
TrafficTracer

Release v1.0.0

KnapSac, Hackerman, ManoukSirag, ikbenwasmachine, Storm506

Jan 28, 2020

API REFERENCE:

1	Doxygen Index	1
2	Contribute	29
2.1	Indices and tables	29
	Index	31

DOXYGEN INDEX

`namespace TrafficJam`

`class Helper`

Public Static Functions

`static string RouteToString (Route route)`

`static float AngleBetween (PointF source, PointF destination)`

`static void CreateMapPreview (Image map, int mapId)`

`class Program`

Private Static Functions

`static void Main ()`

`class TrafficTracer`

Public Functions

`TrafficTracer ()`

`void InitializeSimulation (int situationId, int framesBeforeVehicle, int heavyTrafficPercentage)`

`void ShowResetWindow (MainWindow ownerWindow, bool simulationRunning)`

`void Reset (bool softReset, bool fromResetButton)`

Property

```
property TrafficJam::TileLength
property TrafficJam::TileResourceManager
property TrafficJam::Monitor
property TrafficJam::StartupWindow
property TrafficJam::MainWindow
property TrafficJam::Map
property TrafficJam::Controller
property TrafficJam::VehicleFactory
property TrafficJam::Seed
property TrafficJam::HeavyTrafficPercentage
property TrafficJam::FramesBeforeVehicle
```

Private Functions

```
void CreateStartupWindow ()
void CreateSimulationMonitor ()
void CreateMap (int mapId)
void CreateVehicleFactory ()
void CreateSimulationController ()
void CreateMainWindow ()
```

namespace Environment

```
class Map: public Panel
```

Public Functions

```
Map (TrafficTracer trafficTracer, int mapId)
Road GetTile (PointF point)
List<Road> CalculateRoute (EntryPoint entryPoint)
void DrawNextFrame (Bitmap nextFrame)
List<EntryPoint> GetAvailableEntryPoints ()
void UpdateEntryPointInflow (int entryPointId, int framesBeforeVehicle)
```

Property

```
property TrafficJam::Environment::EntryPointsCount
property TrafficJam::Environment::TrafficTracer
```

Protected Functions

```
override void OnMouseClicked(MouseEventArgs e)
override void OnMouseMove(MouseEventArgs e)
```

Private Functions

```
void Map_Paint (object sender, PaintEventArgs e)
Road GetTile (Point point)
Grid CalculateGrid ()
void SelectMap (int mapId)
void CreateEntryPoints (int mapId)
void StoreEntryPoints (List<Road> entryPoints)
void SelectRoadMap1 (Dictionary<(int, int), Road> dict)
void SelectRoadMap2 (Dictionary<(int, int), Road> dict)
void SelectRoadMap3 (Dictionary<(int, int), Road> dict)
Bitmap DrawRoads (Dictionary<(int, int), Road> roadTiles)
int SelectRandomDestination (int entryPointIndex, int exitPointIndex)
```

Private Members

```
readonly TrafficTracer _trafficTracer
const int HorizontalTiles = 8
const int VerticalTiles = 6
readonly Size _tileSize
readonly Dictionary<(int X, int Y), Road> _roadTiles = new Dictionary<(int X, int Y), Road> ()
readonly List<EntryPoint> _entryPoints = new List<EntryPoint> ()
Grid _grid
Road _selectedTile
Bitmap _nextFrame
int _mapId

namespace Roads
```

```
class CurvedRoad0 : public Road
```

Public Functions

```
CurvedRoad0 (Map map, (int X, int Y) position)
override bool ToolAllowed (Tool tool)
override bool DirectionAllowed (AnimationDirection direction)
override string ToString ()
```

```
class CurvedRoad180 : public Road
```

Public Functions

```
CurvedRoad180 (Map map, (int X, int Y) position)
override bool ToolAllowed (Tool tool)
override bool DirectionAllowed (AnimationDirection direction)
override string ToString ()
```

```
class CurvedRoad270 : public Road
```

Public Functions

```
CurvedRoad270 (Map map, (int X, int Y) position)
override bool ToolAllowed (Tool tool)
override bool DirectionAllowed (AnimationDirection direction)
override string ToString ()
```

```
class CurvedRoad90 : public Road
```

Public Functions

```
CurvedRoad90 (Map map, (int X, int Y) position)
override bool ToolAllowed (Tool tool)
override bool DirectionAllowed (AnimationDirection direction)
override string ToString ()
```

```
class EntryPoint
```


Public Functions

EntryPoint (Road *road*, int *framesBeforeVehicle*)

bool **CanGenerateVehicle** ()

Property

property TrafficJam::Environment::Roads::FramesBeforeVehicle

property TrafficJam::Environment::Roads::Road

Private Members

int **_vehicleGeneratedFrameCount**

int **_framesBeforeVehicle**

class Grass:public *Road*

Public Functions

Grass(Map map, (int X, int Y) position)

override bool ToolAllowed(Tool tool)

override bool DirectionAllowed(AnimationDirection direction)

class HorizontalRoad:public *Road*

Public Functions

HorizontalRoad(Map map, (int X, int Y) position)

override bool ToolAllowed(Tool tool)

override bool DirectionAllowed(AnimationDirection direction)

class Road

Subclassed by *CurvedRoad0*, *CurvedRoad180*, *CurvedRoad270*, *CurvedRoad90*, *Grass*, *HorizontalRoad*, *Junction*, *VerticalRoad*

Public Functions

abstract bool ToolAllowed(Tool tool)= 0

abstract bool DirectionAllowed(AnimationDirection direction)= 0

bool **IsEntryPoint** ()

void **CreateEntryPoint** (int *framesBeforeVehicle*)

