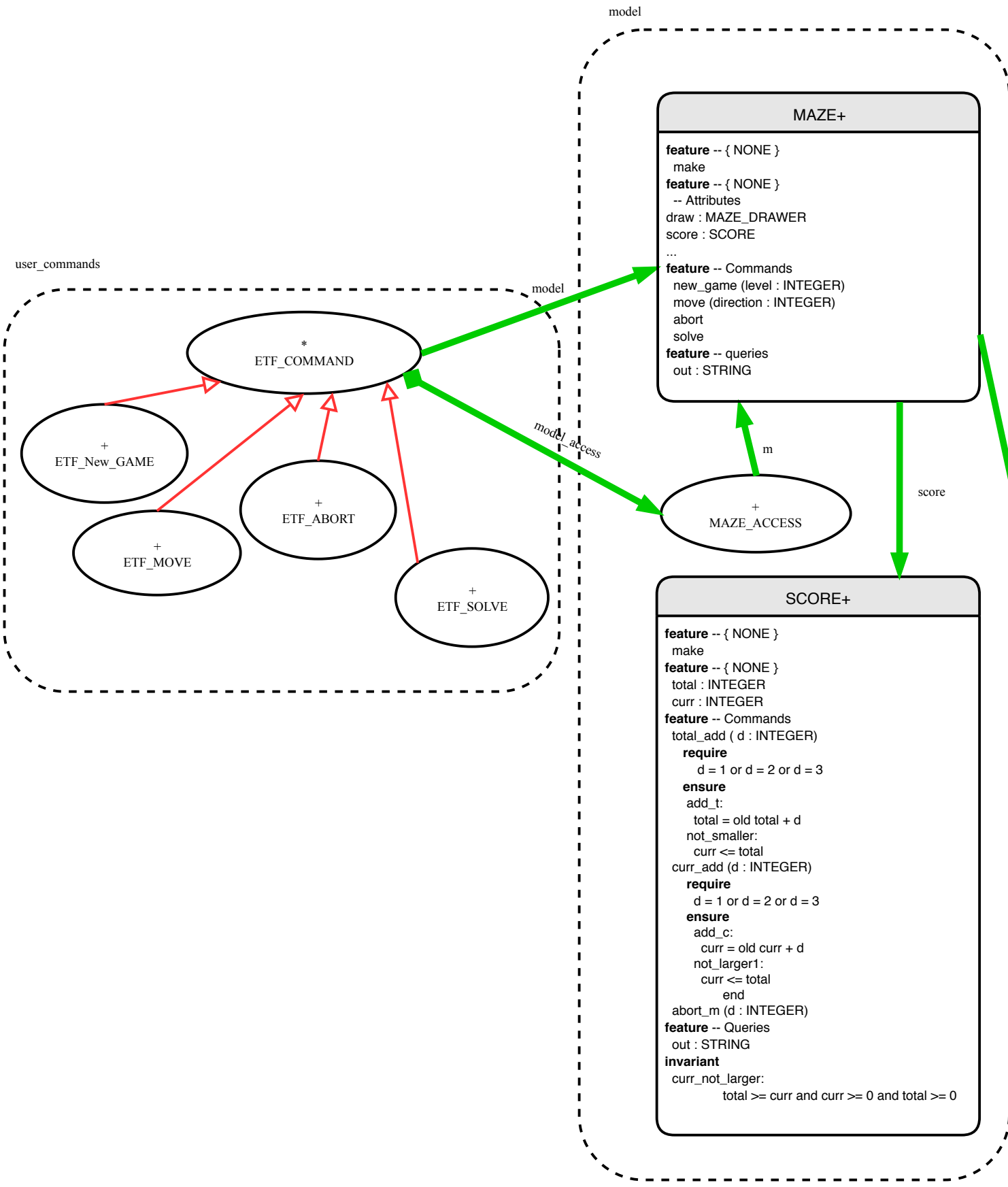
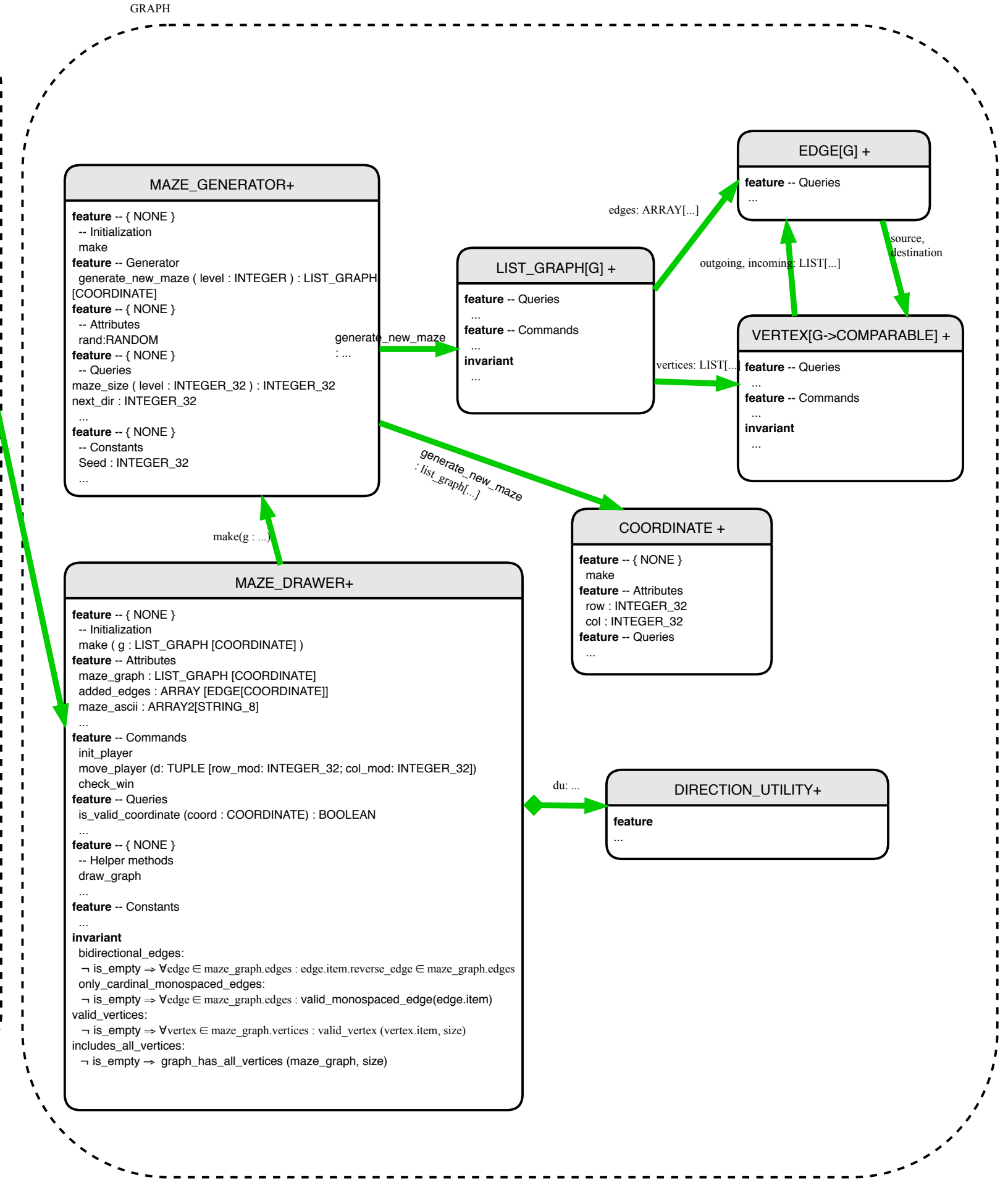

model
MAZE+
feature -- { NONE }
make
feature -- { NONE }
-- Attributes
draw : MAZE_DRAWER
score : SCORE
...
feature -- Commands
new_game (level : INTEGER)
move (direction : INTEGER)
abort
solve
feature -- queries
out : STRING
user_commands
model
*
ETF_COMMAND
+
ETF_New_GAME
+
ETF_ABORT
+
ETF_MOVE
+
ETF_SOLVE
model_access
m
+
MAZE_ACCESS
score
SCORE+
feature -- { NONE }
make
feature -- { NONE }
total : INTEGER
curr : INTEGER
feature -- Commands
total_add (d : INTEGER)
require
d = 1 or d = 2 or d = 3
ensure
add_t:
total = old total + d
not_smaller:
curr <= total
curr_add (d : INTEGER)
require
d = 1 or d = 2 or d = 3
ensure
add_c:
curr = old curr + d
not_larger1:
curr <= total
end
abort_m (d : INTEGER)
feature -- Queries
out : STRING
invariant
curr_not_larger:
total >= curr and curr >= 0 and total >= 0


GRAPH
MAZE_GENERATOR+
feature -- { NONE }
-- Initialization
make
feature -- Generator
generate_new_maze (level : INTEGER) : LIST_GRAPH [COORDINATE]
feature -- { NONE }
-- Attributes
rand:RANDOM
feature -- { NONE }
-- Queries
maze_size (level : INTEGER_32) : INTEGER_32
next_dir : INTEGER_32
...
feature -- { NONE }
-- Constants
Seed : INTEGER_32
...
generate_new_maze : ...
LIST_GRAPH[G] +
feature -- Queries
...
