpolygonsfromcenterpointsandboundariesalgorithm.

Merge Vector Layers: Realiza uma operação de unir as camadas de saída utilizando o algoritmo *mergevectorlayers*.

Parâmetros:

The screenshot shows the 'Construir polígonos a partir de centroides e linhas' (Build polygons from centroids and lines) algorithm interface. The parameters are as follows:

- Camada de ponto central:** centroide_massa_dagua_p
- Processar apenas feições selecionadas:** Não
- Campos à ignorar [opcional]:** o;tamanho_bt;estilo_fonte;cor;cor_buffer;tamanho_buffer;observacao;length_off;operador_criacao;data_criacao;operador_atualizacao;data_atualizacao
- Line Boundary [opcional]:** delimitador_massa_dagua_l
- Delimitadores do tipo linha [opcional]:** A usar saída de algoritmo "Lista múltipla de camadas" do algoritmo "String CSV to Layer List Algorithm"
- Delimitadores do tipo polígono [opcional]:** 0 selecionar entradas
- Limite geográfico [opcional]:** A usar saída de algoritmo "Loaded layer" do algoritmo "moldura"
- Merge output polygons with same attribute set:** Não
- Check output polygons for invalid geometries:** Sim
- Suppress area without center point flag:** Sim
- Polígonos de saída:** corpos_dagua
- Invalid Polygon Location Flags from Construir polígonos a partir de centroides e linhas:** flag_invalida_poligono
- Unused Boundary Flags from Construir polígonos a partir de centroides e linhas:** delimitadores_nao_utilizados
- Polygon Flags from Construir polígonos a partir de centroides e linhas:** flags_poligonos
- Dependencies:** 0 dependências selecionadas

Camadas de entrada:

Tipo de geometria	Camadas de entrada
Ponto	centroide_massa_dagua_p centroide_ilha_p

O parâmetro **"SELECTED"** determina se a geração dos polígonos deve ser realizada apenas nas geometrias selecionadas pelo usuário. Quando marcado como **"true"**, a execução é realizada apenas nas geometrias selecionadas. Caso contrário, todas as geometrias são processadas.

No parâmetro **"CAMPOS A IGNORAR"** é definido uma lista negra de atributos (*attribute blacklist*) que não serão considerados na hora de herdar os atributos dos pontos de centroide. O Parâmetro **"LINE BOUNDARY"** é usado para definir a camada de linha que será usada como limite (delimitadores) para criar polígonos a partir dos centroides.

Nos parâmetros **"DELIMITADORES DO TIPO LINHA E POLIGONO"** é usado para definir as demais camadas de linhas ou de polígonos que será usada como limite (delimitadores) para criar os polígonos a partir dos centroides.

O parâmetro **"CAMADA DE LIMITE GEOGRÁFICO"** se refere à delimitação geográfica para a execução do processamento em uma região determinada. A "moldura" representa a área delimitada em questão, isso evita que sejam incluídas informações irrelevantes ou imprecisas de outras áreas que estão fora da delimitação.

O parâmetro **"MERGE OUTPUT POLYGONS"** com o valor *"False"* no algoritmo indica que os polígonos resultantes não serão mesclados em um único polígono, mas sim mantidos separados.

O parâmetro **"CHECK OUTPUT POLYGONS"** é usado para verificar se os polígonos produzidos são válidos ou não. quando definido *"True"*, o algoritmo verifica se há polígonos inválidos na saída e se houver retorna um conjunto de geometrias inválidas na *"flag"* de parâmetro: *"INVALID_POLYGON_LOCATION"*.

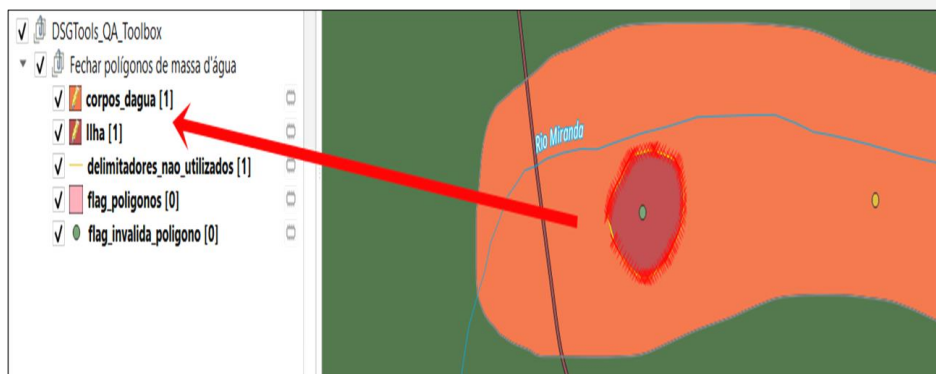
O parâmetro **"SUPRESS AREA WITHOUT CENTER POINT FLAG"** com o valor *"True"*, indica que áreas sem centroide devem ser suprimidas na saída. Ou seja, somente as áreas que possuem um ponto central definido serão consideradas na construção dos polígonos. Esse parâmetro é utilizado para evitar a criação de polígonos em áreas sem informação.

O parâmetro **"FLAGS"** é composto por dois tipos distintos de informações. O primeiro tipo refere-se às localizações que necessitam de correção, enquanto o segundo tipo se relaciona às camadas de massa d'água e de ilhas recém-criadas que serão posteriormente copiadas.

nome da camada de flags de correção: flag_poligonos, flag_invalida_poligono

nome da camada de flags das camadas criadas: Massa_dagua, ilha

Resultado do processo e exemplo de erros:



Rotinas executadas fora do fluxo de validação

Rotina de validade geométrica

Arquivo: rotinas_validade_geometrica_alt_hid.model3

Provedor: dsgtools:identifyandfixinvalidgeometries,dsgtools:identifymultigeometries,dsgtools:identifyduplicatedfeatures

Descrição: Tem como objetivo consolidar as rotinas 1 a 3, ou seja, as etapas de Identificação de geometrias inválidas, Identificação de geometrias com mais de uma parte e Identificação de feições duplicadas.