Public Members

int X

Property

```
property TrafficJam::Environment::Roads::Position
property TrafficJam::Environment::Roads::Texture
property TrafficJam::Environment::Roads::Location
property TrafficJam::Environment::Roads::Tools
property TrafficJam::Environment::Roads::HasTool
property TrafficJam::Environment::Roads::EntryPoint
```

Protected Functions

```
Road(Map map, (int X, int Y) position, string textureName)
```

Private Functions

```
readonly (int X, int Y)
Image SetTexture (Map map, string textureName)
```

Private Members

```
readonly Point _location
readonly Image _texture
EntryPoint _entryPoint
List<Tool> _tools

class VerticalRoad: public Road
```

Public Functions

```
VerticalRoad(Map map, (int X, int Y) position)
override bool ToolAllowed(Tool tool)
override bool DirectionAllowed(AnimationDirection direction)

namespace Junctions

class Intersection: public Junction
```

Public Functions

```
Intersection(Map map, (int X, int Y) position)
override bool ToolAllowed(Tool tool)
override bool DirectionAllowed(AnimationDirection direction)
```

```
class Junction : public Road
    Subclassed by Intersection, Junction0, Junction180, Junction270, Junction90
```

Public Functions

```
void Enqueue (Vehicle vehicle, AnimationDirection direction)
void Dequeue (AnimationDirection direction)
bool QueuesEmpty (AnimationDirection myDirection)
Point GetStoppingPoint (AnimationDirection myDirection)
```

Event

```
event TrafficJam::Environment::Roads::Junctions::VehicleEnqueued
event TrafficJam::Environment::Roads::Junctions::VehicleDequeued
```

Protected Functions

```
Junction(Map map, (int X, int Y) position, string textureName, AnimationDir
virtual void OnVehicleEnqueued (VehicleEnqueuedEventArgs e)
virtual void OnVehicleDequeued (VehicleEnqueuedEventArgs e)
```

Private Functions

```
ConcurrentQueue<Vehicle> GetQueueByDirection (AnimationDirection direction)
void InitializeQueues (AnimationDirection directions)
```

Private Members

```
ConcurrentQueue<Vehicle> _queueTop
ConcurrentQueue<Vehicle> _queueRight
ConcurrentQueue<Vehicle> _queueBottom
ConcurrentQueue<Vehicle> _queueLeft
class Junction0 : public Junction
```

Public Functions

```
Junction0(Map map, (int X, int Y) position)
override bool ToolAllowed(Tool tool)
override bool DirectionAllowed(AnimationDirection direction)
override string ToString()
class Junction180 : public Junction
```

Public Functions

```
Junction180(Map map, (int X, int Y) position)
override bool ToolAllowed(Tool tool)
override bool DirectionAllowed(AnimationDirection direction)
override string ToString()
class Junction270 : public Junction
```

Public Functions

```
Junction270(Map map, (int X, int Y) position)
override bool ToolAllowed(Tool tool)
override bool DirectionAllowed(AnimationDirection direction)
override string ToString()
class Junction90 : public Junction
```

Public Functions

```
Junction90(Map map, (int X, int Y) position)
override bool ToolAllowed(Tool tool)
override bool DirectionAllowed(AnimationDirection direction)
override string ToString()
```

namespace Events

```
class SimulationResetEventArgs : public EventArgs
```

Public Functions

`SimulationResetEventArgs()`

`SimulationResetEventArgs (bool softReset)`

Property

`property TrafficJam::Events::SoftReset`

`class VehicleEnqueuedEventArgs : public EventArgs`

Public Functions

`VehicleEnqueuedEventArgs()`

`VehicleEnqueuedEventArgs (AnimationDirection direction)`

Property

`property TrafficJam::Events::QueueDirection`

`class VehicleStateEventArgs : public EventArgs`

Public Functions

`VehicleStateEventArgs (Vehicle vehicle, VehicleState oldState, VehicleState newState)`

Property

`property TrafficJam::Events::Vehicle`

`property TrafficJam::Events::OldState`

`property TrafficJam::Events::NewState`

Private Members

`readonly Vehicle _vehicle`

`readonly VehicleState _oldState`

`readonly VehicleState _newState`

`namespace Exceptions`

`class ControllerUnavailableException : public Exception`

Public Functions

`ControllerUnavailableException` (string *message*)

`class InvalidDirectionException : public Exception`

Public Functions

`InvalidDirectionException` (Type *roadType*, AnimationDirection *direction*)

`InvalidDirectionException` (AnimationDirection *direction*)

Property

`property TrafficJam::Exceptions::RoadType`

`property TrafficJam::Exceptions::Direction`

`class VehicleCreationFailedException : public Exception`

Public Functions

`VehicleCreationFailedException` (string *message*, *InvalidDirectionException* *innerException*)

`namespace Forms`

`class BufferPanel : public Panel`

Public Functions

`BufferPanel()`

`class EntryPointsPanel : public Panel`

Public Functions

`EntryPointsPanel` (*MainWindow* *mainWindow*)

Private Members

`readonly MainWindow _mainWindow`

`class EntryPointsSub : public Panel`

Public Functions

EntryPointsSub (*EntryPointsPanel* entryPointsPanel, int y)

Private Members

readonly **EntryPointsPanel** _entryPointsPanel

readonly **Font** _trebuchetSmall = **new** **Font** ("Trebuchet MS", 8f)

readonly **Font** _trebuchetLarge = **new** **Font** ("Trebuchet MS", 11f)

class **MainWindow** : **public** **Form**, **public** **IObserver**<*SimulationInfo*>

