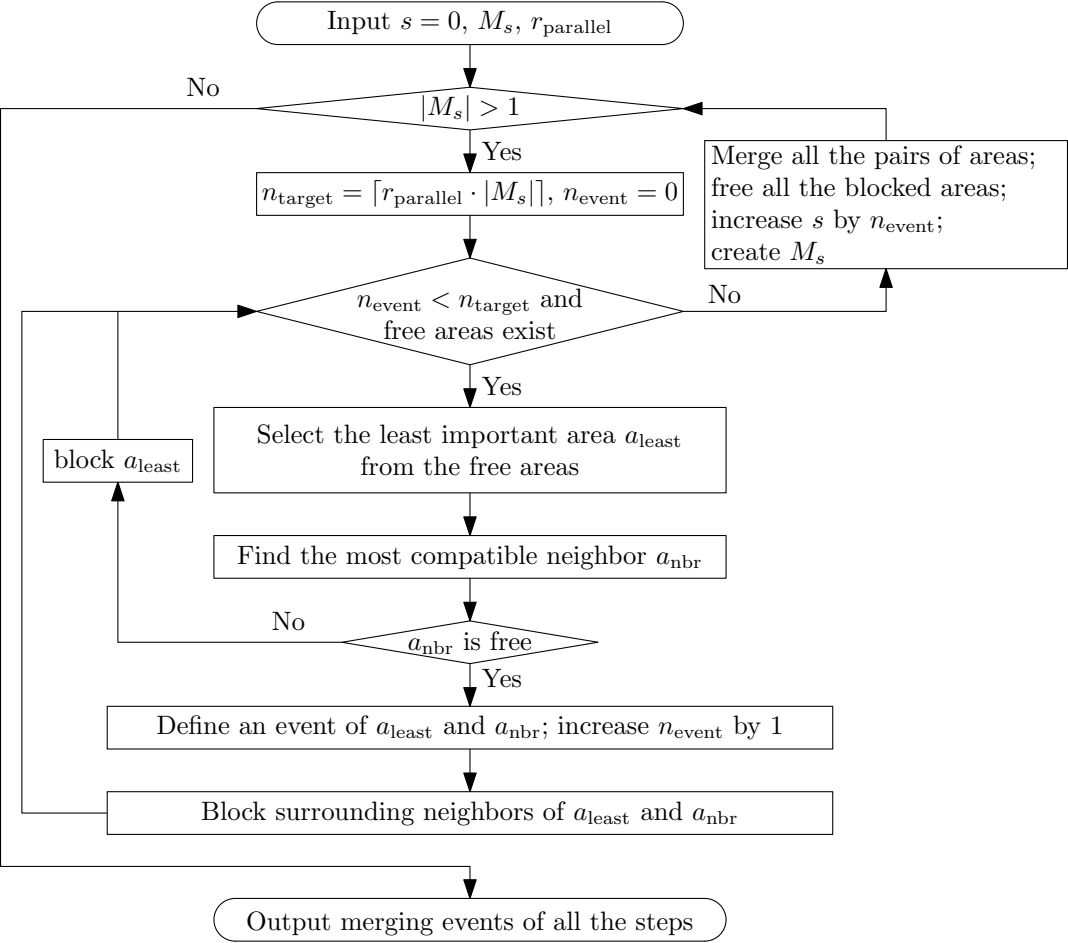
