

# 2. Vectorização Simples e Avançada no QGIS 3

SCAN ME



## TUTORIAL

preparado por Nuno de Santos Loureiro  
DCTMA - FCT - Universidade do Algarve  
[nloure@ualg.pt](mailto:nloure@ualg.pt)

WORK IN  
PROGRESS  
CHECK BACK SOON!

# SUMÁRIO

O presente **TUTORIAL QGIS 3** tem como objectivos:

1. apresentar as **ferramentas simples de edição**, e também as ferramentas **avançadas**, para vectorizar **pontos, linhas e polígonos** numa **shapefile**, a partir de layers em WMS ou de imagens previamente georreferenciadas e adicionadas a um projecto QGIS 3;
2. apresentar os **procedimentos avançados** que permitem assegurar a qualidade e rigor da vectorização e, simultaneamente, o cumprimento das exigências geométricas e topológicas que asseguram a perfeição da **shapefile** produzida;
3. apresentar os **procedimentos de verificação geométrica e topológica** que permitem validar toda a vectorização elevada a cabo e considerar um trabalho concluído e pronto a difundir...
4. apresentar uma ferramenta, o [mapshaper.org](http://mapshaper.org), que pode ajudar a corrigir de forma automática erros geométricos e topológicos...



# VECTORIZAÇÃO

A VECTORIZAÇÃO é a criação de entidades vectoriais (pontos, linhas ou polígonos) numa *shapefile*, através de funcionalidades próprias do **QGIS 3**. A vectorização pode ser feita através de **ferramentas simples de edição**, as quais podem ser, por exemplo, potenciadas com **ferramentas avançadas de edição**.

## FERRAMENTAS SIMPLES DE EDIÇÃO

Icon	Purpose	Icon	Purpose
	Current edits		Toggle editing
	Save layer edits		
	Add new record		Add Feature: Capture Point
	Add Feature: Capture Line		Add Feature: Capture Polygon
	Vertex Tool (All Layers)		Vertex Tool (Current Layer)
	Modify the attributes of all selected features simultaneously		
	Delete Selected		Cut Features
	Copy Features		Paste Features
	Undo		Redo



# VECTORIZAÇÃO

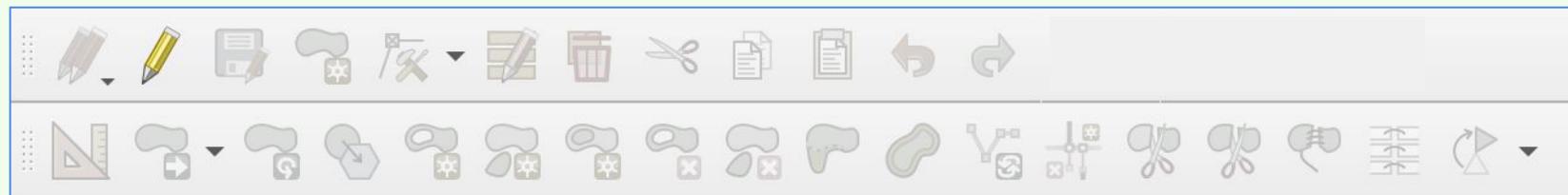
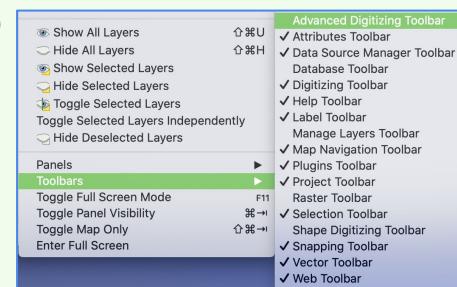
As **ferramentas simples de edição** estão sempre disponíveis, desde que a *shapefile* esteja em modo de edição. As **ferramentas avançadas de edição** activam-se através de Toolbars como o **Advanced Digitizing Toolbar**.



ferramentas  
simples  
de edição

**Toolbar** das ferramentas simples de edição.

## FERRAMENTAS AVANÇADAS DE EDIÇÃO



ferramentas  
avançadas  
de edição

**Toolbar** das ferramentas simples de edição em cima e das ferramentas avançadas de edição em baixo.



# VECTORIZAÇÃO

## FERRAMENTAS AVANÇADAS DE EDIÇÃO

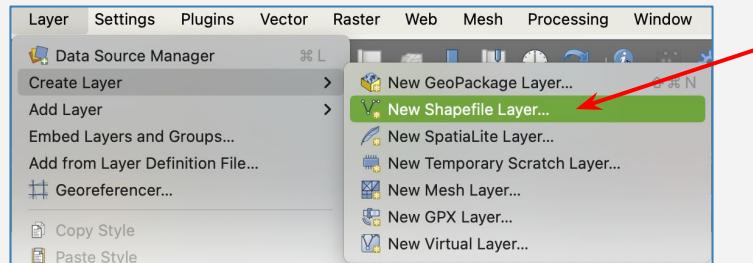
Icon	Purpose	Icon	Purpose
	Enable Advanced Digitizing Tools		Enable Tracing
	Move Feature(s)		Copy and Move Feature(s)
	Rotate Feature(s)		Simplify Feature
	Add Ring		Add Part
	Fill Ring		Swap direction
	Delete Ring		Delete Part
	Offset Curve		Reshape Features
	Split Parts		Split Features
	Merge Attributes of Selected Features		Merge Selected Features
	Rotate Point Symbols		Offset Point Symbols
	Trim or Extend Feature		

# VECTORIZAÇÃO BÁSICA - PONTOS

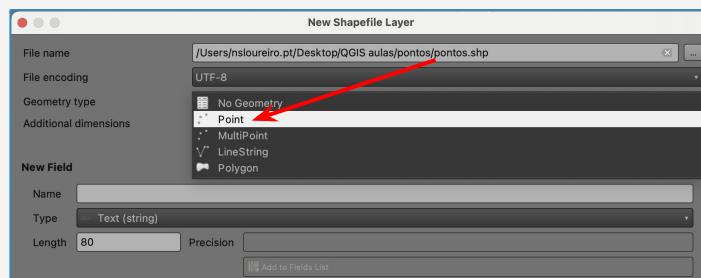
Um primeiro passo, prévio à vectorização de pontos, é a criação da *shapefile* onde essa **informação** vai ser registada e armazenada.

A criação da *shapefile* é feita em

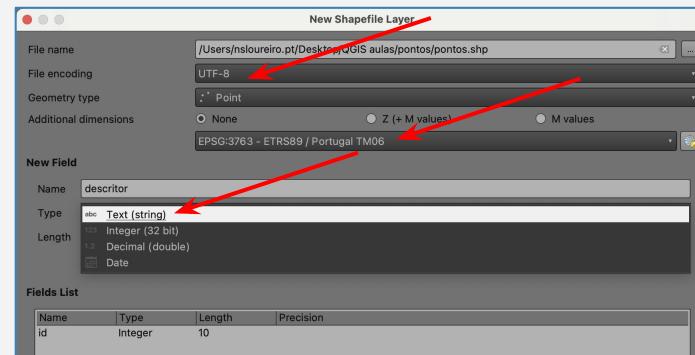
**Layer >>> Create Layer >>> new Shapefile Layer...**



É necessário criar o conjunto de ficheiros da *shapefile* e depois configurar a própria *shapefile*. Os primeiros elementos da configuração são o **File encoding** (UTF-8 ou ISO 8859-1 para a língua portuguesa) e o **Geometry type** (No Geometry - **Point** - Multipoint - Line - Polygon).

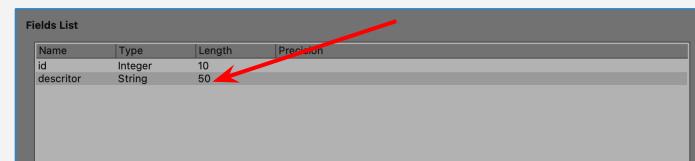


Em seguida é necessário configurar se existem ou não **Additional dimensions** (Z e/ou M values) e também o **EPSG** da *shapefile* (3763 para Portugal continental).



Por fim é necessário criar os campos (**Fields**) que vão existir na tabela de atributos (**Attribute Table**). Por defeito está criado o campo **id**, mas podem ser criados todos os outros necessários.

Os campos podem ser de quatro tipos (**Text Data - Whole Number - Decimal Number - Date**) e, em função do tipo, há especificidades de comprimento máximo de caracteres atribuídos e de grau de precisão).



Para que o novo campo seja adicionado à tabela de atributos é necessário clicar em **Add to Fields List**.

De notar que enquanto que o tipo de geometria é uma característica que é escolhida na configuração inicial, e não pode ser alterada posteriormente, os campos podem ser alterados em qualquer momento, de acordo com as necessidades...



# VECTORIZAÇÃO BÁSICA - PONTOS

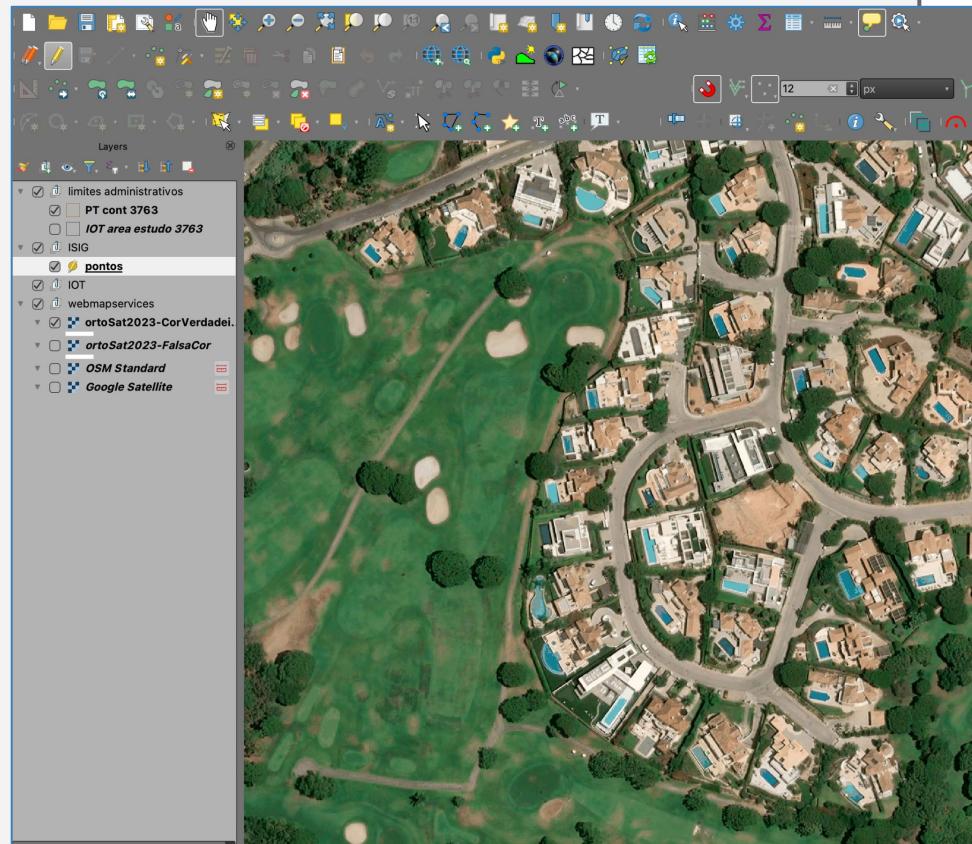
Para que se possa fazer a vectorização é necessário que a *shapefile* seja a **seleccionada** e que esteja em **modo de edição**.

**Seleccionar a *shapefile*** é clicar sobre ela. Colocar em **modo de edição** é clicar no lápis amarelo, que passa a estar também visível sobre o símbolo da própria *shapefile*. Quando a *shapefile* está em modo de edição, alguns **botões** para a edição passam a estar activos...



Na vectorização de pontos, à semelhança do que acontece com as vectorizações de linhas e de polígonos, existem sempre dois passos.

O primeiro é o da **componente gráfica** e o segundo é o da **tabela de atributos**. Para criar ou adicionar um novo ponto basta activar o **botão criar um ponto**, movimentar o cursor até que a mira esteja na posição correcta e, por fim, clicar...



Ainda na componente gráfica, o **botão alterar um ponto existente** permite deslocar o ponto para outra posição e permite também eliminar esse ponto. O botão **mudar a posição de um ponto** permite exactamente isso, ou seja, deslocar um ponto para outra posição...

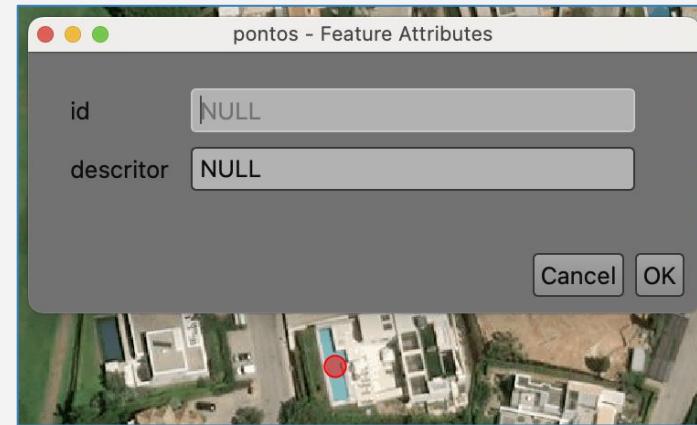
Depois de concluída a criação de um ponto, passa-se à **tabela de atributos**.

# VECTORIZAÇÃO BÁSICA - PONTOS

Neste exemplo vão-se vectorizar piscinas, de forma elementar, através de pontos. Depois de activado o **botão criar um ponto** e assinalada a posição de uma piscina, surge a **tabela de atributos** dessa mesma ocorrência ou **feature**.

Os atributos a preencher e o modo do seu preenchimento estão directamente relacionados com as configurações estabelecidas no momento da criação da *shapefile*. O **atributo id**, que é um número único de identificação da *feature*, não precisa de ser preenchido. Pode sê-lo, de forma semi-automática, num momento posterior.

O **atributo descriptor**, configurado como de texto e com um número máximo de 50 caracteres, pode ser preenchido livremente, escrevendo na janela, ou pode ser preenchido através da escolha de uma lista pré-estabelecida...



## CONFIGURAÇÃO DA LISTA PRÉ-ESTABELECIDA (*DROP-DOWN LIST*) PARA UM ATRIBUTO DA TABELA DE ATRIBUTOS

A primeira tarefa é criar a lista pré-estabelecida de possíveis preenchimentos, geralmente chamada de **CATÁLOGO**. É, em síntese, uma lista completa de todos as opções possíveis para preencher esse atributo, e é necessário ter sempre presente que é possível adicionar novas opções mas não é simples alterar ou eliminar opções anteriormente estabelecidas.

Neste exemplo vão-se considerar quatro classes:

- piscina descoberta
- piscina com cobertura
- a confirmar
- NULL

As três primeiras classes são óbvias e a quarta classe permite deixar o atributo sem estar preenchido.

Para criar a ***drop-down list*** é necessário ir a

[Layer Properties >>> Attributes Form](#)

e depois escolher o atributo **descriptor**.