Public Functions

MainWindow (*TrafficTracer* trafficTracer)

void **OnNext** (*SimulationInfo* simInfo)

void **OnError** (Exception error)

void **OnCompleted** ()

Property

property **TrafficJam::Forms::TrafficTracer**

property **TrafficJam::Forms::ToolsPanel**

property **TrafficJam::Forms::TrafficFlowPanel**

property **TrafficJam::Forms::EntryPointsPanel**

Protected Functions

override **void** **OnFormClosed**(**FormClosedEventArgs** e)

override **void** **Dispose**(**bool** disposing)

Private Functions

void **CreatePanelBars** ()

void **BorderSizeMenuButtons** (object sender)

void **ChangeButtonMenu** (Button btn)

void **BorderSizeTopButtons** (object sender)

void **ChangeButtonMainWindow** (Button btn)

void **StartSimulationMain_Click** (object sender, EventArgs e)

void **PauseSimulation_Click** (object sender, EventArgs e)

```
void StepSimulation_Click (object sender, EventArgs e)
void ResetSimulation_Click (object sender, EventArgs e)
void ChangeMenuHeader (object sender)
void ToolsMain_Click (object sender, EventArgs e)
void TrafficFlowMain_Click (object sender, EventArgs e)
void EntryPointsMain_Click (object sender, EventArgs e)
void ShowPanel (Button sender)
void InitializeComponent ()
```

Private Members

```
readonly TrafficTracer _trafficTracer
readonly ToolsPanel _toolsPanel
readonly TrafficFlowPanel _trafficFlowPanel
readonly EntryPointsPanel _entryPointsPanel
bool _simulationStarted
System.ComponentModel.IContainer components = null
System.Windows.Forms.Button stepSimulation
System.Windows.Forms.Button entryPointsMain
System.Windows.Forms.Button trafficFlowMain
System.Windows.Forms.Button toolsMain
System.Windows.Forms.Button resetSimulation
System.Windows.Forms.Button pauseSimulation
System.Windows.Forms.Button startSimulationMain
System.Windows.Forms.Label menuHeader
class ResetWindow: public Form
```

Public Functions

```
ResetWindow (TrafficTracer trafficTracer, bool simulationRunning)
```


Protected Functions

```
override void Dispose(bool disposing)
```

Private Functions

```
void newSituation_Click (object sender, EventArgs e)
```

```
void sameSituation_Click (object sender, EventArgs e)
```

```
void InitializeComponent ()
```

Private Members

```
readonly TrafficTracer _trafficTracer
```

```
readonly bool _simulationRunning
```

```
System.ComponentModel.IContainer components = null
```

```
System.Windows.Forms.Label label1
```

```
System.Windows.Forms.Button newSituation
```

```
System.Windows.Forms.Button sameSituation
```

```
class StartupWindow : public Form
```

Public Functions

```
StartupWindow (TrafficTracer trafficTracer)
```

Protected Functions

```
override void Dispose(bool disposing)
```

Private Functions

```
void ChangeHeader ()
```

```
void ChangeBorderSize ()
```

```
void ChangeButton (Button btn)
```

```
void MoveSelectionBars ()
```

```
void situation1_Click (object sender, EventArgs e)
```

```
void situation2_Click (object sender, EventArgs e)
```

```
void situation3_Click (object sender, EventArgs e)
```

```
void startSimulationStartup_Click (object sender, EventArgs e)
```

```
void UpdateWindow ()
```

```
void InitializeComponent ()
```

Private Members

```
readonly TrafficTracer _trafficTracer
readonly ResourceManager _resourceManager = new ResourceManager( "TrafficJam.Pr
Button _selectedButton
readonly PictureBox _mapPreview
string _headerInformation
const string Situation1Info = "This is a map that functions as example of a realistic traffic situation."
Image _situation1
const string Situation2Info = "This is a map that has many intersections and junctions close together, to simula
Image _situation2
const string Situation3Info = "This map shows a part of the centre of Utrecht. the area in question is the area a
Image _situation3
int _situationId = 1
System.ComponentModel.IContainer components = null
System.Windows.Forms.Label situationInfo
System.Windows.Forms.Label situationHeader
System.Windows.Forms.Button startSimulationStartup
System.Windows.Forms.Label speed100
System.Windows.Forms.Label speed80
System.Windows.Forms.Label speed60
System.Windows.Forms.Label speed40
System.Windows.Forms.Label speed20
System.Windows.Forms.Label speedHeader
System.Windows.Forms.Label manyHeavyTraffic
System.Windows.Forms.Label averageHeavyTraffic
System.Windows.Forms.Label fewHeavyTraffic
System.Windows.Forms.Label heavyTrafficHeader
System.Windows.Forms.TrackBar speedBar
System.Windows.Forms.TrackBar heavyTrafficBar
System.Windows.Forms.Label highTraffic
System.Windows.Forms.Label averageTraffic
System.Windows.Forms.Label lowTraffic
System.Windows.Forms.TrackBar trafficBar
System.Windows.Forms.Label trafficHeader
System.Windows.Forms.Button situation3
System.Windows.Forms.Button situation2
```

```
System.Windows.Forms.Button situation1
System.Windows.Forms.Panel firstSelectedBar
System.Windows.Forms.Panel secondSelectedBar

class ToolsPanel : public Panel
```

Public Functions

```
ToolsPanel (MainWindow mainWindow)
```

Property

```
property TrafficJam::Forms::CurrentTool
```

Private Functions

```
void BorderSizeToolMenu ()
void CrossRoad_Click (object sender, EventArgs e)
void RoadBlock_Click (object sender, EventArgs e)
void SpeedBump_Click (object sender, EventArgs e)
void SpeedSign_Click (object sender, EventArgs e)
void TrafficLight_Click (object sender, EventArgs e)
void throughputMeter_Click (object sender, EventArgs e)
```