feature -- Commands
...
invariant
...
edges: ARRAY[...]
EDGE[G] +
feature -- Queries
...
source, destination
outgoing, incoming: LIST[...]
vertices: LIST[...]
VERTEX[G->COMPARABLE] +
feature -- Queries
...
feature -- Commands
...
invariant
...
generate_new_maze : list_graph[...]
COORDINATE +
feature -- { NONE }
make
feature -- Attributes
row : INTEGER_32
col : INTEGER_32
feature -- Queries
...
make(g : ...)
MAZE_DRAWER+
feature -- { NONE }
-- Initialization
make (g : LIST_GRAPH [COORDINATE])
feature -- Attributes
maze_graph : LIST_GRAPH [COORDINATE]
added_edges : ARRAY [EDGE[COORDINATE]]
maze_ascii : ARRAY2[STRING_8]
...
feature -- Commands
init_player
move_player (d: TUPLE [row_mod: INTEGER_32; col_mod: INTEGER_32])
check_win
feature -- Queries
is_valid_coordinate (coord : COORDINATE) : BOOLEAN
...
feature -- { NONE }
-- Helper methods
draw_graph
...
feature -- Constants
...
invariant
bidirectional_edges:
¬ is_empty ⇒ ∀edge ∈ maze_graph.edges : edge.item.reverse_edge ∈ maze_graph.edges
only_cardinal_monospaced_edges:
¬ is_empty ⇒ ∀edge ∈ maze_graph.edges : valid_monospaced_edge(edge.item)
valid_vertices:
¬ is_empty ⇒ ∀vertex ∈ maze_graph.vertices : valid_vertex (vertex.item, size)
includes_all_vertices:
¬ is_empty ⇒ graph_has_all_vertices (maze_graph, size)
du: ...
DIRECTION_UTILITY+
feature
...