# VECTORIZAÇÃO BÁSICA - PONTOS

**CONFIGURAÇÃO DA LISTA  
PRÉ-ESTABELECIDA  
(DROP-DOWN LIST) PARA UM  
ATRIBUTO DA TABELA DE  
ATRIBUTOS**

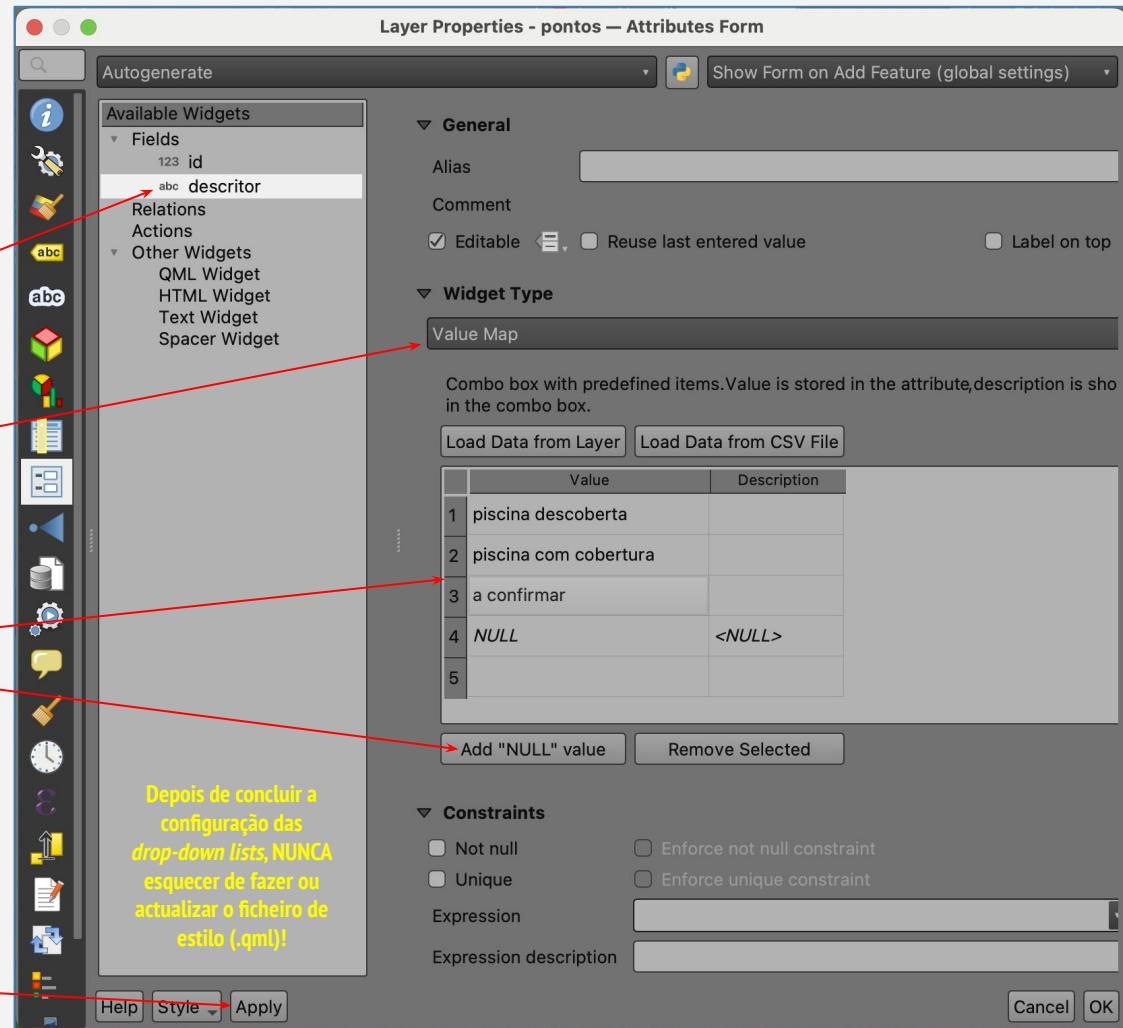
selecção do atributo  
(descriptor)

selecção do Widget  
Type: Value Map

preenchimento do  
catálogo e no final  
inclusão da classe  
NULL com o botão  
ADD “NULL” value

o preenchimento também pode ser feito  
a partir de valores já existentes na  
shapefile ou através de um ficheiro .csv

no final **Apply** e  
depois **OK**...



A utilização de **drop-down lists** torna o preenchimento da tabela de atributos muito mais **rápida** e **não sujeita a erros** de digitalização !!!

# VECTORIZAÇÃO BÁSICA - PONTOS



já estão criadas 26 ocorrências (features)

The screenshot shows the QGIS interface with two windows open. On the left is the 'Field Calculator' dialog for the 'pontos' layer. It has several options: 'Only update 0 selected features' (unchecked), 'Create a new field' (checked), 'Update existing field' (checked), 'Create virtual field' (unchecked). The 'Output field name' is set to '123 id', 'Output field type' is '123 Integer (32 bit)', and 'Output field length' is '10' with 'Precision' at '3'. In the 'Expression' tab, the expression '@id+1' is entered. A yellow arrow points from the text '@id+1' to the variable list on the right. The variable list includes 'Record and Attributes', 'Sensors', 'String', 'Variables' (with 'row\_number', 'feature', 'geometry', 'id', etc.), and 'Geometry'. On the right is the attribute table for the 'pontos' layer, showing 26 features. The columns are 'id' and 'descritor'. Most rows have 'NULL' in the 'id' column and various values in the 'descritor' column like 'piscina descoberta', 'piscina com cobe', and 'a confirmar'. A red arrow points from the 'id' column header in the table to the 'id' field in the Field Calculator's output field name.

Para preencher de forma semi-automática o atributo id basta ir a Field Calculator. Depois:

- Update existing field
- Expression @id+1
- OK

Não esquecer que antes de fazer o preenchimento é necessário gravar os pontos já vectorizados (de outra forma os id podem não ser correctamente atribuídos) e que depois de fazer o preenchimento é necessário fazer novamente a gravação da shapefile...

O +1 está associado ao primeiro valor de id que vai ser preenchido. Com o número 1 os valores de id começam em 1; para começar, por exemplo, em 21, escrever @id+21...

# VECTORIZAÇÃO BÁSICA - PONTOS

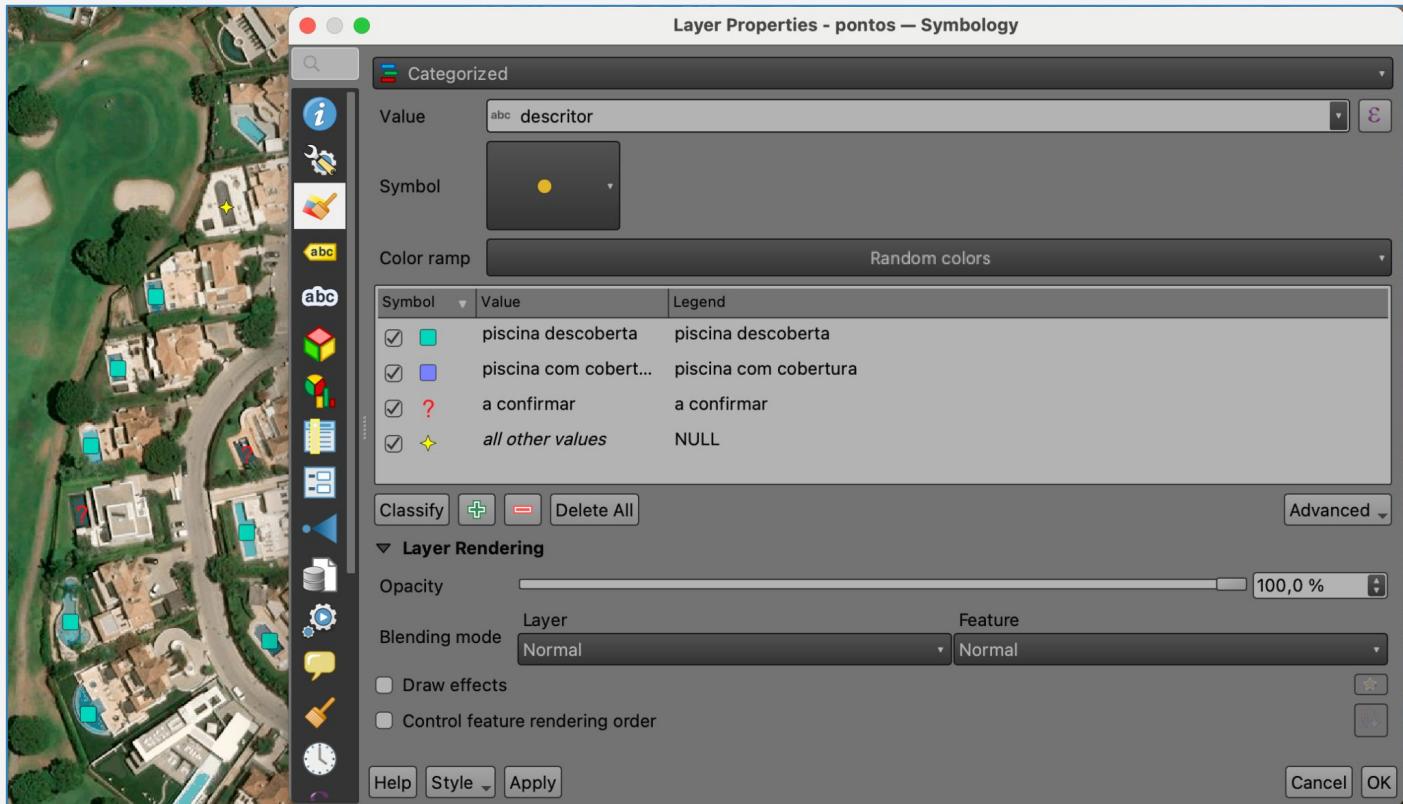
A drop-down list não interfere na simbologia que permite, na componente gráfica da shapefile, representar de forma diferenciada cada categoria.

Essa tarefa é executada em

[Layer Properties >>>  
Symbology](#)

e depois existem configurações muito diversificadas em função de cada shapefile e atributo...

Neste exemplo a simbologia é aplicada aos valores do atributo descritor e os símbolos criados estão ao lado...



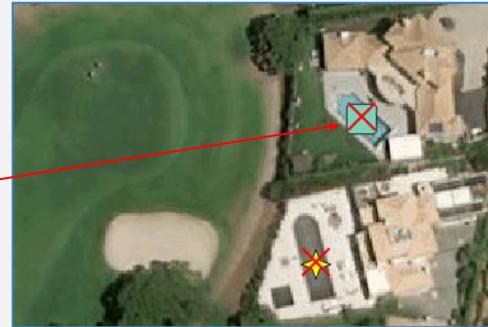


# VECTORIZAÇÃO BÁSICA - PONTOS

**MUDAR A POSIÇÃO DE UM PONTO** é uma tarefa muito simples. Basta activar o botão correcto, clicar sobre o ponto em questão, que fica destacado, movimentar o cursor para a nova posição e clicar sobre ela...



ponto mal  
posicionado



ponto bem  
posicionado



ponto já  
selecionado para  
ser movimentado



o cursor já está  
sobre a posição  
correcta

**ELIMINAR UM PONTO** também é uma tarefa muito simples. Basta activar o botão correcto, clicar sobre o ponto em questão, que fica destacado, e carregar nas teclas **Escape** ou **Delete** do teclado do computador...



ponto já  
selecionado para  
ser eliminado



piscina descoberta

# VECTORIZAÇÃO BÁSICA - PONTOS

**EDITAR OS ATRIBUTOS DE UM PONTO** é uma tarefa que pode, também, ser feita de diversas formas. Uma das mais simples, quando se quer saber como está preenchida a tabela de atributos relativa a uma ocorrência (*feature*) é a partir do botão **Identify Features**, que está na barra superior de botões.

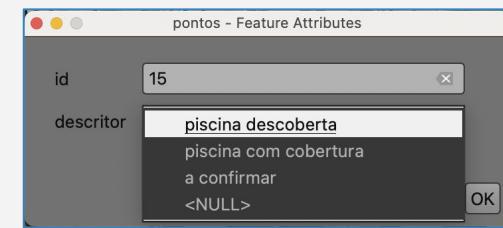
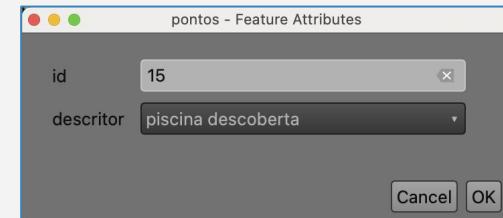


Com o botão seleccionado basta clicar sobre uma ocorrência e surge do lado direito uma janela com os atributos e os seus valores.

Identify Results	
Feature	Value
<b>pontos</b>	
descriptor	piscina descoberta
(Derived)	
(Actions)	
id	15
descriptor	piscina descoberta

No exemplo acima está-se a consultar uma *shapefile* cujo nome é “**pontos**”, cujo **atributo** utilizado para a **simbologia** é “descriptor” e que tem, no total, **dois atributos** (“**id**” e “**descriptor**”).

Para **editar os atributos** é necessário que, previamente, a *shapefile* esteja em **modo de edição**. Depois basta clicar no botão **Edit feature form...**



Surge, então, a janela da tabela de atributos, que pode ser alterada (ou não alterada) livremente, como no caso do **atributo id**, ou que pode ser alterada (ou não) através da *drop-down list*, como no caso do **atributo descriptor**...

Uma outra forma de editar e alterar os atributos é através da sua selecção através de funcionalidades de [Select features using an expression...](#)

id		=	Select features using an expression	Update All	Update Selected
5		1	piscina descoberta		
6		2	piscina descoberta		
7		3	piscina descoberta		
8		4	piscina descoberta		

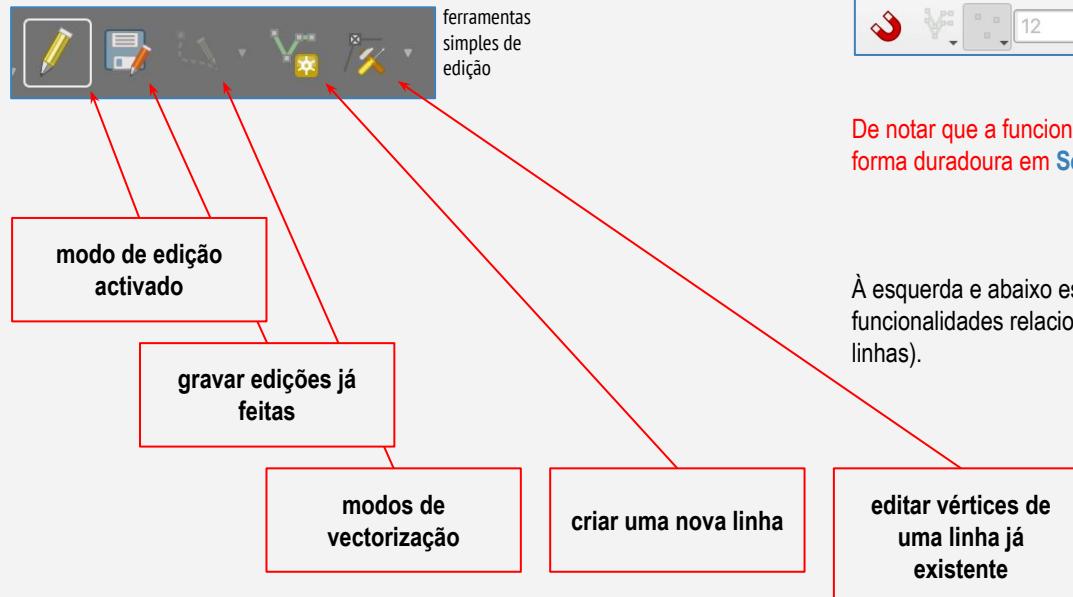
# VECTORIZAÇÃO BÁSICA - LINHAS

A criação da *shapefile* para a vectorização de linhas é idêntica à da de pontos. A única diferença reside do **Geometry type** (No Geometry - Point - Multipoint - Line - Polygon), que deverá ser de linhas em vez de pontos.

A vectorização de linhas tem de ter sempre presente o conceito de linha numa *shapefile*: é um segmento de recta entre dois pontos, denominados vértices.



Assim, **vectorizar linhas é, na verdade, vectorizar vértices!** Na sua versão mais simples, no QGIS 3, basta activar as funcionalidades próprias e, depois, clicar sucessivamente para criar a sequência de vértices...



A **criação da primeira linha** é uma tarefa elementar. Num primeiro momento é desenhada a sua configuração gráfica e depois, num segundo momento, são preenchidos os valores da tabela de atributos.