Private Members

```
readonly MainWindow _mainWindow
Button _selectedToolButton

readonly Button _crossRoad = new Button ( )
readonly Button _roadBlock = new Button ( )
readonly Button _speedBump = new Button ( )
readonly Button _speedSign = new Button ( )
readonly Button _trafficLight = new Button ( )
readonly Button _throughputMeter = new Button ( )
```

Private Static Functions

```
static void ChangeButton (ButtonBase btn)  
  
class TrafficFlowPanel : public Panel
```

Public Functions

```
TrafficFlowPanel (MainWindow mainWindow)
```

Private Members

```
readonly MainWindow _mainWindow  
  
readonly Font _trebuchetSmall = new Font ( "Trebuchet MS", 8f )  
readonly Font _trebuchetLarge = new Font ( "Trebuchet MS", 11f )  
  
namespace Properties  
  
class RoadTiles
```

Property

```
property TrafficJam::Properties::ResourceManager  
property TrafficJam::Properties::Culture  
property TrafficJam::Properties::Car0  
property TrafficJam::Properties::CurvedRoad  
property TrafficJam::Properties::Grass  
property TrafficJam::Properties::HorizontalRoad  
property TrafficJam::Properties::Intersection  
property TrafficJam::Properties::Junction  
property TrafficJam::Properties::Map1  
property TrafficJam::Properties::Map2  
property TrafficJam::Properties::Map3  
property TrafficJam::Properties::VerticalRoad
```

Private Functions

```
internal RoadTiles ()
```

Private Static Attributes

```
global::System.Resources.ResourceManager resourceMan
```

```
global::System.Globalization.CultureInfo resourceCulture
```

```
namespace Simulation
```

Enums

```
enum SimulationState
```

```
    Values:
```

```
    Undefined
```

```
    Running
```

```
    Paused
```

```
    Resetting
```

```
class EventStore
```

Public Functions

```
template<>
```

```
void Subscribe<T> (T subscriber)
```

```
template<>
```

```
void UnSubscribe<T> (T subscriber)
```

Event

```
event TrafficJam::Simulation::SimulationStarted
```

```
event TrafficJam::Simulation::SimulationPaused
```

```
event TrafficJam::Simulation::SimulationReset
```

Protected Functions

```
virtual void OnSimulationStarted ()
```

```
virtual void OnSimulationPaused ()
```

```
virtual void OnSimulationReset ()
```

```
class SimulationController: public IObservable<SimulationInfo>
```

Public Functions

SimulationController (*TrafficTracer* trafficTracer)

void **OnNext** (*SimulationInfo* simInfo)

void **OnError** (Exception error)

void **OnCompleted** ()

void **StartSimulation** ()

void **Step** ()

Vehicle **FindLeader** (Vector2 myDir, PointF myPos)

RoadTrain **GetOrCreateRoadTrain** (*Vehicle* vehicle)

Property

property TrafficJam::Simulation::State

property TrafficJam::Simulation::Continue

property TrafficJam::Simulation::TrafficTracer

Event

event TrafficJam::Simulation::SimulationStarted

event TrafficJam::Simulation::SimulationPaused

event TrafficJam::Simulation::SimulationReset

Protected Functions

virtual void OnSimulationStarted ()

virtual void OnSimulationPaused ()

virtual void OnSimulationReset ()

Private Functions

void **SetupBackgroundWorker** ()

void **DoWork** (object sender, DoWorkEventArgs e)

void **RunWorkerCompleted** (object sender, RunWorkerCompletedEventArgs e)

Bitmap **CreateNextFrame** ()

void **UpdateVehicles** ()

void **GenerateVehicles** ()

void **AddVehicleManager** ()

Private Members

```
readonly TrafficTracer _trafficTracer
readonly List<VehicleManager> _vehicleManagers
readonly Dictionary<string, RoadTrain> _roadTrains
VehicleManager _availableManager
bool _continue
Bitmap _nextFrame
BackgroundWorker _backgroundWorker
```

```
class SimulationInfo
```

Public Functions

```
SimulationInfo (SimulationState previousState, SimulationState currentState)
SimulationInfo (SimulationState previousState, SimulationState currentState, bool softReset)
```

Property

```
property TrafficJam::Simulation::PreviousState
property TrafficJam::Simulation::CurrentState
property TrafficJam::Simulation::SoftReset
```

```
class SimulationMonitor: public IObservable<SimulationInfo>
```

Public Functions

```
SimulationMonitor()
void SetController (SimulationController controller)
IDisposable Subscribe (IObserver<SimulationInfo> observer)
void StartSimulation()
void PauseSimulation()
void ResetSimulation()
```

Private Functions

```
void NotifyObservers (SimulationInfo simInfo)
```

Private Members

```
readonly List<IObserver<SimulationInfo> > _observers
```

```
SimulationController _controller
```

```
class Unsubscriber : public IDisposable
```

Public Functions

```
Unsubscriber (List<IObserver<SimulationInfo>> observers, IObserver<SimulationInfo>  
              observer)
```

```
void Dispose ()
```

Private Members

```
readonly List<IObserver<SimulationInfo> > _observers
```

```
readonly IObserver<SimulationInfo> _observer
```

```
namespace Tools
```

```
interface IRoadTool
```

```
Subclassed by Tool
```

Public Functions

```
void MouseClicked (Point p)
```

```
void MouseDown (Point p)
```

```
class Tool : public IRoadTool
```

```
Subclassed by CrossRoad, RoadBlock, SpeedBump, SpeedSign, ThroughputMeter, TrafficLight
```