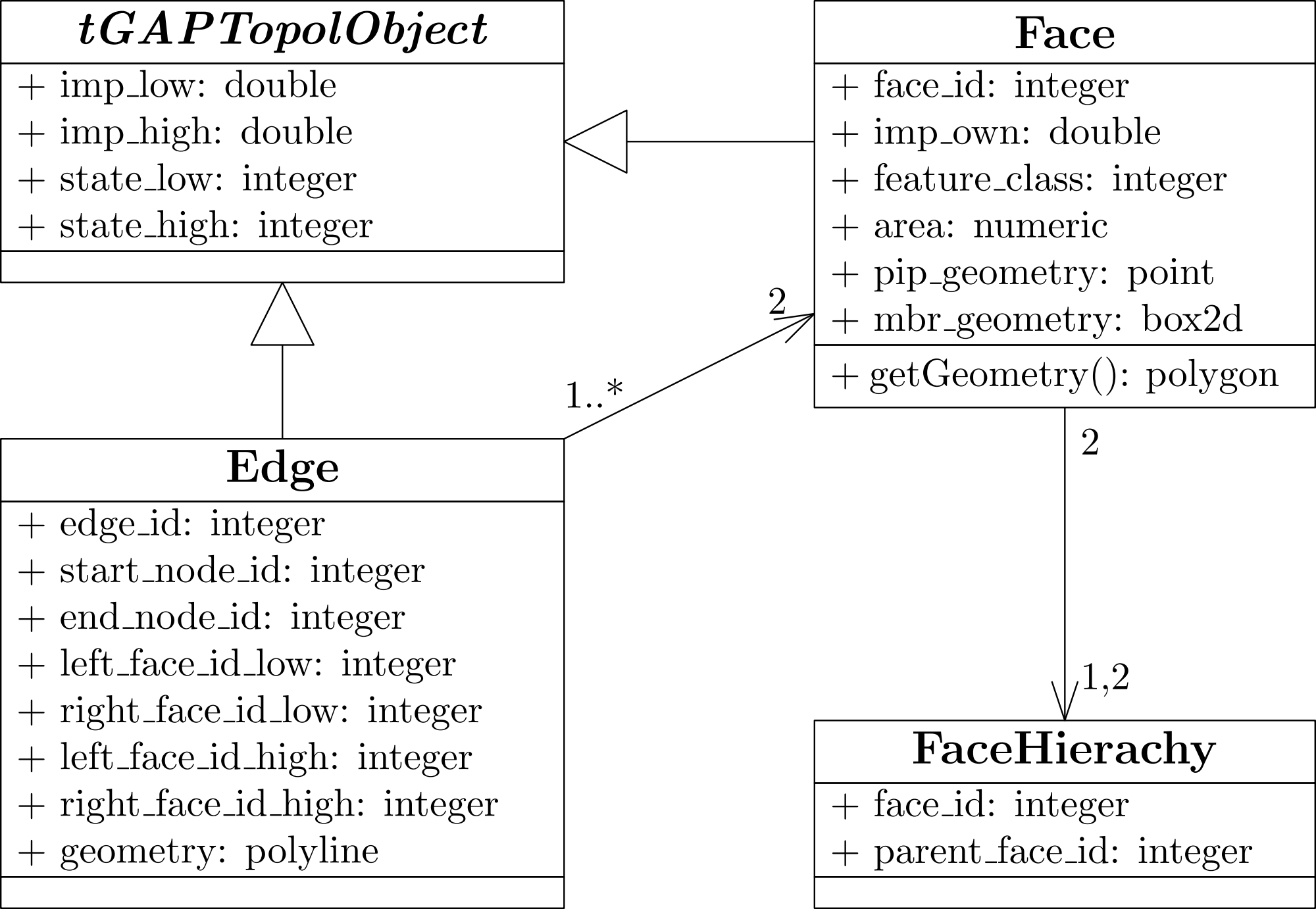
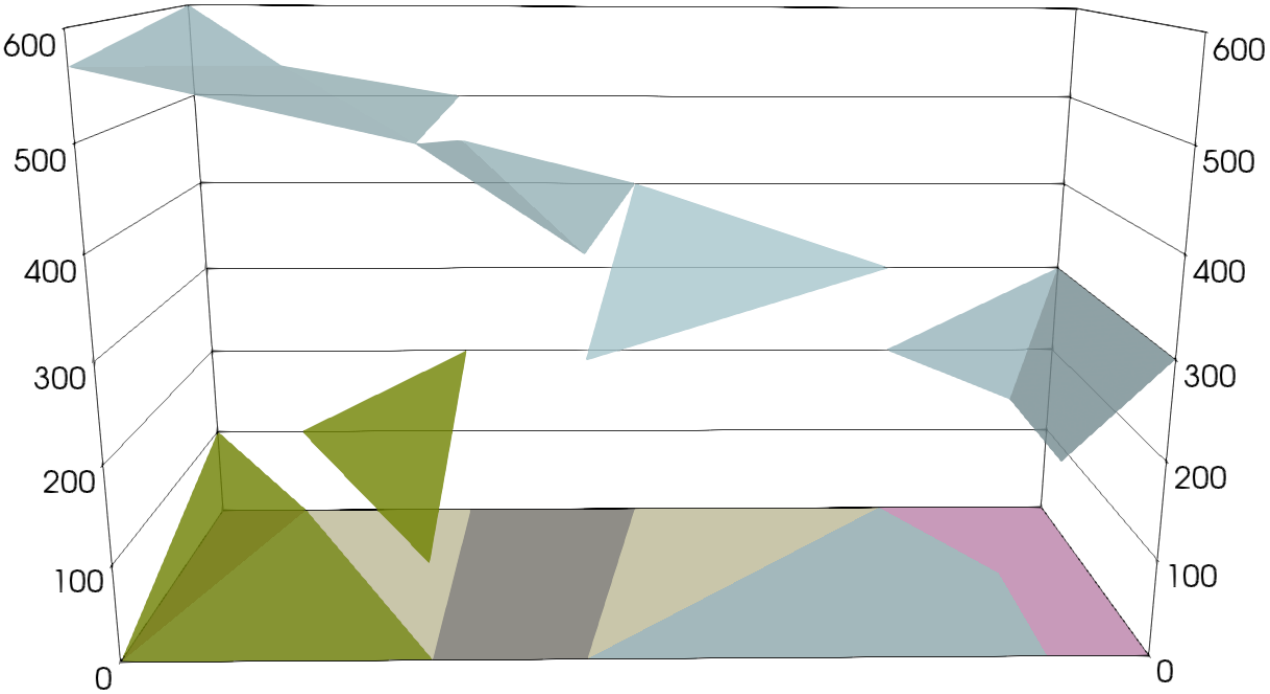