Quando as linhas seguintes são contíguas à primeira são necessárias algumas precauções para evitar **erros geométricos e topológicos**. Para tal, por exemplo, é fundamental activar a funcionalidade **Snapping**.



Quando a **Toolbar** está activa surge a **barra de botões** da funcionalidade...



De notar que a funcionalidade **Snapping** pode também ser configurada de forma duradoura em [Settings >> Options >> Digitizing](#).

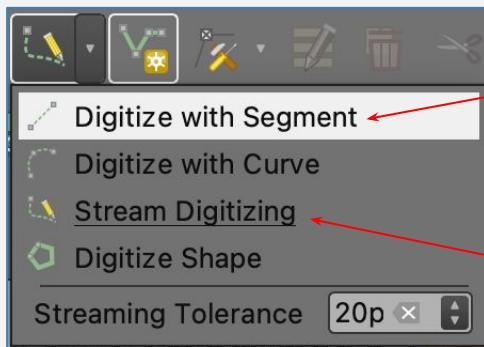
À esquerda e abaixo estão ilustrados os **botões** que dão acesso às diversas funcionalidades relacionadas com a vectorização de linhas (em *shapefiles* de linhas).



# VECTORIZAÇÃO BÁSICA - LINHAS



Quando a funcionalidade criar uma nova linha está activa, activam-se automaticamente os modos de vectorização!



vectorização convencional,  
marcando vértice a vértice...

vectorização contínua,  
desenhando a linha...

As versões mais recentes do QGIS 3 introduziram diversas funcionalidades que facilitam bastante a vectorização de linhas.

Uma dessas novas funcionalidades é a da existência de quatro modos de vectorização:

- **por segmentos de recta** (vectorização convencional, marcando vértice a vértice)
- em curvas
- **contínua**, com vértices automáticos, em distâncias fixas
- em formas

Neste tutorial destacam-se os dois modos mais utilizados e utilizáveis...

# VECTORIZAÇÃO BÁSICA - LINHAS



ferramentas  
simples de  
edição



**VECTORIZAÇÃO POR  
SEGMENTOS DE RETA,  
MARCANDO VÉRTICE A  
VÉRTICE...**

Cada vértice marcado fica  
representado por um X.

Durante a vectorização, para  
apagar o último ponto  
assinalado, ←. Para apagar  
toda a linha em edição Escape.

Para terminar a vectorização,  
botão direito do mouse...

# VECTORIZAÇÃO BÁSICA - LINHAS



ferramentas  
simples de  
edição



## VECTORIZAÇÃO CONTÍNUA, DESENHANDO A LINHA...

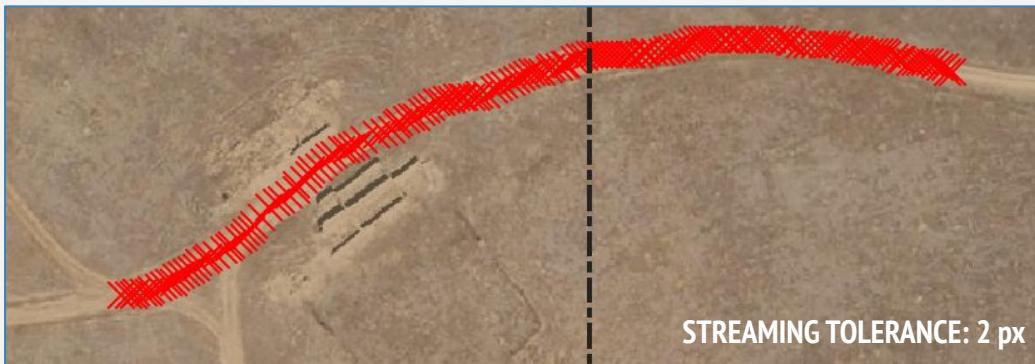
Cada vértice marcado fica  
representado por um **X**.

Durante a vectorização,  
começar por clicar com o botão  
esquerdo do mouse para  
marcar o vértice inicial.

Depois, desenhar  
continuamente a linha  
passando o cursor sobre a  
estrada de terra batida a  
vectorizar.

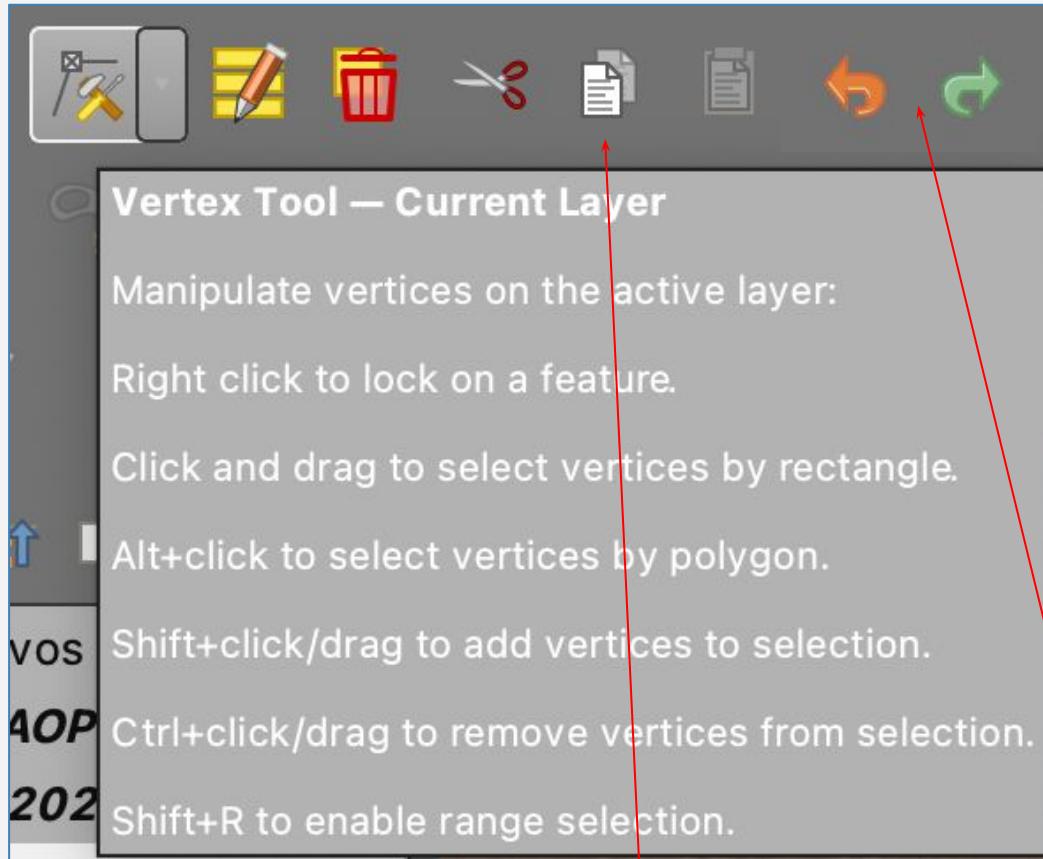
Para terminar a vectorização,  
botão direito do mouse...

Este método é particularmente  
adequado a ser utilizado com  
uma **mesa digitalizadora ou**  
**gráfica e com a caneta própria.**





# VECTORIZAÇÃO BÁSICA - LINHAS



EDITAR / ALTERAR A COMPONENTE GRÁFICA DE UMA LINHA (VÉRTICES)

A edição é feita vértice a vértice ou em conjuntos de vértices, a partir do botão Vertex Tool.

Se se deixar o cursor ficar uns instantes sobre o botão surge um popup com algumas instruções básicas!

BOTÕES CUT &  
COPY & PASTE

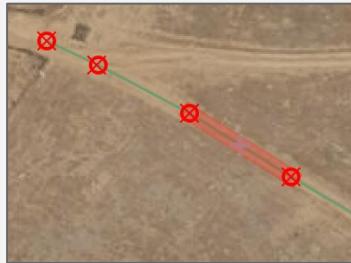
BOTÕES UNDO &  
REDO

# VECTORIZAÇÃO BÁSICA - LINHAS



posicionar o cursor sobre o vértice para o seleccionar e, eventualmente, editar...

posicionar o rato sobre o vértice  
usar **botão esquerdo do rato** para seleccionar  
movimentar o rato para a nova posição  
usar **botão esquerdo do rato** para fixar na nova posição...

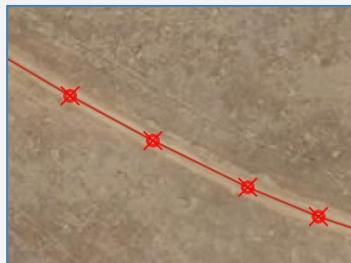


posicionar o cursor sobre um segmento de recta para o seleccionar e, eventualmente, editar...

se o X que está no ponto médio do segmento está cinzento e o próprio segmento tem a barra vermelha translúcida vermelha activada, e movimentando todo o segmento, os dois vértices que o definem, e os segmentos contíguos...

se o rato está sobre o X que está no ponto médio do segmento, o X muda de cor para vermelho

usar **botão esquerdo do rato** para seleccionar  
movimentar o rato para a nova posição  
usar **botão esquerdo do rato** para fixar na nova posição...



posicionar o cursor sobre uma linha completa para a seleccionar e, eventualmente, editar...



bloquear uma linha completa para a seleccionar e, eventualmente, editar, evitando edições em outras linhas...

usar **botão direito do rato** ou  
usar **Ctrl + botão esquerdo do rato**

o mesmo para desbloquear...

# VECTORIZAÇÃO BÁSICA - LINHAS



ferramentas  
avançadas de  
edição

Existe um outro conjunto de funcionalidades que estão associadas às **ferramentas avançadas de edição**. Na vectorização de linhas podem ser particularmente úteis funcionalidades como:

- **Move Feature**
- **Rotate Feature**
- **Scale Feature**
- **Simplify Feature**
  
- **Split Features**
- **Split Parts**
- **Merge Selected Features**
- **Merge Attributes of Selected Features**

Para a utilização de qualquer uma destas funcionalidades há sempre um primeiro passo - a selecção da(s) feature(s) ou ocorrência(s) - e depois de seleccionada pode, então, utilizar-se a funcionalidade...

# VECTORIZAÇÃO BÁSICA - LINHAS



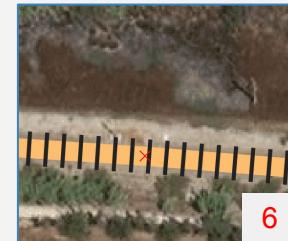
ferramentas  
avançadas de  
edição



## MOVE FEATURE

A “linha férrea” está mal posicionada!

1. *Shapefile* em modo de edição.
2. Selecionar a linha com o botão **Select Features**.
3. Com o botão **Move Feature** clicar sobre a “linha férrea” (*feature* antes selecionada) e fica visível a linha completa (vértices e segmentos de recta assinalados a vermelho).
4. Posicionar o cursor sobre a posição correcta (a *feature* antes selecionada mantém-se na posição original mas a os vértices e a linha movem-se para a nova posição).
5. Clicar depois da *feature* estar bem posicionada e a “linha férrea” movimenta-se para a posição definitiva.
6. Desselecionar a linha com o botão **Deselect Features**.



# VECTORIZAÇÃO BÁSICA - LINHAS



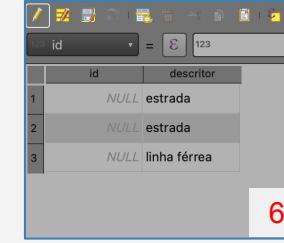
ferramentas  
avançadas de  
edição



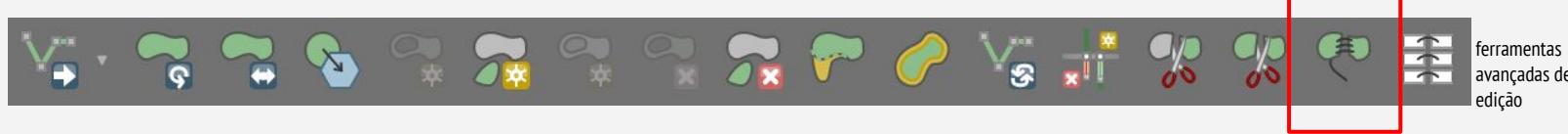
## SPLIT FEATURES

A “estrada” é uma linha única e não permite desenhar uma nova linha e usar as funcionalidades de [escolher o melhor percurso](#)...

1. *Shapefile* em modo de edição.
2. Seleccionar a linha com o botão [Select Features](#).
3. Com o botão [Split Features](#) clicar de forma a iniciar uma linha de corte sobre a “estrada”, ou seja, a feature a dividir em duas.
4. Quando é feito o segundo click para finalizar a linha de corte, a feature divide-se automaticamente em duas...
5. Desselecionar a parte da “estrada” ainda seleccionada com o botão [Deselect Features](#).
6. Na tabela de atributos passam a existir duas features. Antes da operação era apenas uma...



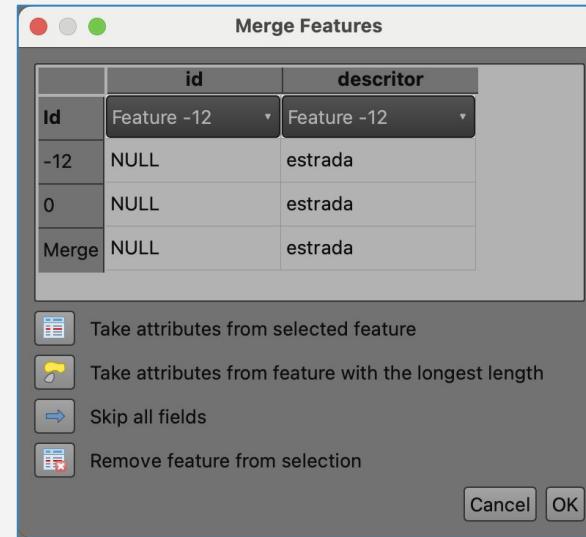
# VECTORIZAÇÃO BÁSICA - LINHAS



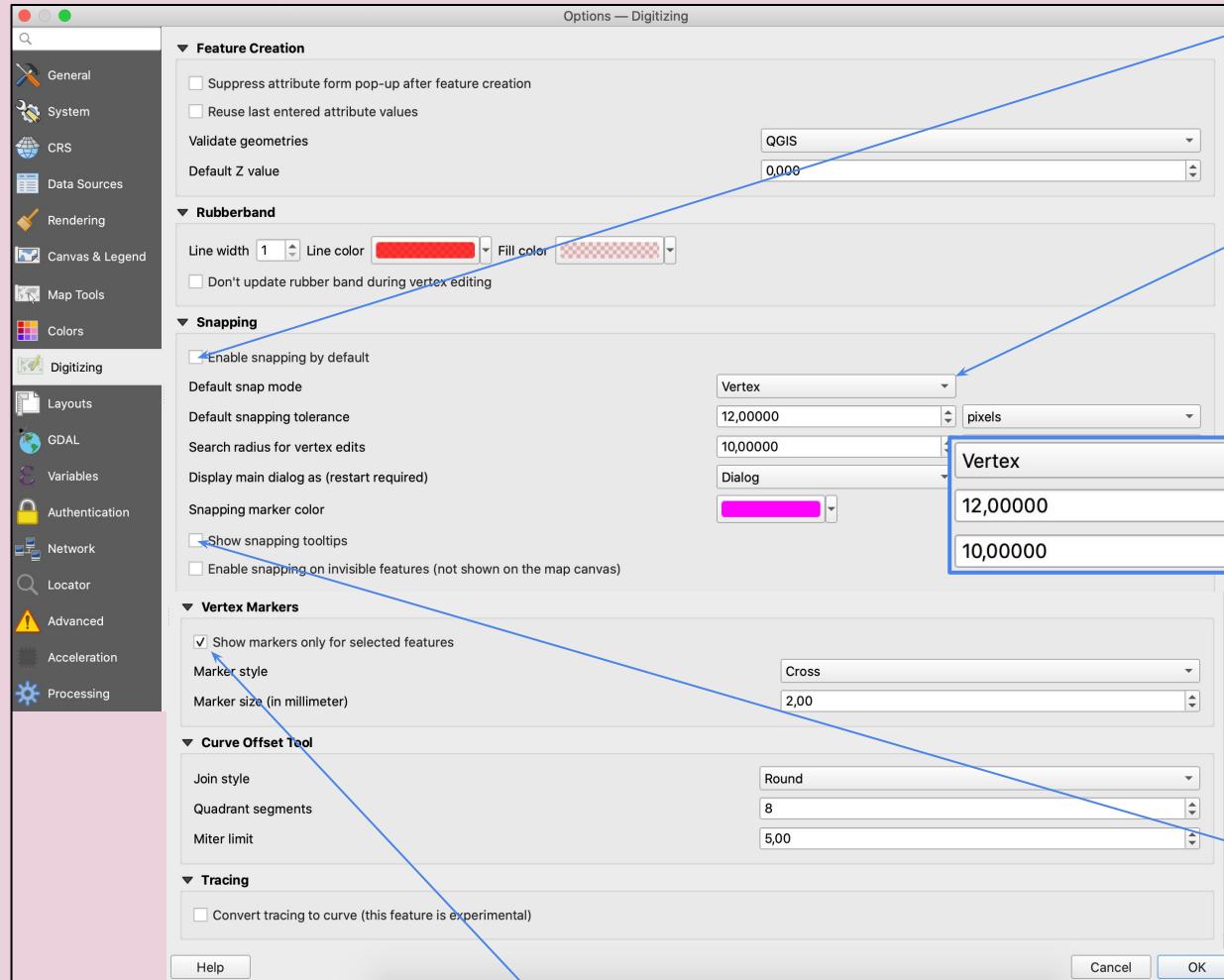
## MERGE SELECTED FEATURES

A funcionalidade **Merge Selected Features** permite fazer o oposto.