Public Functions

```
virtual void MouseClicked (Point p)
```

```
virtual void MouseDown (Point p)
```


Private Members

Point *_position*

namespace Vehicles

Enums

enum VehicleState

Values:

Undefined

Stopped

Moving

SpeedingUp

SlowingDown

Slowing4Junction

class Bus : public *Vehicle*

Public Functions

Bus (*VehicleManager* manager, *EntryPoint* entryPoint, int colorId)

Property

property TrafficJam::Vehicles::MaxSpeed

class Car : public *Vehicle*

Public Functions

Car (*VehicleManager* manager, *EntryPoint* entryPoint, int colorId)

Property

property TrafficJam::Vehicles::MaxSpeed

class RoadTrain

Public Functions

RoadTrain (*Vehicle* leader)

Vehicle **GetLeader** ()

void **HandOffLeadership** ()

void **NotifyFollowers** ()

Public Static Functions

static string **CreateId** (Vector2 *direction*, PointF *position*)

Private Members

Vehicle **_leader**

List<*Vehicle*> **_followers**

string **_id**

class Route

Public Functions

Route (List<Road> *roads*)

Road **GetRoad** (int *index*)

ReadOnlyCollection<Road> **GetRoads** ()

AnimationDirection **GetDirection** (int *index*)

ReadOnlyCollection<AnimationDirection> **GetDirections** ()

void **AddDirection** (AnimationDirection *direction*)

Private Members

readonly List<Road> **_roads**

readonly List<AnimationDirection> **_directions**

class Truck : public *Vehicle*

Public Functions

Truck (*VehicleManager* manager, *EntryPoint* entryPoint, int colorId)

Property

property TrafficJam::Vehicles::MaxSpeed

class Vehicle

Subclassed by *Bus*, *Car*, *Truck*

Public Functions

void **Update** (Graphics *graphics*)

void **JunctionEnqueued** (object *sender*, *VehicleEnqueuedEventArgs* *e*)

void **JunctionDequeued** (object *sender*, *VehicleEnqueuedEventArgs* *e*)

Property

property TrafficJam::Vehicles::MaxSpeed

property TrafficJam::Vehicles::Subscribed

property TrafficJam::Vehicles::Direction

property TrafficJam::Vehicles::Position

Protected Functions

Vehicle (*VehicleManager* manager, *EntryPoint* entryPoint, int colorId)

Private Functions

void **Move** ()

void **Draw** (Graphics *graphics*)

void **CalcCurrentSpeed** ()

float **CalcDeceleration** (Point *stoppingPoint*)

float **CalcPixelSpeed** (float *speed*)

Vector2 **GetDirection** ()

Road **GetRoad** (PointF *currentLocation*)

PointF **MoveLocation** (PointF *source*, PointF *target*, float *pixelSpeed*)

void **ShiftIndex** ()

void **ChangeState** (VehicleState *newState*)

```
void ResetJunctionData ()
```

```
Image SetTexture (TrafficTracer trafficTracer, int colorId)
```

Private Members

```
readonly VehicleManager _manager
```

```
readonly Route _route
```

```
readonly PointF [] _pathPoints
```

```
readonly Image _texture
```

```
VehicleState _state
```

```
float _speed
```

```
int _index
```

```
bool _finalDirection
```

```
bool _discard
```

```
Road _currentRoad
```

```
int _routeIndex
```

```
Junction _nextJunction
```

```
bool _nextJunctionHandled
```

```
float _deceleration
```

```
class VehicleFactory
```

Public Functions

```
VehicleFactory (TrafficTracer trafficTracer)
```

```
Vehicle Create (VehicleManager manager, EntryPoint entryPoint)
```

Private Functions

```
int GetVehicleCreationId ()
```

Private Members

```
readonly TrafficTracer _trafficTracer
```

```
readonly Random _random
```

```
const int BusPercentage = 20
```

```
class VehicleManager
```

Public Functions

VehicleManager (*SimulationController* controller)

void **AddVehicle** (*EntryPoint* entryPoint)

void **MoveVehicles** (Bitmap *nextFrame*)

void **Remove** (*Vehicle* vehicle)

Property

property TrafficJam::Vehicles::Controller

property TrafficJam::Vehicles::Full

Private Members

List<*Vehicle*> **_vehicles**

readonly SimulationController **_controller**

namespace Animation

Enums

enum AnimationDirection

Values:

Undefined = 0

TopIn = 1 << 0

TopOut = 1 << 1

RightIn = 1 << 2

RightOut = 1 << 3

BottomIn = 1 << 4

BottomOut = 1 << 5

LeftIn = 1 << 6

LeftOut = 1 << 7

TopRight = *TopIn* | *RightOut*

TopBottom = *TopIn* | *BottomOut*

TopLeft = *TopIn* | *LeftOut*

RightBottom = *RightIn* | *BottomOut*

RightLeft = *RightIn* | *LeftOut*

RightTop = *RightIn* | *TopOut*

BottomLeft = *BottomIn* | *LeftOut*

BottomTop = *BottomIn* | *TopOut*

BottomRight = *BottomIn* | *RightOut*

LeftTop = *LeftIn* | *TopOut*

LeftRight = *LeftIn* | *RightOut*

LeftBottom = *LeftIn* | *BottomOut*

class AnimationDirectionExtensions

Public Static Functions

static AnimationDirection MergeDirections(this AnimationDirection entranceDir

static bool IsHorizontal(this AnimationDirection direction)

static bool IsVertical(this AnimationDirection direction)

static bool IsCurve(this AnimationDirection direction)

class AnimationPathHelper

Public Functions

AnimationPathHelper (*Route route*)