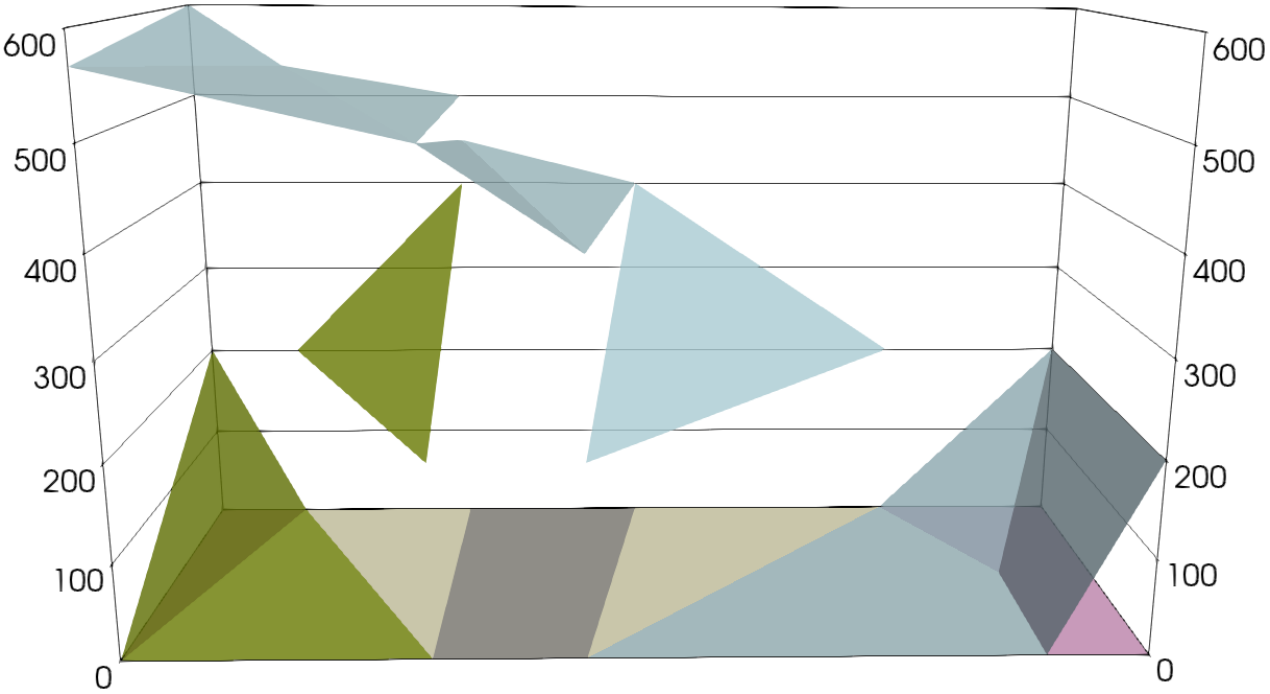