1. Com o botão **Select Features by Polygon** seleccionar as duas (ou mais) linhas contíguas que se vão unir e converter numa só.
2. Seguidamente clicar no botão **Merge Selected Features**. A componente gráfica une-se automaticamente e surge uma janela com a componente da tabela de atributos. Nessa janela surgem em cima as features (-12 e 0) seleccionadas no passo inicial e surge em baixo (Merge) a previsão da forma como ficará preenchida a tabelas de atributos relativa à feature unida. A linha superior da tabela (id) permite escolher quais os valores que serão utilizados para tal...



# VECTORIZAÇÃO BÁSICA - SNAPPING



**Vertex Markers:** permite ter assinalados todos os vértices de todas as ocorrências da shapefile, quando esta está em modo de edição, ou apenas das ocorrências que estão seleccionadas e/ou a ser alteradas.

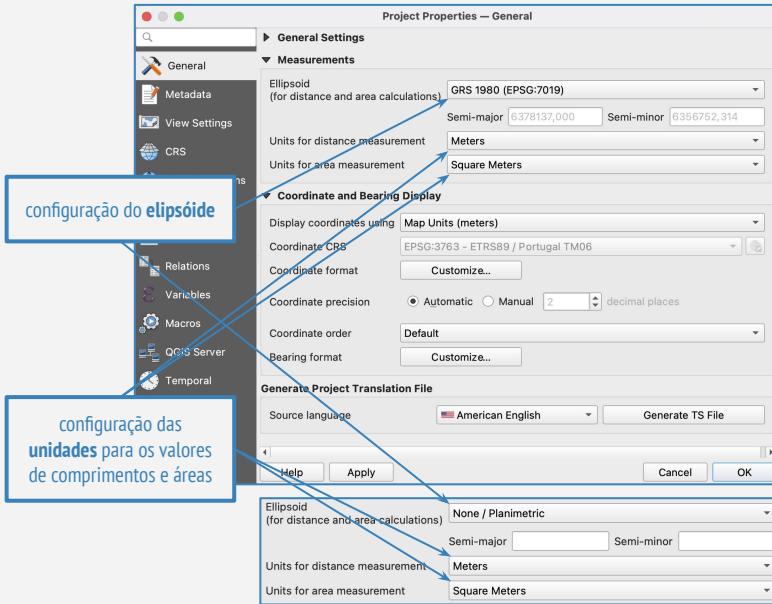


# VECTORIZAÇÃO BÁSICA - LINHAS

A **MEDIDA DE COMPRIMENTOS** é um requisito muito comum em qualquer **shapefile** de linhas. A tarefa é executada com grande facilidade recorrendo ao **Field Calculator**.

Há duas expressões distintas que podem ser utilizadas para calcular o comprimento das linhas:

- **\$length** - calcula o comprimento baseado no **elipsóide** adoptado no projecto e apresenta-o nas **unidades** configuradas também no projecto. Se o elipsóide estiver configurado como **None / Planimetric**, o cálculo será puramente cartesiano, ou seja, sobre o plano.
- **length(@geometry)** - calcula o comprimento planimétrico no **CRS (Coordinate Reference System :: EPSG)** adoptado na **shapefile**. Consequentemente **length(@geometry) = \$length** se o elipsóide estiver configurado como planimétrico.

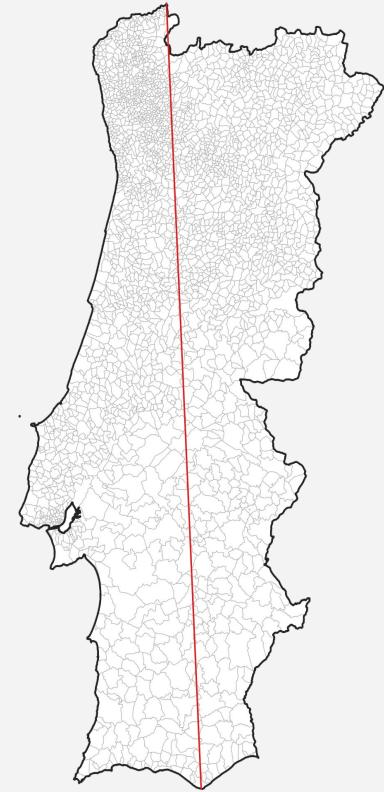


Neste exemplo medi-se o comprimento da linha **vermelha** com as três opções referidas à esquerda.

**length geo** é o resultado obtido com **length(@geometry)**.

**length \$** é o obtido com o elipsóide GRS 1980 e com a expressão **\$length**, enquanto que **plani \$** é o obtido com o elipsóide planimétrico e com a expressão **\$length**.

Todos os comprimentos estão **expressos em metros** e a diferença entre as duas medições é, neste exemplo específico, de 0,874 metros.



length geo	length \$	plani \$
57045,529	577044,655	577045,529

De notar que **length(@geometry)** também permite medir comprimentos em EPSG diferentes, através da expressão  
**length(transform(@geometry, EPSG:3763; 'EPSG:diferente'))**

Por exemplo: **length(transform(@geometry, EPSG:3763; 'EPSG:4326'))**. Não esquecer, no entanto, que as unidades são as do próprio EPSG...

plani \$	4326 geo	20790 geo
577045,529	5,201	577055,771

# VECTORIZAÇÃO BÁSICA - LINHAS

## MEDIÇÃO DE COMPRIMENTOS

O **Field Calculator** permite operações adicionais e simultâneas. Por exemplo, a transformação de metros em quilómetros.

O EPSG:3763 é métrico e os resultados lineares são, por defeito, expressos em metros. A expressão `$length / 1000` permite que o resultado passe a estar expresso em quilómetros!

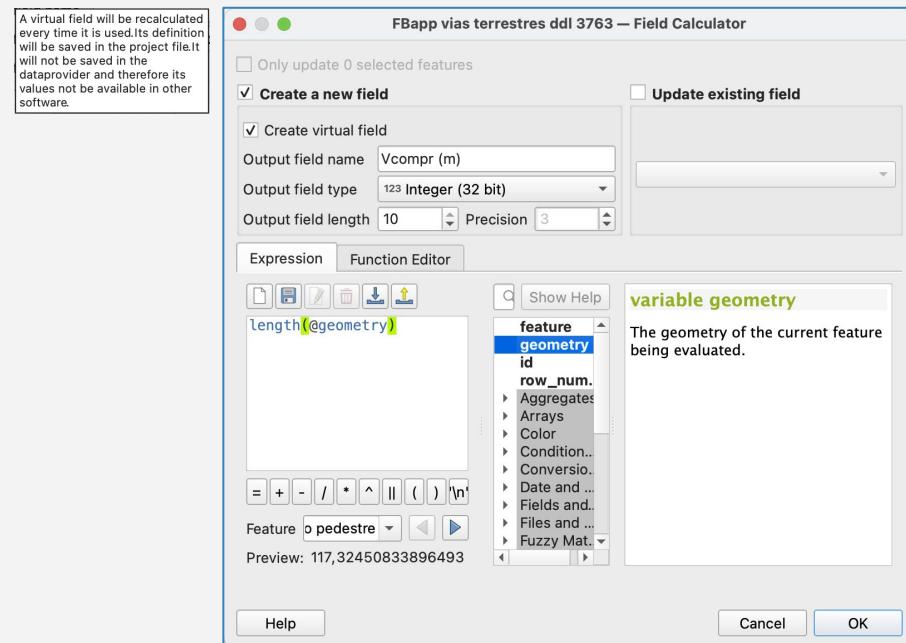
Os atributos relativos às funcionalidades **Create a new field** ou **Update a field** são de preenchimento estático, ou seja, os valores são determinados no momento do cálculo e não se alteram mesmo que as features sejam modificadas.

A criação de um **VIRTUAL FIELD** permite contornar essa limitação. Os valores passam a ser dinâmicos, ou seja, se uma *feature* é modificada, o seu valor é imediatamente ajustado! Para criar uma **Virtual field** basta activar a funcionalidade no **Field calculator**.

Há, no entanto, algumas limitações relacionadas com os **Virtual Fields**.

*Virtual fields are not permanent in the layer attributes, meaning that they're only saved and available in the project file they've been created.*

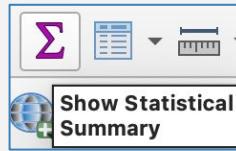
Consequentemente, se a *shapefile* for enviada para outro utilizador ou instalada num outro computador, o **virtual field** e os seus valores serão perdidos.



# VECTORIZAÇÃO BÁSICA - LINHAS

## SHOW STATISTICAL SUMMARY

A funcionalidade **Show Statistical Summary** permite conhecer um conjunto de indicadores sobre uma determinada *shapefile* de linhas...



Neste exemplo escolheu-se uma *shapefile* (**FBapp vias terrestres ddl 3763**) e particularmente um atributo da mesma (**compr (m)**). Na *shapefile* selecionaram-se algumas das *features* (ocorrências).

A funcionalidade **Show Statistical Summary** permite conhecer indicadores estatísticos da totalidade da *shapefile* ou apenas das ocorrências na mesma seleccionadas...

Statistic	Value
Count	202
Sum	68214,7
Mean	337,697
Median	150,45
St dev (pop)	574,579
St dev (sample)	576,007
Minimum	10,4
Maximum	5026,8
Range	5016,4
Minority	10,4
Majority	84,1
Variety	179
Q1	61,3
Q3	423,9
IQR	362,6
Missing (null) values	0

Selected features only

Statistic	Value
Count	86
Sum	28059,3
Mean	326,271
Median	216,45
St dev (pop)	307,606
St dev (sample)	309,41
Minimum	11
Maximum	1512,5
Range	1501,5
Minority	11
Majority	22,4
Variety	72
Q1	84,1
Q3	481,8
IQR	397,7
Missing (null) values	0

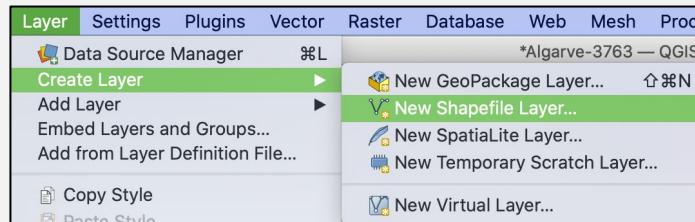
Selected features only



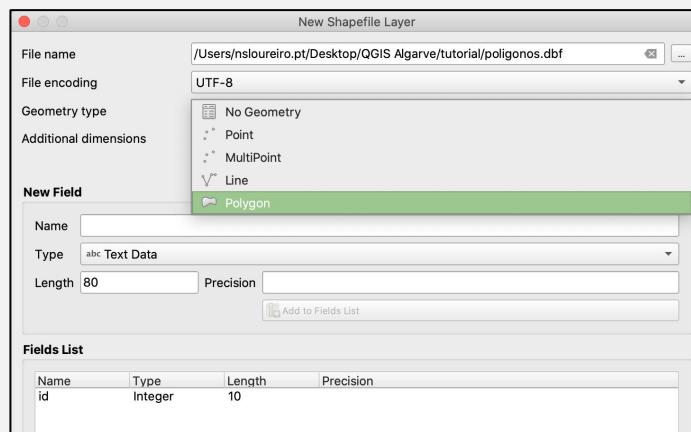
# VECTORIZAÇÃO BÁSICA - POLÍGONOS

Um primeiro passo, prévio à vectorização de polígonos, é a criação da **shapefile** onde essa **informação** vai ser registada e armazenada.

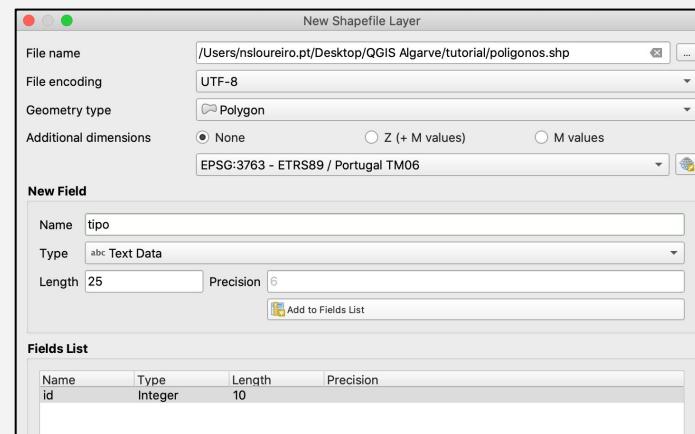
A criação da **shapefile** é feita em



É necessário criar o conjunto de ficheiros da **shapefile** e depois configurar a própria **shapefile**. Os primeiros elementos da configuração são o **File encoding** (UTF-8 ou ISO 8859-1 para a língua portuguesa) e o **Geometry type** (No Geometry - Point - Multipoint - Line - **Polygon**).

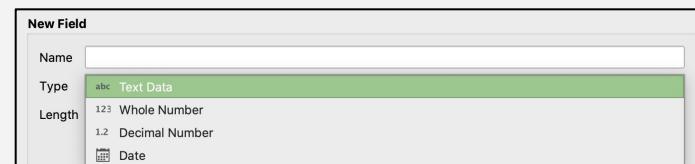


Em seguida é necessário configurar se existem ou não **Additional dimensions** (Z e/ou M values) e também o **EPSG** da **shapefile** (3763 para Portugal continental).



Por fim é necessário criar os campos (**Fields**) que vão existir na tabela de atributos (**Attribute Table**). Por defeito está criado o campo **id**, mas podem ser criados todos os outros necessários.

Os campos podem ser de quatro tipos (Text Data - Whole Number - Decimal Number - Date) e, em função do tipo, há especificidades de comprimento máximo de caracteres atribuídos e de grau de precisão.



Para que o novo campo seja adicionado à tabela de atributos é necessário clicar em **Add to Fields List**.

De notar que enquanto que o **tipo de geometria** é uma característica que é escolhida na configuração inicial, e não pode ser alterada posteriormente, os **campos** podem ser acrescentados, alterados ou eliminados em qualquer momento, de acordo com as necessidades...



# VECTORIZAÇÃO BÁSICA - POLÍGONOS

Depois de concluir a criação da *shapefile*, ela é carregada automaticamente no **Layer Panels**.