GraphicsPath **GetAnimationPath** ()

Public Static Functions

static Point GetStartingPoint (Road *road*, AnimationDirection *direction*)

Private Functions

void **CalculateAnimationPath** (List<Road> *route*)

AnimationDirection **GetEntryPointDirection** (List<Road> *pathSection*)

AnimationDirection **GetExitPointDirection** (List<Road> *pathSection*)

AnimationDirection **GetDirection** (List<Road> *pathSection*)

AnimationDirection **GetEntranceSide** (Road *entryPoint*)

AnimationDirection **GetExitSide** (Road *exitPoint*)

AnimationDirection **GetExitDirection** (Road *road1*, Road *road2*)

void **AddAnimationPathPoints** (Road *road*, AnimationDirection *direction*)

void **AddHorizontalLine** (Road *road*, AnimationDirection *direction*)

void **AddVerticalLine** (Road *road*, AnimationDirection *direction*)

void **AddArc** (Road *road*, AnimationDirection *direction*)

Rectangle **CreateRectangle** (Point *source*, Point *destination*)

Private Members

```
const int TileLength = 150
const int SmallOffset = 50
const int LargeOffset = 100
readonly Route _route
readonly List<Road> _roads
readonly Road _exitPoint
readonly GraphicsPath _animationPath
```

Private Static Attributes

```
readonly IDictionary GetOppositeDirection= new Dictionary<AnimationDirection,
```


CONTRIBUTE

- Issue Tracker: <https://github.com/KnapSac/TrafficJam/issues>
- Source Code: <https://github.com/KnapSac/TrafficJam>

2.1 Indices and tables

- `genindex`
- `modindex`
- `search`

Symbols

_availableManager (C++ member), 19
 _backgroundWorker (C++ member), 19
 _continue (C++ member), 19
 _controller (C++ member), 20
 _currentRoad (C++ member), 24
 _deceleration (C++ member), 24
 _discard (C++ member), 24
 _entryPoint (C++ member), 6
 _finalDirection (C++ member), 24
 _followers (C++ member), 22
 _framesBeforeVehicle (C++ member), 5
 _grid (C++ member), 3
 _headerInformation (C++ member), 14
 _id (C++ member), 22
 _index (C++ member), 24
 _leader (C++ member), 22
 _nextFrame (C++ member), 3, 19
 _nextJunction (C++ member), 24
 _nextJunctionHandled (C++ member), 24
 _position (C++ member), 21
 _queueBottom (C++ member), 7
 _queueLeft (C++ member), 7
 _queueRight (C++ member), 7
 _queueTop (C++ member), 7
 _routeIndex (C++ member), 24
 _selectedButton (C++ member), 14
 _selectedTile (C++ member), 3
 _selectedToolButton (C++ member), 15
 _simulationStarted (C++ member), 12
 _situation1 (C++ member), 14
 _situation2 (C++ member), 14
 _situation3 (C++ member), 14
 _situationId (C++ member), 14
 _speed (C++ member), 24
 _state (C++ member), 24
 _tools (C++ member), 6
 _vehicleGeneratedFrameCount (C++ member), 5
 _vehicles (C++ member), 25

A

AddAnimationPathPoints (C++ function), 26
 AddArc (C++ function), 26
 AddDirection (C++ function), 22
 AddHorizontalLine (C++ function), 26
 AddVehicle (C++ function), 25
 AddVehicleManager (C++ function), 18
 AddVerticalLine (C++ function), 26
 AngleBetween (C++ function), 1
 AnimationPathHelper (C++ function), 26

B

BorderSizeMenuButtons (C++ function), 11
 BorderSizeToolMenu (C++ function), 15
 BorderSizeTopButtons (C++ function), 11
 BufferPanel (C++ function), 10
 Bus (C++ function), 21
 BusPercentage (C++ member), 24

C

CalcCurrentSpeed (C++ function), 23
 CalcDeceleration (C++ function), 23
 CalcPixelSpeed (C++ function), 23
 CalculateAnimationPath (C++ function), 26
 CalculateGrid (C++ function), 3
 CalculateRoute (C++ function), 2
 CanGenerateVehicle (C++ function), 5
 Car (C++ function), 21
 ChangeBorderSize (C++ function), 13
 ChangeButton (C++ function), 13, 16
 ChangeButtonMainWindow (C++ function), 11
 ChangeButtonMenu (C++ function), 11
 ChangeHeader (C++ function), 13
 ChangeMenuHeader (C++ function), 12
 ChangeState (C++ function), 23
 ControllerUnavailableException (C++ function), 10
 Create (C++ function), 24
 CreateEntryPoint (C++ function), 5
 CreateEntryPoints (C++ function), 3
 CreateId (C++ function), 22
 CreateMainWindow (C++ function), 2

CreateMap (C++ function), 2
CreateMapPreview (C++ function), 1
CreateNextFrame (C++ function), 18
CreatePanelBars (C++ function), 11
CreateRectangle (C++ function), 26
CreateSimulationController (C++ function), 2
CreateSimulationMonitor (C++ function), 2
CreateStartupWindow (C++ function), 2
CreateVehicleFactory (C++ function), 2
CrossRoad_Click (C++ function), 15

D

Dequeue (C++ function), 7
Dispose (C++ function), 20
DoWork (C++ function), 18
Draw (C++ function), 23
DrawNextFrame (C++ function), 2
DrawRoads (C++ function), 3

E

Enqueue (C++ function), 7
EntryPoint (C++ function), 5
EntryPointsMain_Click (C++ function), 12
EntryPointsPanel (C++ function), 10
EntryPointsSub (C++ function), 11