Input $s = 0$, M_s, r_{parallel}
No
$|M_s| > 1$
Yes
$n_{\text{target}} = \lceil r_{\text{parallel}} \cdot |M_s| \rceil$, $n_{\text{event}} = 0$
Merge all the pairs of areas;
free all the blocked areas;
increase s by n_{event};
create M_s
$n_{\text{event}} < n_{\text{target}}$ and
free areas exist
No
Yes
block a_{least}
Select the least important area a_{least}
from the free areas
Find the most compatible neighbor a_{nbr}
No
a_{nbr} is free
Yes
Define an event of a_{least} and a_{nbr}; increase n_{event} by 1
Block surrounding neighbors of a_{least} and a_{nbr}
Output merging events of all the steps

(a)
(b)


tGAPTopolObject
+ imp_low: double
+ imp_high: double
+ state_low: integer
+ state_high: integer
Face
+ face_id: integer
+ imp_own: double
+ feature_class: integer
+ area: numeric
+ pip_geometry: point
+ mbr_geometry: box2d
+ getGeometry(): polygon
Edge
+ edge_id: integer
+ start_node_id: integer
+ end_node_id: integer
+ left_face_id_low: integer
+ right_face_id_low: integer
+ left_face_id_high: integer
+ right_face_id_high: integer
+ geometry: polyline
FaceHierachy
+ face_id: integer
+ parent_face_id: integer
2
1..*
2
1,2


600
500
400
300
200
100
0
600
500
400
300
200
100
0


600
500
400
300
200
100
0
600
500
400
300
200
100
0

Interaction settings

zooming factor

0.25 0.5 1 2 4

zooming duration (short-long)

short (0s) long (5s)

panning duration (short-long)

short (0s) long (5s)