Para começar a vectorizar basta colocar a *layer* (*shapefile*) em modo de **edição**:



Existem dois botões que ficam activos:



O botão **Add Polygon Feature** é utilizado para criar um novo polígono.

O botão **Vertex Tool** é utilizado para efectuar alterações em polígonos já existentes. É um botão que tem dois modos: **Vertex Tool (All Layers)** e **Vertex Tool (Current Layer)**. Preferencialmente, deve ser escolhido o modo **Current Layer** para evitar edições indesejadas...

A **criação do primeiro polígono** é uma tarefa elementar. Num primeiro momento é desenhada a sua configuração gráfica (geométrica) e depois, num segundo momento, são preenchidos os valores da tabela de atributos.

Quando os polígonos seguintes são contíguos ao primeiro são necessárias algumas precauções para evitar **erros geométricos e topológicos**. Para tal, por exemplo, é fundamental activar a funcionalidade **Snapping**.





# TOPOLOGIA

**Topologia** é a existência de relações espaciais entre **entidades vectoriais** (*features: points, lines and polygons*) que ocorrem numa *shapefile* e que são contíguas.

Por exemplo, dois segmentos de uma estrada que têm de estar em perfeito contacto um com outro, para que o ‘trânsito’ possa circular. Ou duas parcelas contíguas de terreno que têm de estar exactamente em contacto uma com a outra, sem sobreposições ou lacunas...

A **análise topológica**, num SIG, é fundamental para estudar a existência de redes (*network analysis*) e também para identificar a eventual existência de erros de vectorização e, em seguida, para os corrigir.

## ERROS TOPOLÓGICOS

### POLÍGONOS

- **sobreposição** (*overlapping*) entre dois polígonos contíguos
- **lacuna** (*gapping*) entre dois polígonos contíguos

### LINHAS

- **sobreposição** (*overshoot*) entre duas linhas que se sobrepõem e criam um ou dois nós órfãos ou pendentes (*dangling nodes*)
- **lacuna** (*undershoot*) entre duas linhas que não se chegam a tocar e criam dois nós órfãos ou pendentes (*dangling nodes*)

O QGIS 3 tem **ferramentas próprias** para evitar a criação de erros topológicos e também para os eliminar quando os mesmos existam. O **Snapping** é uma ferramenta fundamental para **evitar a criação de erros topológicos!**



# GEOMETRIA

Enquanto que a topologia está focada nas relações espaciais entre duas ou mais entidades vectoriais contíguas, a **geometria** está focada nas características de cada uma dessas entidades, de forma independente, sem se preocupar com as relações entre essa entidade vectorial e as outras entidades que possam existir na *shapefile*.

A **análise geométrica**, num SIG, é fundamental para assegurar o rigor e o sucesso de análises espaciais numa mesma *shapefile*, ou entre *shapefiles* distintas, para além de permitir a identificação de eventuais erros de vectorização e, em seguida, para os corrigir.

## ERROS GEOMÉTRICOS

### POLÍGONOS

- **auto-sobreposição** (*self-intersection*)

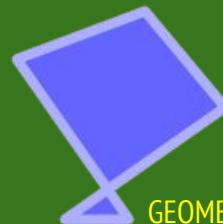
### LINHAS

- **duplicação de vértices**

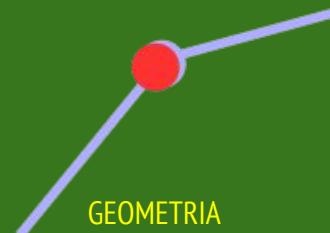
## TOPOLOGIA E GEOMETRIA



## Self intersections   Duplicate nodes



## GEOMETRIA

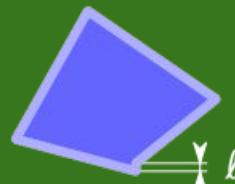


GEOMETRIA

## Holes

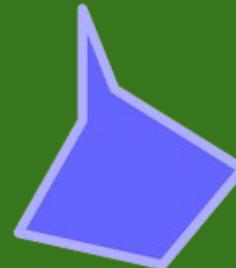


## Segment length



三

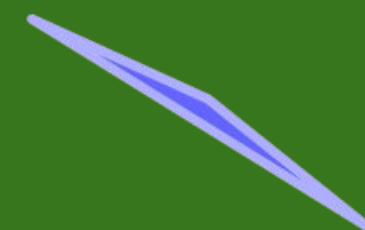
## Minimum angle



## Minimum area



## Sliver polygons



## Duplicates



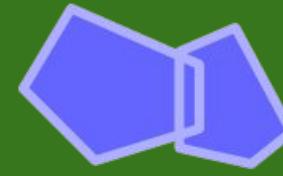
TOPOLOGIA

## Feature within feature



# TOPOLOGIA

## Overlaps



TOPOLOGIA

Gaps



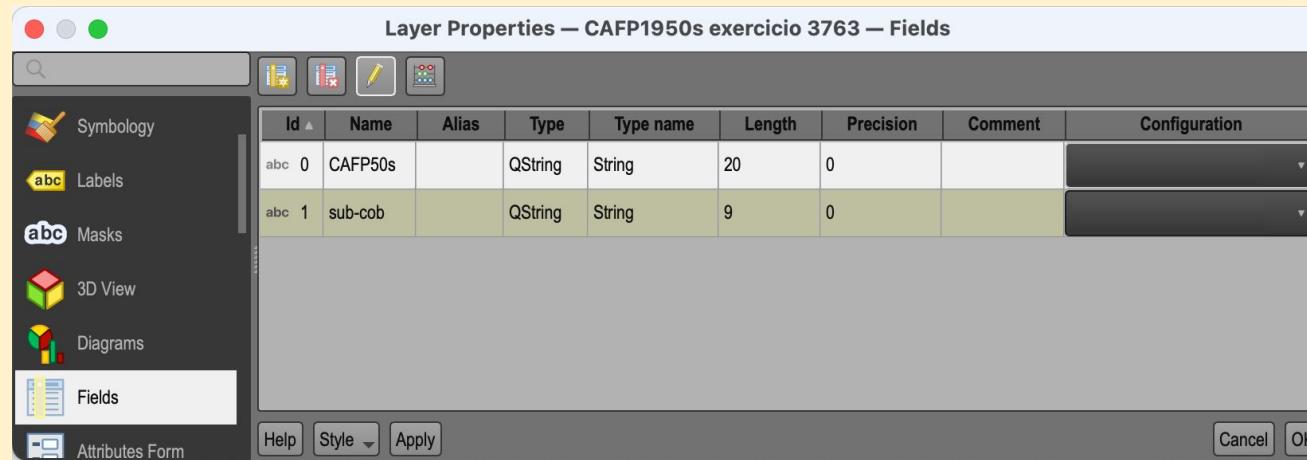
TOPOLOGIA

# VECTORIZAÇÃO BÁSICA - POLÍGONOS

O exemplo que se apresenta consiste da vectorização de um excerto da Carta Agrícola e Florestal de Portugal na década de 1950.

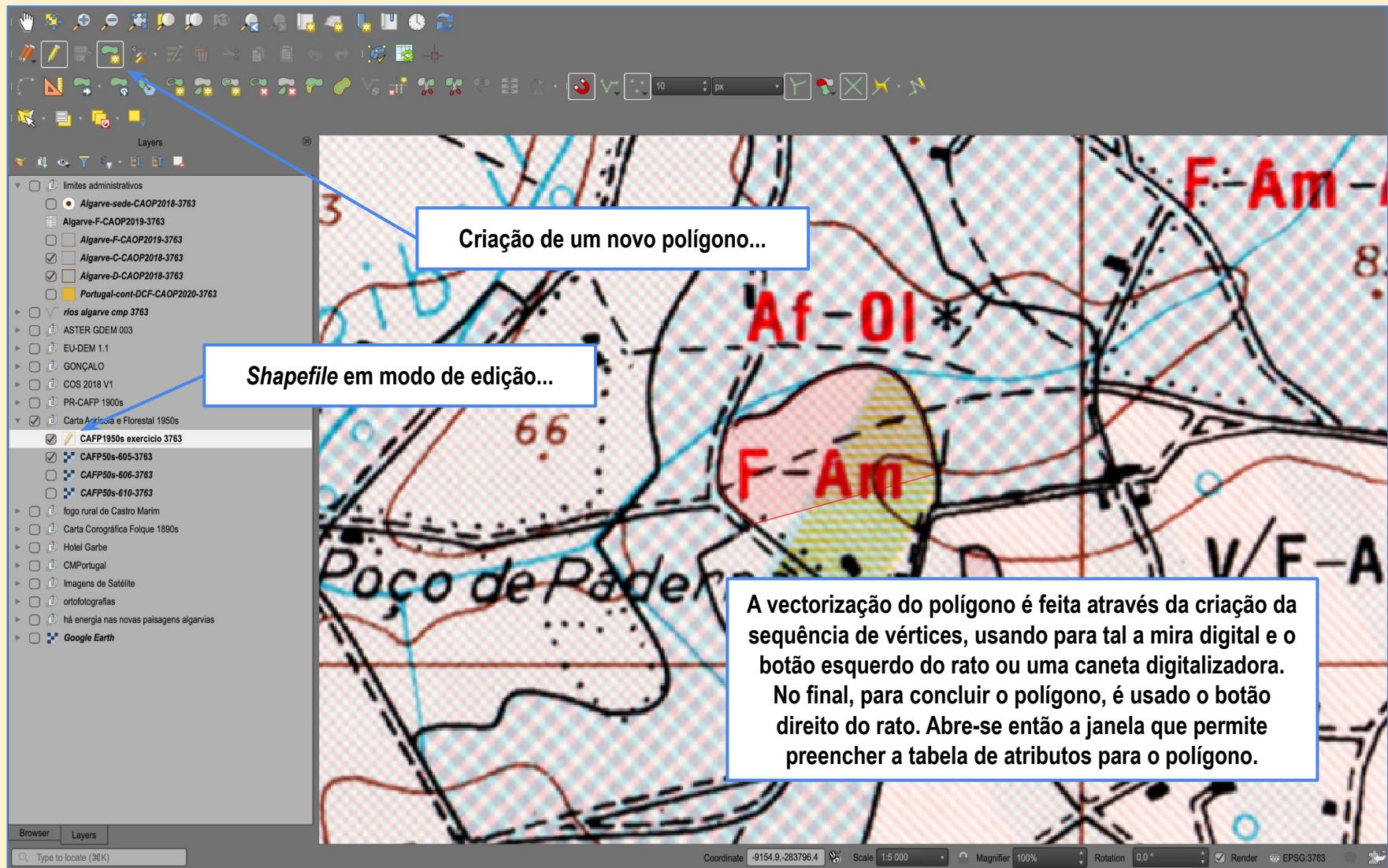
A base cartográfica é a Folha 605 da CAFP 1950s, que foi previamente georreferenciada em EPSG 3763.

Foi criada uma *shapefile* para polígonos, com dois campos na tabela de atributos...



... e foram estabelecidos os correspondentes critérios de preenchimento.

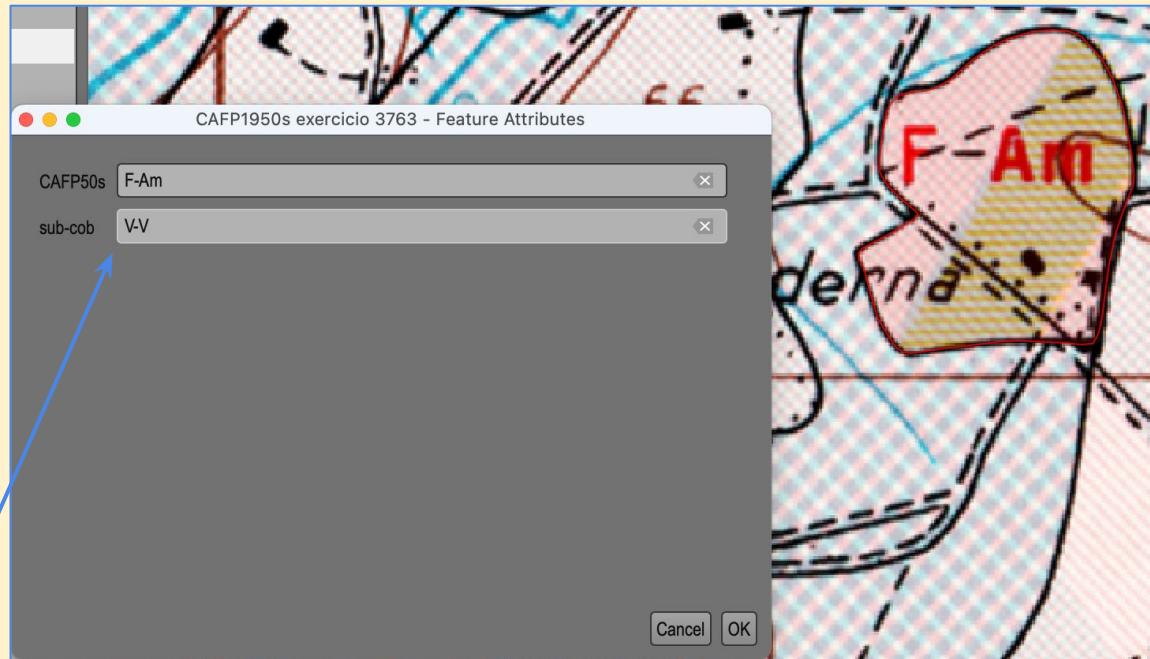
# VECTORIZAÇÃO BÁSICA - POLÍGONOS



# VECTORIZAÇÃO BÁSICA - POLÍGONOS



Durante a criação é possível  
eliminar o último vértice criado  
através da tecla ← | Del

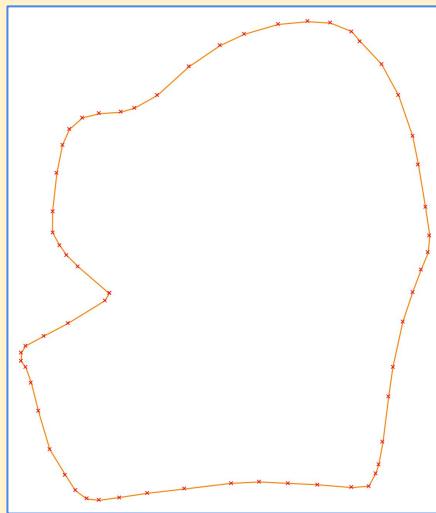


Janela para preenchimento da  
tabelas de atributos.  
Se necessário, também pode ser  
posteriormente corrigida...

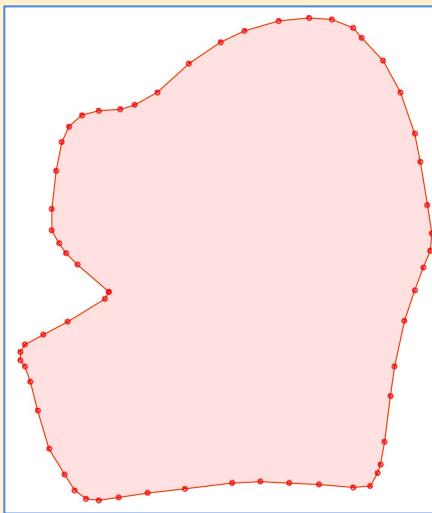
# VECTORIZAÇÃO BÁSICA - POLÍGONOS



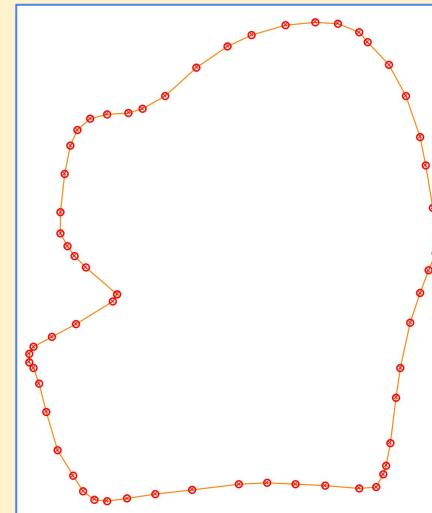
Vertex  
Tool



**Shapefile em modo de Edição:**  
os vértices estão assinalados com X...



**Shapefile em modo Vertex Tool:**  
os vértices estão assinalados com o e o polígono destacado.



**Shapefile em modo Vertex Tool:**  
os vértices estão assinalados com O no polígono em edição.

Enquanto que o **Add Feature** permite criar novos pontos, linhas ou polígonos, o **Vertex Tool** permite fazer alterações em pontos, linhas ou polígonos anteriormente criados.