F

FindLeader (C++ function), 18

G

GenerateVehicles (C++ function), 18
GetAnimationPath (C++ function), 26
GetAvailableEntryPoints (C++ function), 2
GetDirection (C++ function), 22, 23, 26
GetDirections (C++ function), 22
GetEntranceSide (C++ function), 26
GetEntryPointDirection (C++ function), 26
GetExitDirection (C++ function), 26
GetExitPointDirection (C++ function), 26
GetExitSide (C++ function), 26
GetLeader (C++ function), 22
GetOrCreateRoadTrain (C++ function), 18
GetQueueByDirection (C++ function), 7
GetRoad (C++ function), 22, 23
GetRoads (C++ function), 22
GetStartingPoint (C++ function), 26
GetStoppingPoint (C++ function), 7
GetTile (C++ function), 2, 3
GetVehicleCreationId (C++ function), 24

H

HandOffLeadership (C++ function), 22
HorizontalTiles (C++ member), 3

I

InitializeComponent (C++ function), 12, 13
InitializeQueues (C++ function), 7
InitializeSimulation (C++ function), 1
InvalidDirectionException (C++ function), 10
IsEntryPoint (C++ function), 5

J

JunctionDequeued (C++ function), 23
JunctionEnqueued (C++ function), 23

L

LargeOffset (C++ member), 27

M

Main (C++ function), 1
MainWindow (C++ function), 11
Map (C++ function), 2
Map_Paint (C++ function), 3
MouseClicked (C++ function), 20
MouseDown (C++ function), 20
Move (C++ function), 23
MoveLocation (C++ function), 23
MoveSelectionBars (C++ function), 13
MoveVehicles (C++ function), 25

N

newSituation_Click (C++ function), 13
NotifyFollowers (C++ function), 22
NotifyObservers (C++ function), 20

O

OnCompleted (C++ function), 11, 18
OnError (C++ function), 11, 18
OnNext (C++ function), 11, 18
OnSimulationPaused (C++ function), 17, 18
OnSimulationReset (C++ function), 17, 18
OnSimulationStarted (C++ function), 17, 18
OnVehicleDequeued (C++ function), 7
OnVehicleEnqueued (C++ function), 7

P

PauseSimulation (C++ function), 19
PauseSimulation_Click (C++ function), 11

Q

QueuesEmpty (C++ function), 7

R

readonly (C++ function), 6
Remove (C++ function), 25
Reset (C++ function), 1

ResetJunctionData (C++ function), 23
 ResetSimulation (C++ function), 19
 ResetSimulation_Click (C++ function), 12
 ResetWindow (C++ function), 12
 RoadBlock_Click (C++ function), 15
 RoadTiles (C++ function), 17
 RoadTrain (C++ function), 22
 Route (C++ function), 22
 RouteToString (C++ function), 1
 RunWorkerCompleted (C++ function), 18

S

sameSituation_Click (C++ function), 13
 SelectMap (C++ function), 3
 SelectRandomDestination (C++ function), 3
 SelectRoadMap1 (C++ function), 3
 SelectRoadMap2 (C++ function), 3
 SelectRoadMap3 (C++ function), 3
 SetController (C++ function), 19
 SetTexture (C++ function), 6, 24
 SetupBackgroundWorker (C++ function), 18
 ShiftIndex (C++ function), 23
 ShowPanel (C++ function), 12
 ShowResetWindow (C++ function), 1
 SimulationController (C++ function), 18
 SimulationInfo (C++ function), 19
 SimulationMonitor (C++ function), 19
 SimulationResetEventArgs (C++ function), 9
 situation1_Click (C++ function), 13
 Situation1Info (C++ member), 14
 situation2_Click (C++ function), 13
 Situation2Info (C++ member), 14
 situation3_Click (C++ function), 13
 Situation3Info (C++ member), 14
 SmallOffset (C++ member), 27
 SpeedBump_Click (C++ function), 15
 SpeedSign_Click (C++ function), 15
 StartSimulation (C++ function), 18, 19
 StartSimulationMain_Click (C++ function), 11
 startSimulationStartup_Click (C++ function), 13
 StartupWindow (C++ function), 13
 Step (C++ function), 18
 StepSimulation_Click (C++ function), 11
 StoreEntryPoints (C++ function), 3
 Subscribe (C++ function), 19
 Subscribe<T> (C++ function), 17

T

throughputMeter_Click (C++ function), 15
 TileLength (C++ member), 27
 ToolsMain_Click (C++ function), 12
 ToolsPanel (C++ function), 15
 TrafficFlowMain_Click (C++ function), 12