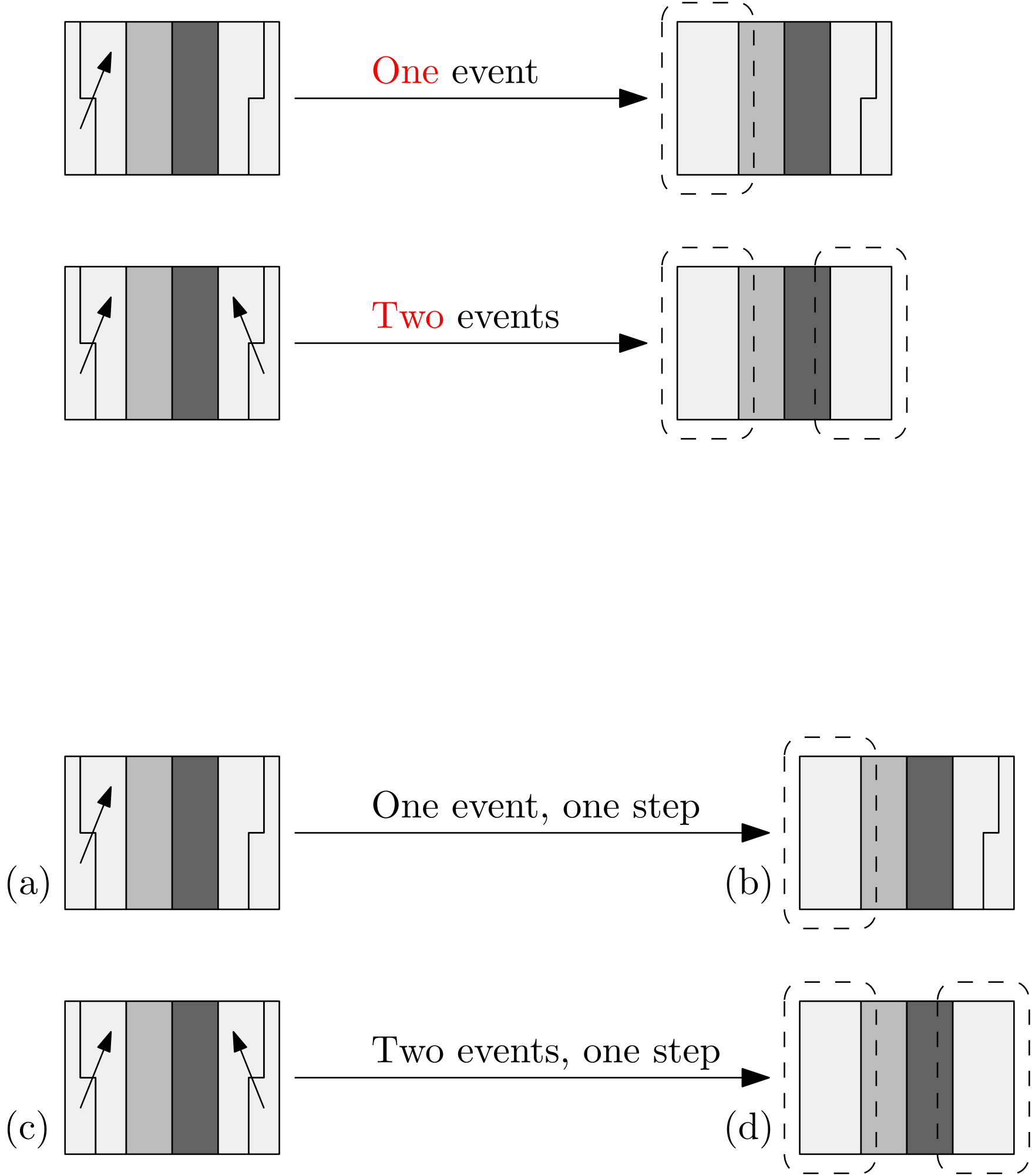
Displaying settings

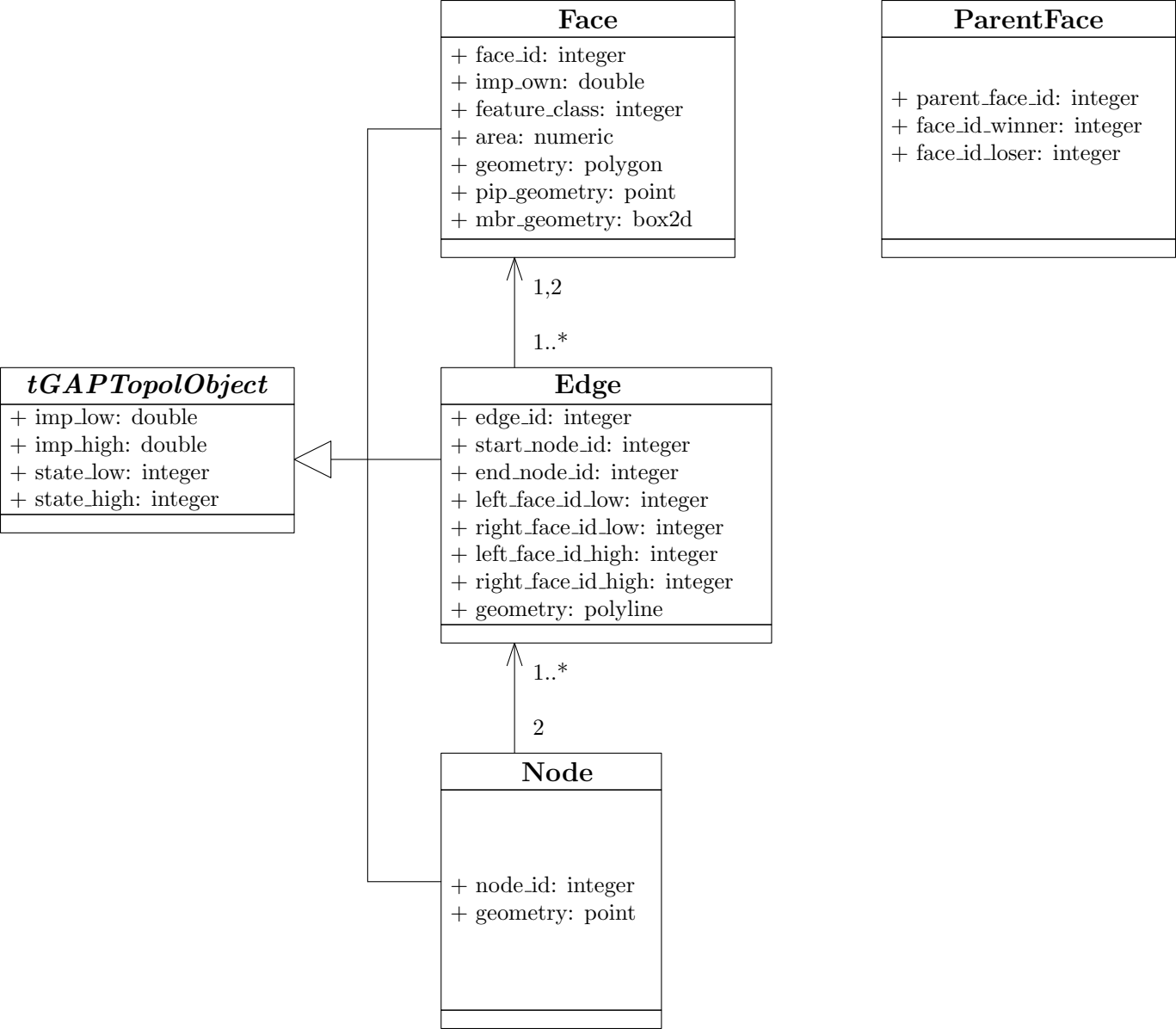
boundary width

line-width value (mm): 0.2

narrow wide

Close


One event
Two events
(a)
One event, one step
(b)
(c)
Two events, one step
(d)


Face
+ face_id: integer
+ imp_own: double
+ feature_class: integer
+ area: numeric
+ geometry: polygon
+ pip_geometry: point
+ mbr_geometry: box2d
ParentFace
+ parent_face_id: integer
+ face_id_winner: integer
+ face_id_loser: integer
1,2
1..*
tGAPTopolObject
+ imp_low: double
+ imp_high: double
+ state_low: integer
+ state_high: integer
Edge
+ edge_id: integer
+ start_node_id: integer
+ end_node_id: integer
+ left_face_id_low: integer
+ right_face_id_low: integer
+ left_face_id_high: integer
+ right_face_id_high: integer
+ geometry: polyline
1..*
2
Node
+ node_id: integer
+ geometry: point