Quando está a ser utilizado o **Vertex Tool** basta colocar o cursor em cima de um ponto, linha ou polígono para que o mesmo fique destacado e com os vértices assinalados através de círculos vermelhos.

Para que os vértices fiquem permanentemente activados e seja simples e segura a tarefa de os editar (evitando, por exemplo, modificações em polígonos errados) é necessário clicar no botão direito do rato. Os círculos ficam então com um diâmetro maior...

# VECTORIZAÇÃO BÁSICA - POLÍGONOS

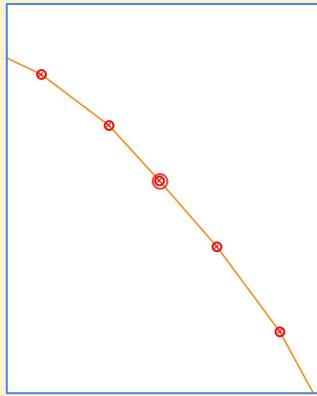


Vertex  
Tool

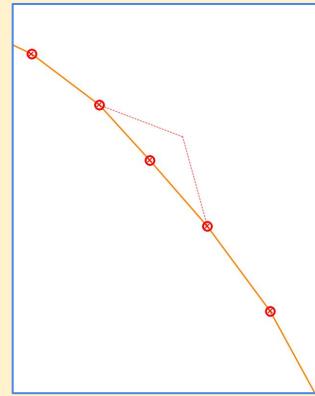


Com o polígono permanentemente activado tornam-se possíveis **três operações fundamentais, que se executam vértice a vértice:**

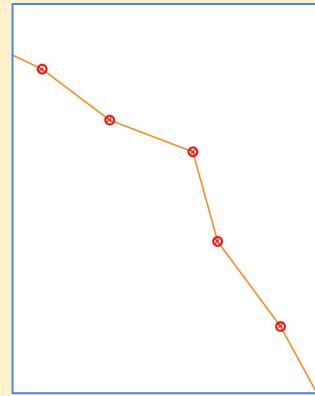
- **ajustar a posição do vértice**, o que se consegue colocando o cursor sobre o vértice, clicando sobre ele e depois movimentando-o
- **eliminar o vértice**, o que se consegue colocando o cursor sobre o vértice, clicando sobre ele e depois clicando na tecla ← | Del
- **criar um novo vértice**, o que consegue sobrepondo o cursor ao segmento onde se pretende criar o novo vértice e depois sobre a cruz cinzenta; em seguida, quando a cruz fica vermelha, ao clicar sobre ela o novo vértice fica criado, podendo ser de imediato reposicionado...



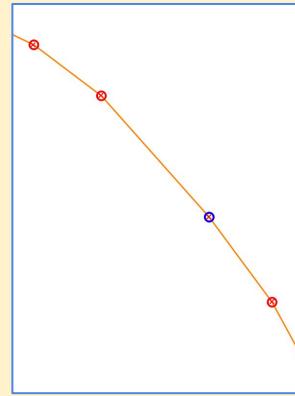
selecção do vértice  
a editar...



movimentar o vértice...



vértice reposicionado...

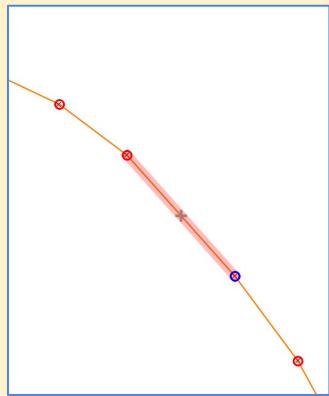


selecionar e apagar  
o vértice...

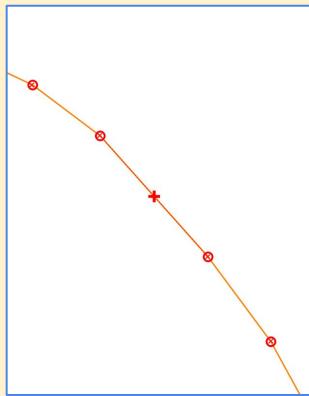
# VECTORIZAÇÃO BÁSICA - POLÍGONOS



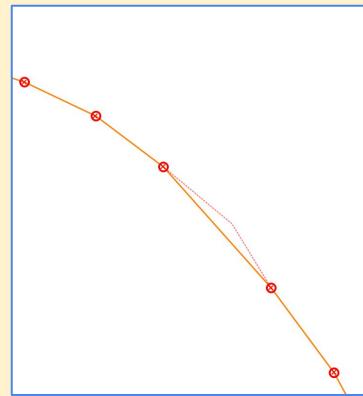
Vertex  
Tool



criação de novo vértice  
a partir do ponto  
central...



repositionamento do  
novo vértice...



repositionamento do novo  
vértice...

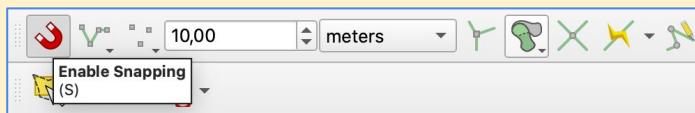
# SNAPPING - ADVANCED CONFIG...



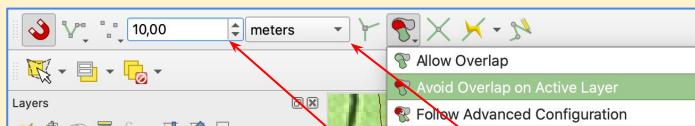
Snapping  
Tool



Para vectorizar, vértice a vértice, um segundo polígono é necessário utilizar o **Snapping**, para evitar a criação de erros topológicos.



Activar o Snapping e depois...



activar a funcionalidade **Avoid Overlap on Active Layer**.

Com a funcionalidade activada é possível desenhar um novo polígono sobrepondo-o ao(s) já existente(s). Quando terminar a criação do elemento gráfico e estiverem preenchidos os campos na tabela de atributos, o novo polígono vai ‘ajustar-se’ sem qualquer erro topológico (sobreposição ou lacuna) aos limites do(s) polígono(s) contíguo(s).

unidades do  
Snapping

amplitude do  
Snapping



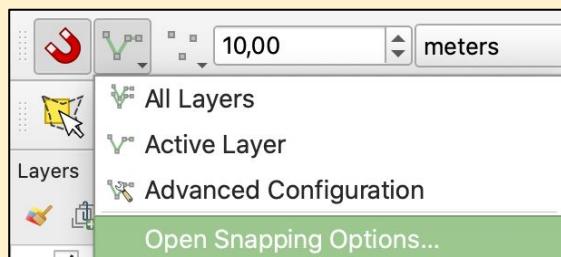
# SNAPPING - ADVANCED CONFIG...



Snapping  
Tool



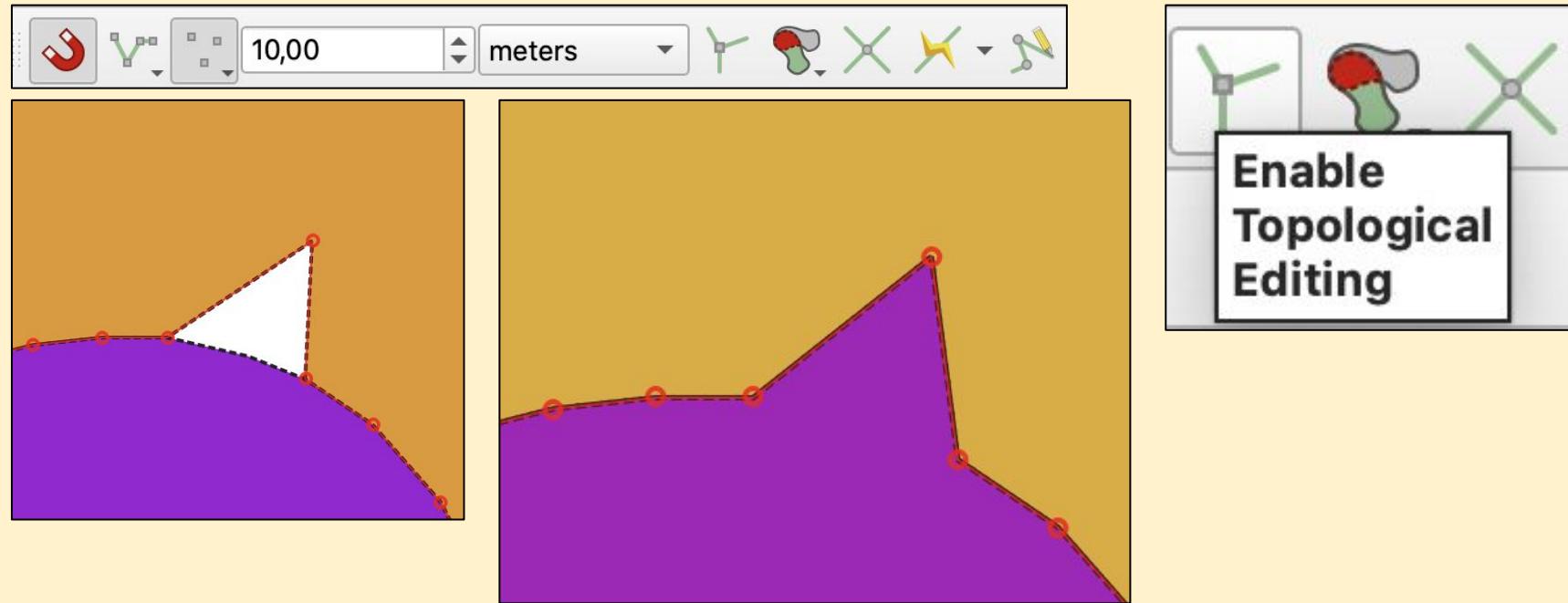
Para desenhar (vectorizar) os polígonos seguintes continua a poder-se utilizar a mesma regra. Os limites contíguos desenham-se com uma sobreposição temporária e elementar, que é automaticamente corrigida quando se faz o **OK** na janela relativa ao preenchimento da tabela de atributos.



As funcionalidades detalhadas do Snapping têm, com frequência, de ser ajustadas a situações concretas e particulares da edição. Por vezes é **muito útil** abrir a janela flutuante (**Open Snapping Options...**) que permite fazer esses ajustamentos...



# VECTORIZAÇÃO BÁSICA - POLÍGONOS



Quando está a ser utilizado o **Vertex Tool** para ajustar vértices e quando existem polígonos contíguos, a edição pode ser feita em dois modos distintos:

- com o **Topological Editing inactivo**, o que permite alterar a posição de um vértice de apenas um polígono
- com o **Topological Editing activo**, o que permite alterar a posição do vértice ‘comum’ aos dois polígonos, assegurando que não surgem erros topológicos

# verificação geométrica e topológica



# GEOMETRIA

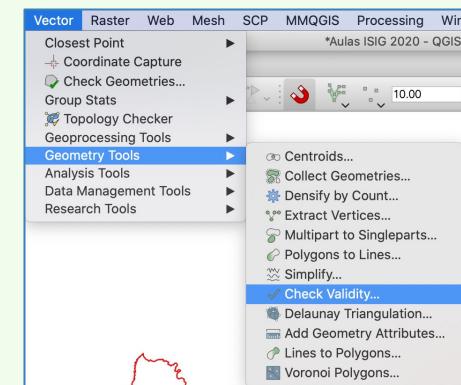
O QGIS 3 tem **ferramentas próprias** para verificar a inexistência ou existência de erros geométricos e topológicos e também para os identificar e até, eventualmente, eliminar quando existam...

## FERRAMENTAS PARA IDENTIFICAR A EXISTÊNCIA DE ERROS GEOMÉTRICOS

### CHECK VALIDITY...

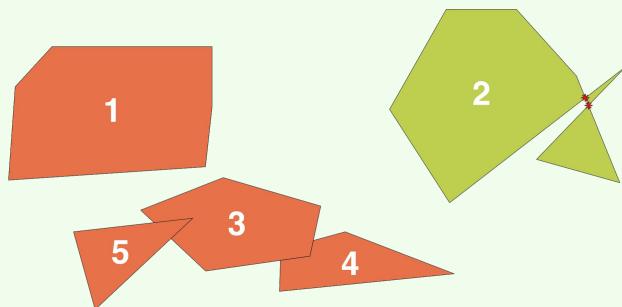
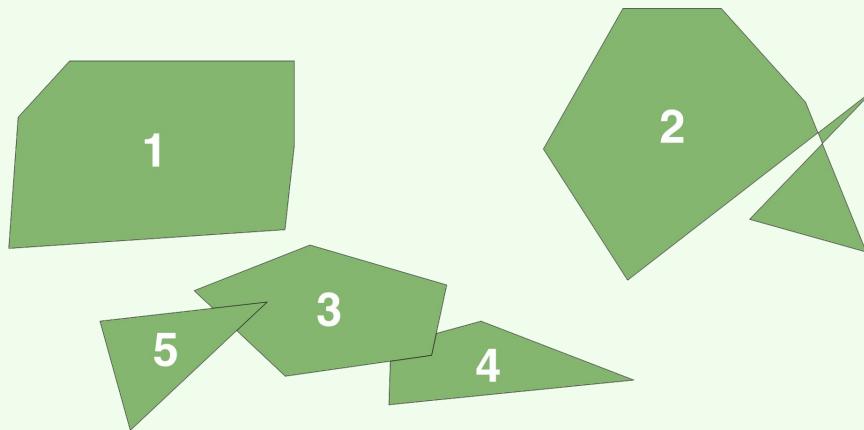
A funcionalidade **CHECK VALIDITY...** é uma opção bastante útil para identificar, localizar e, eventualmente, facilitar a eliminação de erros geométricos.

- **Vector >>> Geometry Tools >>> Check Validity...**
  - This algorithm performs a **validity check on the geometries** of a vector layer.
  - The geometries are classified in three groups (**valid**, invalid and error), and a vector layer is generated with the features in each of these categories.
  - By default the algorithm uses the strict OGC definition of polygon validity, where a polygon is marked as invalid if a self-intersecting ring causes an interior hole. If the "Ignore ring self intersections" option is checked, then this rule will be ignored and a more lenient validity check will be performed.
  - The **GEOS method** is faster and performs better on larger geometries, but is limited to only returning the first error encountered in a geometry. **The QGIS method will be slower but reports all errors encountered in the geometry**, not just the first.
- **Method >>> QGIS**





# Check Validity...



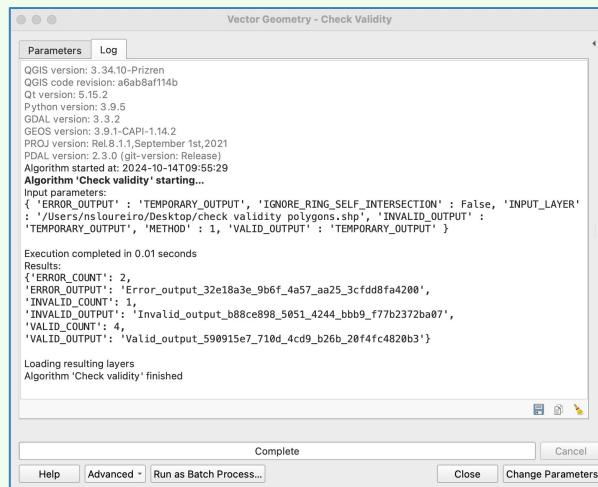
- Error output**
- Valid output**
- Invalid output**

O Check Validity... cria três novas layers:

- uma de pontos, a identificar e localizar as posições onde há erros
- uma de polígonos, a assinalar as features correctas
- uma outra de polígonos, a assinalar as features inválidas

Neste exemplo estão representados cinco polígonos:

1. sem quaisquer erros
2. com um erro geométrico (self intersection)
3. sem quaisquer erros
4. sem quaisquer erros (vectorizado com o auxílio da funcionalidade Avoid Overlap)
5. com um erro topológico (sobreposição com o polígono 3)



O Check Validity... incide apenas sobre a geometria. Identifica e localiza erros geométricos mas ignora os erros topológicos.

Neste exemplo são identificados e assinalados:

- dois pontos com erros
- um polígono inválido
- quatro polígonos válidos \*

\* ou seja, o erro topológico não é nem identificado nem assinalado...

O Log do Check Validity... indica o número de erros, o número de features correctas e o de features inválidas.

# Check Validity...

● ● ● Error output — Features Total: 2, Filtered: 2, Select...

message

1	segments 4 and 6 of line 0 intersect at 87897.5,132001
2	segments 4 and 7 of line 0 intersect at 87885.2,132032

Show All Features

● ● ● Invalid output — Features Total: 1, Filtered: 1, Selected: 0

errors

1	2 segments 4 and 6 of line 0 intersect at 87897.5,132001
	2 segments 4 and 7 of line 0 intersect at 87885.2,132032

Show All Features

● ● ● Valid output — Features Total: 4, Filtered: 4, Select...

ID

1	1
2	3
3	4
4	5

Show All Features

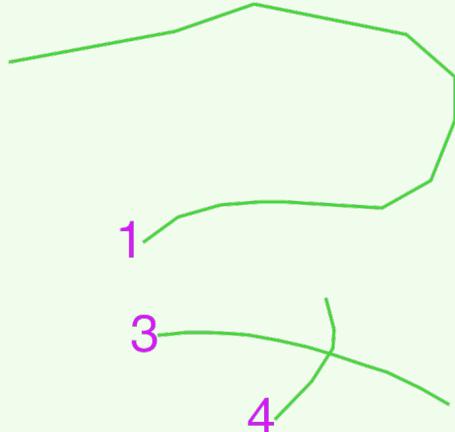
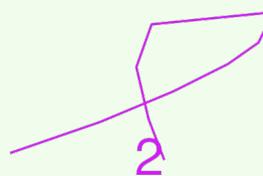
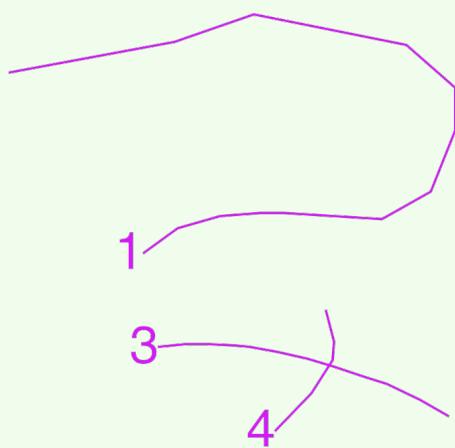
As tabelas de atributos das três novas layers contêm informação adicional sobre os erros.

A layer **Error output** indica as coordenadas dos pontos onde estão os erros.

A layer **Invalid output** indica o ID do(s) polígono(s) onde estão os erros.

A layer **Valid output** indica o ID do(s) polígono(s) que estão isentos de erros geométricos.

# Check Validity...



✳ Error output  
— Valid output  
— Invalid output



Neste exemplo estão representadas quatro linhas, por ordem de criação:

1. sem quaisquer erros
2. com um erro geométrico (self intersection)
3. sem quaisquer erros (ver 4.)
4. com um erro topológico (sobreposição com a linha 3)

Mais uma vez, o **Check Validity...** incide apenas sobre a **geometria**. Identifica e localiza **erros geométricos** mas ignora os **erros topológicos**.

Neste exemplo são identificados e assinalados:

- um ponto com erros
- uma linha inválida
- três linhas válidas \*

\* ou seja, o erro topológico entre as linhas 3 e 4 não é nem identificado nem assinalado...



# TOPOLOGIA

## FERRAMENTAS PARA IDENTIFICAR ERROS TOPOLÓGICOS

### TOPOLOGY CHECKER PLUGIN

O plugin **TOPOLOGY CHECKER** é de grande utilidade para identificar e localizar erros topológicos e também geométricos. Como qualquer outro plugin, para ser utilizado tem de estar activado. O **botão de acesso** ao plugin surge no menu principal **Vector** e também na barra de menus.

É eficaz para lidar com os seguintes erros:

- auto-intersecções (*self intersections*)
- duplicação de nós (*duplicate nodes*)
- buracos (*holes*) no interior de polígonos
- polígonos duplicados (*duplicate features*)
- polígonos dentro de polígonos (*feature within feature*)
- sobreposições (*overlaps*) entre polígonos
- lacunas (*gaps*) entre polígonos

E igualmente:

- segmentos mínimos (*segment length*), ou seja, segmentos com comprimento inferior a um determinado valor
- ângulos mínimos (*minimum angle*), ou seja, dois segmentos a formar entre si um ângulo inferior a um determinado valor
- áreas mínimas (*minimum area*), ou seja, polígonos com uma área inferior a um determinado valor
- polígonos inconsistentes (*sliver polygons*), ou seja polígonos que têm uma relação entre a área (muito reduzida) e o perímetro (muito elevado) inferior a um determinado valor

saber mais [AQUI](#)



# Topology Checker...

O plugin TOPOLOGY CHECKER pode ser utilizado com layers de **pontos**, de **linhas** e de **polígonos**...

On point layers the following rules are available:

- **Must be covered by** : Here you can choose a vector layer from your project. Points that aren't covered by the given vector layer occur in the 'Error' field.
- **Must be covered by endpoints of** : Here you can choose a line layer from your project.
- **Must be inside** : Here you can choose a polygon layer from your project. The points must be inside a polygon. Otherwise, QGIS writes an 'Error' for the point.
- **Must not have duplicates** : Whenever a point is represented twice or more, it will occur in the 'Error' field.
- **Must not have invalid geometries** : Checks whether the geometries are valid.
- **Must not have multi-part-geometries** : All multi-part points are written into the 'Error' field.

On line layers, the following rules are available:

- **End points must be covered by** : Here you can select a point layer from your project.
- **Must not have dangles** : This will show the overshoots in the line layer.
- **Must not have duplicates** : Whenever a line feature is represented twice or more, it will occur in the 'Error' field.
- **Must not have invalid geometries** : Checks whether the geometries are valid.
- **Must not have multi-part geometries** : Sometimes, a geometry is actually a collection of simple (single-part) geometries. Such a geometry is called multi-part geometry. If it contains just one type of simple geometry, we call it multi-point, multi-linestring or multi-polygon. All multi-part lines are written into the 'Error' field.
- **Must not have pseudos** : A line geometry's endpoint should be connected to the endpoints of two other geometries. If the endpoint is connected to only one other geometry's endpoint, the endpoint is called a pseudo node.

On polygon layers, the following rules are available:

- **Must contain** : Polygon layer must contain at least one point geometry from the second layer.
- **Must not have duplicates** : Polygons from the same layer must not have identical geometries. Whenever a polygon feature is represented twice or more it will occur in the 'Error' field.
- **Must not have gaps** : Adjacent polygons should not form gaps between them. Administrative boundaries could be mentioned as an example (US state polygons do not have any gaps between them...).
- **Must not have invalid geometries** : Checks whether the geometries are valid. Some of the rules that define a valid geometry are:
  - Polygon rings must close.
  - Rings that define holes should be inside rings that define exterior boundaries.
  - Rings may not self-intersect (they may neither touch nor cross one another).
  - Rings may not touch other rings, except at a point.
- **Must not have multi-part geometries** : Sometimes, a geometry is actually a collection of simple (single-part) geometries. Such a geometry is called multi-part geometry. If it contains just one type of simple geometry, we call it multi-point, multi-linestring or multi-polygon. For example, a country consisting of multiple islands can be represented as a multi-polygon.
- **Must not overlap** : Adjacent polygons should not share common area.
- **Must not overlap with** : Adjacent polygons from one layer should not share common area with polygons from another layer.

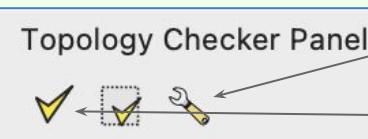
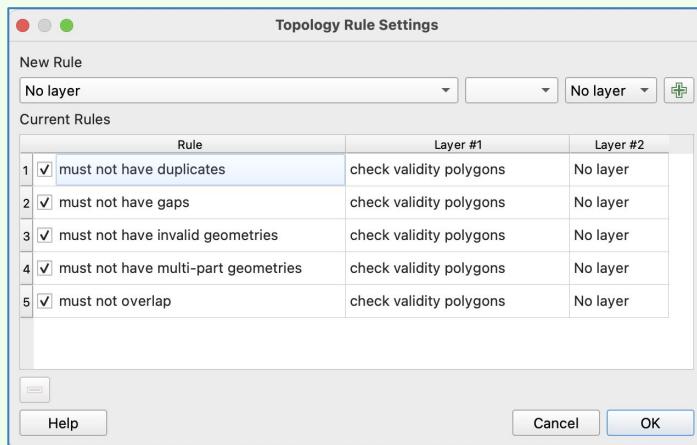
O plugin TOPOLOGY CHECKER pode ser utilizado para verificar erros numa só layer ou entre duas layers...

As **setas a azul** assinalam verificações nas features de uma única layer, seja ela de **pontos**, de **linhas** ou de **polígonos**.

As **setas a laranja** assinalam verificações entre duas layers distintas...



# Topology Checker...

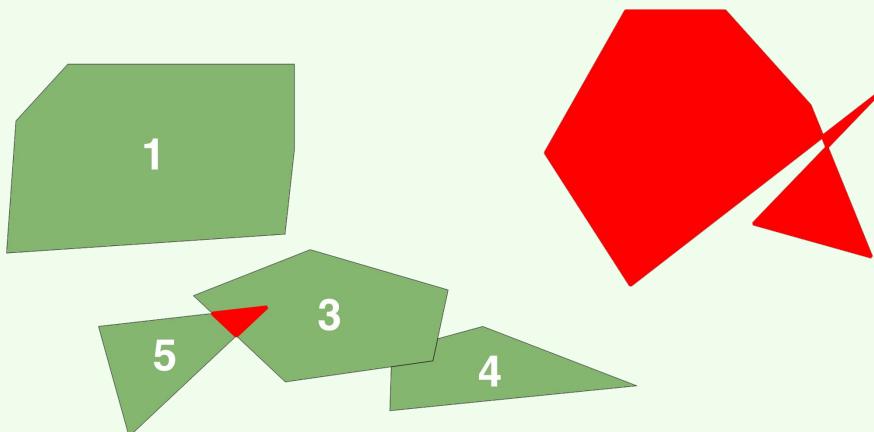


configuração da verificação

efectuar a verificação completa

O plugin tem uma **ferramenta de configuração** onde se estabelecem detalhadamente as verificações pretendidas (**Topology Rule Settings**).

Depois tem um botão para efectuar a **verificação (Validate All)**..

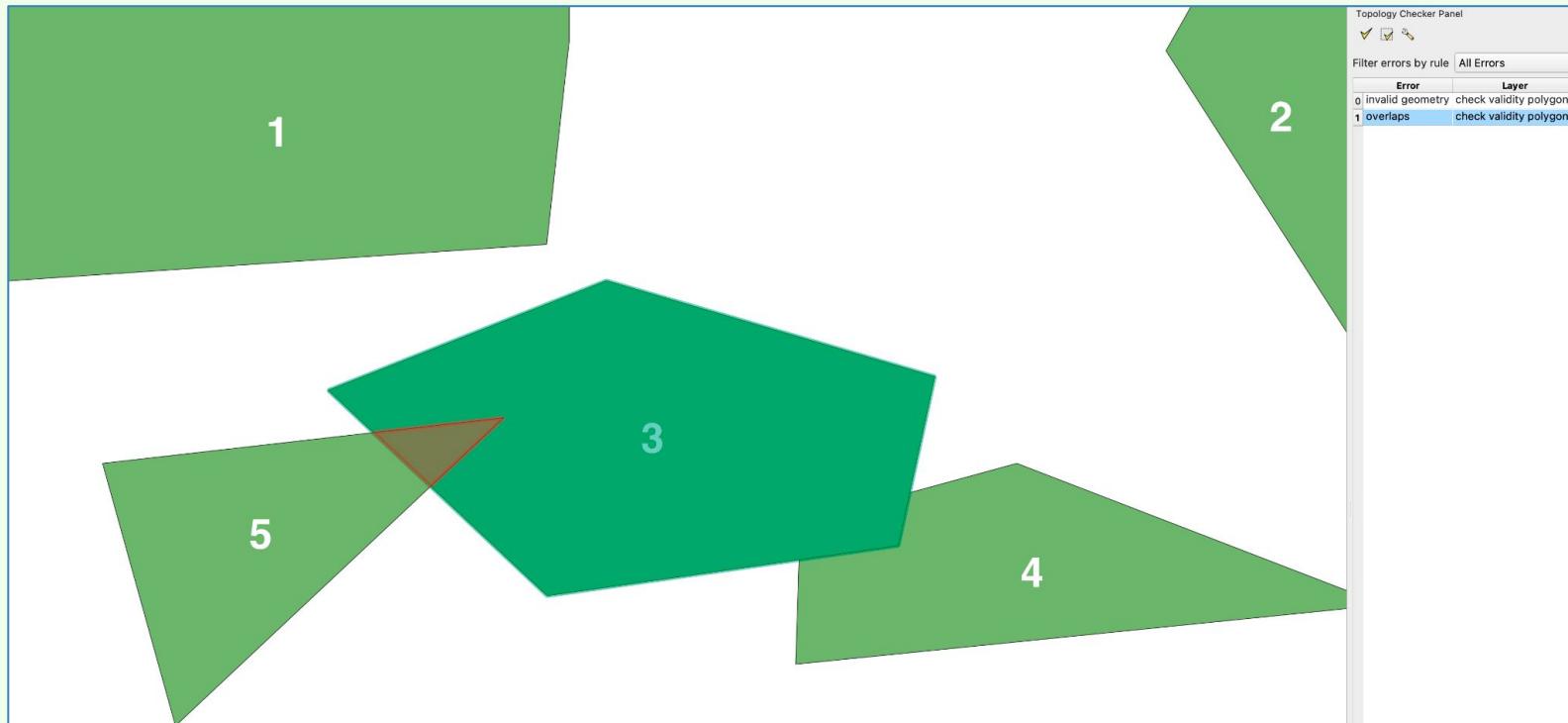


Quando se aplica o Topology Checker plugin ao exemplo anterior identificam-se e localizam-se dois erros. Um é geométrico (invalid geometry) e o outro é topológico (overlaps)...

Filter errors by rule		
Error	Layer	Feature ID
0 invalid geometry	check validity polygons	1
1 overlaps	check validity polygons	2

De notar que as Feature ID não têm qualquer correspondência com as ID originais das features...

# Topology Checker...



Selecionando cada um dos erros que estão listados na tabela de erros que está à direita, o mesmo surge destacado na janela principal do QGIS 3.

Neste exemplo seleccionou-se o overlap entre os polígonos 3 e 5.

Os dois polígonos surgem com tonalidades de verde distintas, e o overlap a *bordeaux*...

# introdução ao [mapshaper.org](http://mapshaper.org)



# mapshaper.org

A identificação e localização de erros é uma tarefa que pode ser concretizada no **QGIS 3** com alguma facilidade, como se viu anteriormente. Outra tarefa distinta é a correcção desses erros, que pode ser levada a cabo manualmente, ou de forma automática ou semi-automática.

O [mapshaper.org](#) é uma **aplicação online** que não faz parte do **QGIS 3** e que pode ser utilizada com **razoável sucesso** para a correcção automática de erros.

The screenshot shows the 'Import files' dialog box from the mapshaper.org website. At the top left is the 'mapshaper' logo. To the right are links for 'Wiki' and 'GitHub'. The dialog box has a title 'Import files' with a close button ('X') at the top right. Below the title is a checkbox labeled 'with advanced options'. A large text area contains the instruction: 'Drop, paste or select files to import.' followed by 'Shapefile, GeoJSON, TopoJSON, KML and CSV formats are supported. Files can be zipped or gzipped.' The entire dialog box is set against a light blue background.

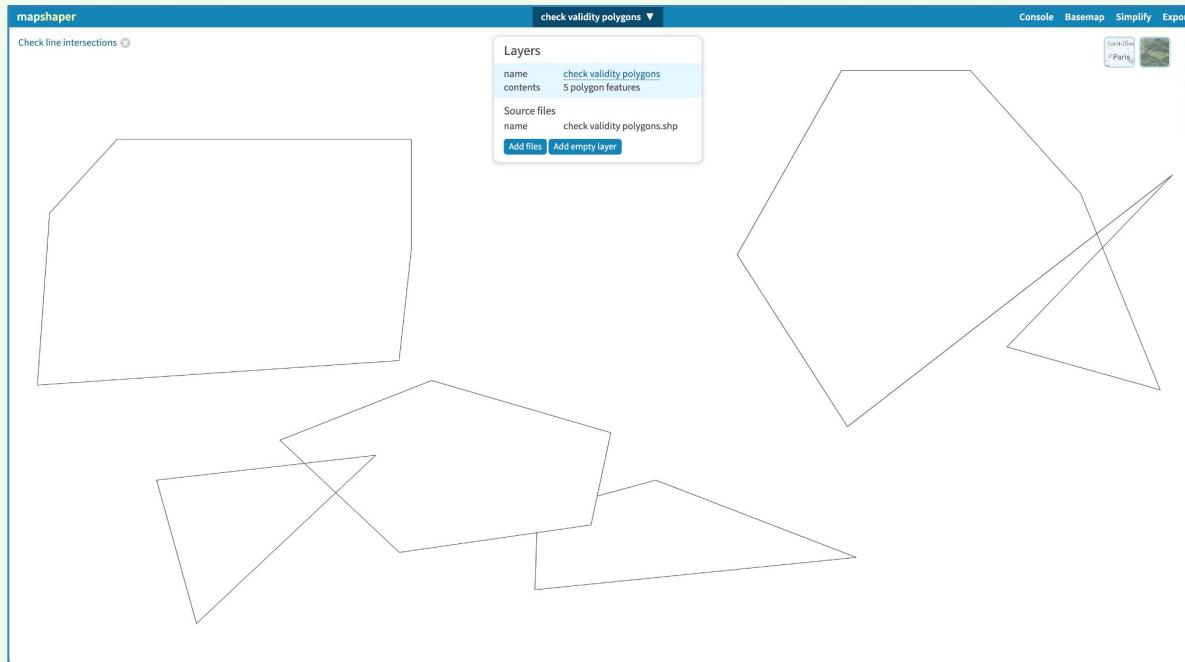
MANUAL DE UTILIZAÇÃO DISPONÍVEL [AQUI](#).



# mapshaper.org

O primeiro passo é a importação da shapefile para o mapshaper.org.

**Devem ser seleccionados todos os ficheiros que constituem a layer e não apenas o ficheiro de extensão .shp (no caso de uma shapefile).**



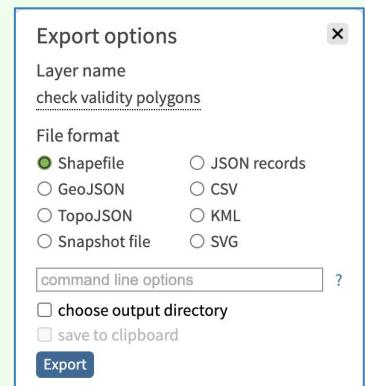
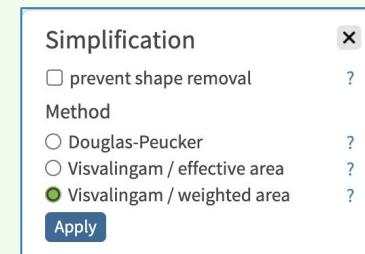
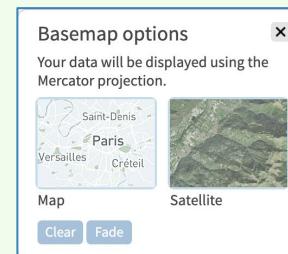
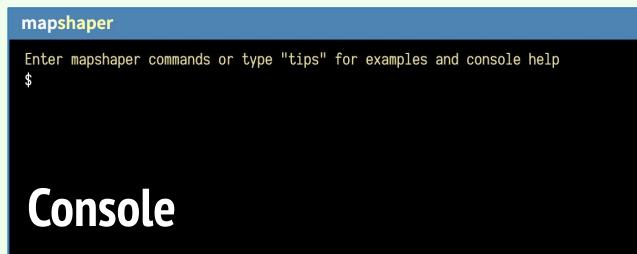
A shapefile de polígonos que tem estado a ser utilizada como exemplo já foi importada para o mapshaper.org.

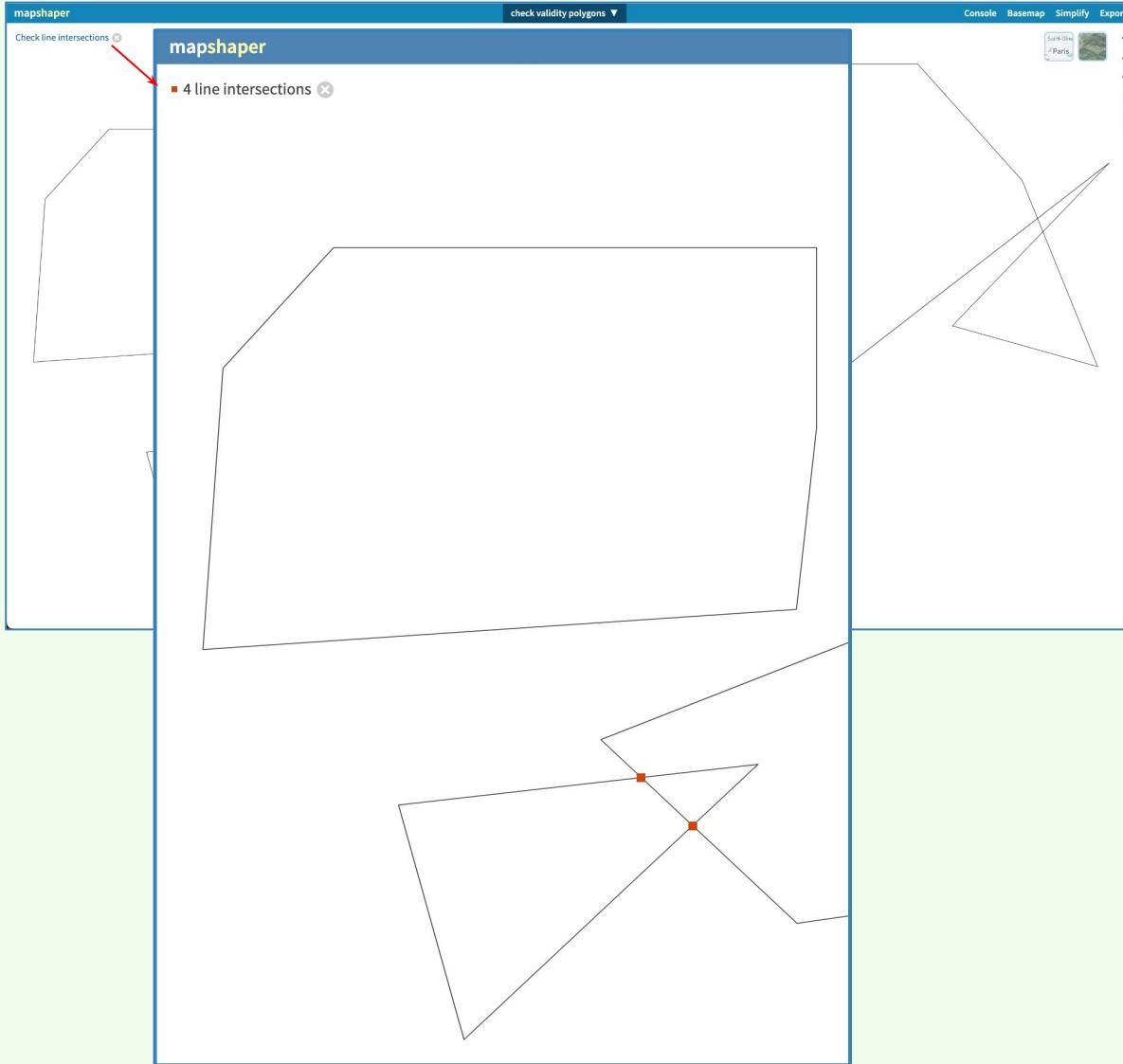
**O botão que existe no centro da barra superior permite saber alguma informação detalhada sobre a shapefile...**

No lado direito da barra superior existem mais quatro botões:

- Console
- Basemap
- Simplify \*\*
- Export

\* ver também Vector >> Geometry Tools >> Simplify...





Depois de importada a shapefile surge no canto superior esquerdo uma linha **Check line intersections**. clicando sobre ela é possível saber o número e a localização dos erros.

O segundo passo é levado a cabo a partir do menu **Console**. Escrevendo **help** na linha de comandos é possível ter acesso a todos os comandos que estão disponíveis...

```
mapshaper
Enter mapshaper commands or type "tips" for examples and console help
$ help
Usage: mapshaper -<command> [options] ...

I/O commands
-i           input one or more files
-o           output edited content

Editing commands
-affine      transform coordinates by shifting, scaling and rotating
-classify   assign colors or values using one of several methods
-clean       fixes geometry issues, such as polygon overlaps and gaps
-clip        use a polygon layer to clip another layer
-colorizer  define a function to convert data values to color classes
-dashlines   split lines into sections, with or without a gap
-dissolve    merge features within a layer
-dissolve2   merge adjacent polygons (repairs overlaps and gaps)
-divide     divide lines by polygons, copy polygon data to lines
-dots       fill polygons with dots of one or more colors
-drop       delete layer(s) or elements within the target layer(s)
-each       create/update/delete data fields using a JS expression
-erase      use a polygon layer to erase another layer
-explode    divide multi-part features into single-part features
-filter     delete features using a JS expression
-filter-fields retain a subset of data fields
-filter-islands remove small detached polygon rings (islands)
-filter-silvers remove small polygon rings
-graticule  create a graticule layer
-grid       create a grid of square or hexagonal polygons
-include    import JS data and functions for use in JS expressions
-inlay     inscribe a polygon layer inside another polygon layer
-innerlines convert polygons to polylines along shared edges
-join      join data records from a file or layer to a layer
-lines     convert a polygon or point layer to a polyline layer
-merge-layers merge multiple layers into a few layers as possible
-mosaic    convert a polygon layer with overlaps into a flat mosaic
-point-grid create a rectangular grid of points
-points   create a point layer from a different layer type
```



O comando que permite corrigir erros geométricos e topológicos é o **clean**.

Na linha de comandos é possível escrever **help clean** para obter mais informação sobre a funcionalidade...

```
Enter mapshaper -help <command> to view options for a single command
$ help clean
COMMAND
-clean           fixes geometry issues, such as polygon overlaps and gaps

OPTIONS
gap-fill-area=   threshold for filling gaps, e.g. 1.5km2 (default is small)
sliver-control=  boost gap-fill-area of slivers (0-1, default is 1)
snap-interval=   snapping distance in source units (default is tiny)
allow-overlaps   allow polygons to overlap (disables gap fill)
overlap-rule=    how to resolve overlaps: min-id|max-id|min-area|[max-area]
allow-empty      keep null geometries (removed by default)
rewind          fix errors in the CW/CCW winding order of polygon rings
only-arcs       delete unused arcs but don't remove gaps and overlaps
target=         layer(s) to target (comma-sep. list)
$ |
```

Para executar o comando basta escrever o seu nome na linha de comandos, sem ou com as opções adicionais!

# mapshaper.org

```
$ clean  
[clean] Removed 1 of 1 sliver using 7600+ sqm variable threshold  
[clean] Retained 5 of 5 features  
$
```

Após concluir a operação surge a indicação das **correcções efectuadas**.

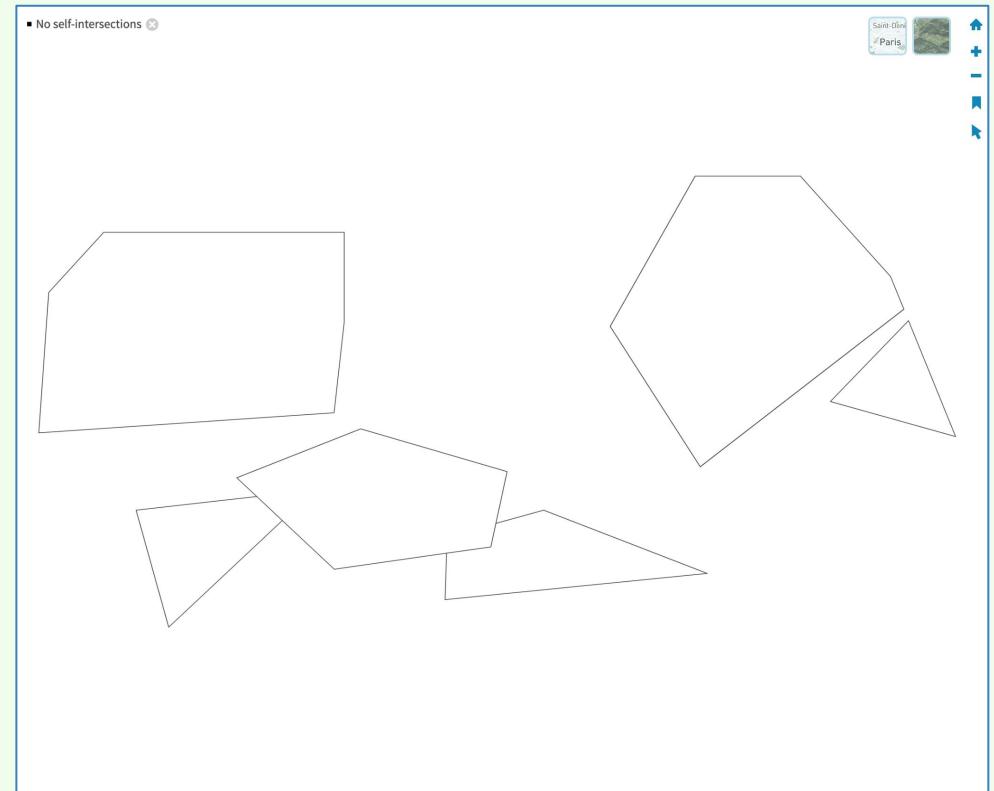
A representação gráfica também é actualizada, e é possível observar e analisar em detalhe o sucesso alcançado com a utilização do mapshaper.org.

Por vezes é necessário recorrer às options do comando. Por exemplo:

**clean gap-fill-area=2km2**

para preencher todos os gaps (vazios ou lacunas) com área até 2 km<sup>2</sup>.

O terceiro e último passo é a **exportação** da layer, que poderá voltar a ser importada e analisada no **QGIS 3...**



# ligações úteis

- QGIS 3 (originalmente designado de Quantum GIS) - [ligação](#)
- Universidade do Algarve - [ligação](#)
- QGIS - tutoriais by nsloureiro.pt - [ligação](#)
- mapshaper.org- [ligação](#)

*Se tiver dúvidas, quiser fazer sugestões ou recomendar alterações não deixe de contactar!*