TrafficFlowPanel (C++ function), 16
 TrafficJam (C++ type), 1
 TrafficJam::Environment (C++ type), 2
 TrafficJam::Environment::Map (C++ class), 2
 TrafficJam::Environment::Roads (C++ type), 3
 TrafficJam::Environment::Roads::CurvedRoad0 (C++ class), 3
 TrafficJam::Environment::Roads::CurvedRoad180 (C++ class), 4
 TrafficJam::Environment::Roads::CurvedRoad270 (C++ class), 4
 TrafficJam::Environment::Roads::CurvedRoad90 (C++ class), 4
 TrafficJam::Environment::Roads::EntryPoint (C++ class), 4
 TrafficJam::Environment::Roads::Grass (C++ class), 5
 TrafficJam::Environment::Roads::HorizontalRoad (C++ class), 5
 TrafficJam::Environment::Roads::Junctions (C++ type), 6
 TrafficJam::Environment::Roads::Junctions::Intersection (C++ class), 6
 TrafficJam::Environment::Roads::Junctions::Junction0 (C++ class), 7
 TrafficJam::Environment::Roads::Junctions::Junction1 (C++ class), 7
 TrafficJam::Environment::Roads::Junctions::Junction2 (C++ class), 8
 TrafficJam::Environment::Roads::Junctions::Junction3 (C++ class), 8
 TrafficJam::Environment::Roads::Junctions::Junction4 (C++ class), 8
 TrafficJam::Environment::Roads::Road (C++ class), 5
 TrafficJam::Environment::Roads::VerticalRoad (C++ class), 6
 TrafficJam::Events (C++ type), 8
 TrafficJam::Events::SimulationResetEventArgs (C++ class), 8
 TrafficJam::Events::VehicleEnqueuedEventArgs (C++ class), 9
 TrafficJam::Events::VehicleStateEventArgs (C++ class), 9
 TrafficJam::Exceptions (C++ type), 9
 TrafficJam::Exceptions::ControllerUnavailableException (C++ class), 9
 TrafficJam::Exceptions::InvalidDirectionException (C++ class), 10
 TrafficJam::Exceptions::VehicleCreationFailedException (C++ class), 10
 TrafficJam::Forms (C++ type), 10
 TrafficJam::Forms::BufferPanel (C++

class), 10
 TrafficJam::Forms::EntryPointsPanel (*C++ class*), 10
 TrafficJam::Forms::EntryPointsSub (*C++ class*), 10
 TrafficJam::Forms::MainWindow (*C++ class*), 11
 TrafficJam::Forms::ResetWindow (*C++ class*), 12
 TrafficJam::Forms::StartupWindow (*C++ class*), 13
 TrafficJam::Forms::ToolsPanel (*C++ class*), 15
 TrafficJam::Forms::TrafficFlowPanel (*C++ class*), 16
 TrafficJam::Helper (*C++ class*), 1
 TrafficJam::Program (*C++ class*), 1
 TrafficJam::Properties (*C++ type*), 16
 TrafficJam::Properties::RoadTiles (*C++ class*), 16
 TrafficJam::Simulation (*C++ type*), 17
 TrafficJam::Simulation::EventStore (*C++ class*), 17
 TrafficJam::Simulation::Paused (*C++ enumerator*), 17
 TrafficJam::Simulation::Resetting (*C++ enumerator*), 17
 TrafficJam::Simulation::Running (*C++ enumerator*), 17
 TrafficJam::Simulation::SimulationControl (*C++ class*), 17
 TrafficJam::Simulation::SimulationInfo (*C++ class*), 19
 TrafficJam::Simulation::SimulationMonitor (*C++ class*), 19
 TrafficJam::Simulation::SimulationMonitorSub (*C++ class*), 20
 TrafficJam::Simulation::SimulationState (*C++ enum*), 17
 TrafficJam::Simulation::Undefined (*C++ enumerator*), 17
 TrafficJam::Tools (*C++ type*), 20
 TrafficJam::Tools::IRoadTool (*C++ class*), 20
 TrafficJam::Tools::Tool (*C++ class*), 20
 TrafficJam::TrafficTracer (*C++ class*), 1
 TrafficJam::Vehicles (*C++ type*), 21
 TrafficJam::Vehicles::Animation (*C++ type*), 25
 TrafficJam::Vehicles::Animation::AnimationEffect (*C++ enum*), 25
 TrafficJam::Vehicles::Animation::AnimationDirection (*C++ class*), 26
 TrafficJam::Vehicles::Animation::AnimationPath (*C++ class*), 26
 TrafficJam::Vehicles::Animation::BottomIn (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::BottomLeft (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::BottomOut (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::BottomRight (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::BottomTop (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::LeftBottom (*C++ enumerator*), 26
 TrafficJam::Vehicles::Animation::LeftIn (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::LeftOut (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::LeftRight (*C++ enumerator*), 26
 TrafficJam::Vehicles::Animation::LeftTop (*C++ enumerator*), 26
 TrafficJam::Vehicles::Animation::RightBottom (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::RightIn (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::RightLeft (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::RightOut (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::RightTop (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::TopBottom (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::TopIn (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::TopLeft (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::TopOut (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::TopRight (*C++ enumerator*), 25
 TrafficJam::Vehicles::Animation::Undefined (*C++ enumerator*), 25
 TrafficJam::Vehicles::Bus (*C++ class*), 21
 TrafficJam::Vehicles::Car (*C++ class*), 21
 TrafficJam::Vehicles::Moving (*C++ enumerator*), 21
 TrafficJam::Vehicles::RoadTrain (*C++ class*), 21
 TrafficJam::Vehicles::Route (*C++ class*), 22
 TrafficJam::Vehicles::Slowing4Junction (*C++ enumerator*), 21
 TrafficJam::Vehicles::SlowingDown (*C++ enumerator*), 21

TrafficJam::Vehicles::SpeedingUp (C++
 enumerator), 21
 TrafficJam::Vehicles::Stopped (C++ *enu-*
 merator), 21
 TrafficJam::Vehicles::Truck (C++ *class*), 22
 TrafficJam::Vehicles::Undefined (C++
 enumerator), 21
 TrafficJam::Vehicles::Vehicle (C++ *class*),
 23
 TrafficJam::Vehicles::VehicleFactory
 (C++ *class*), 24
 TrafficJam::Vehicles::VehicleManager
 (C++ *class*), 24
 TrafficJam::Vehicles::VehicleState (C++
 enum), 21
 TrafficLight_Click (C++ *function*), 15
 TrafficTracer (C++ *function*), 1
 Truck (C++ *function*), 23

U

UnSubscribe<T> (C++ *function*), 17
 Unsubscriber (C++ *function*), 20
 Update (C++ *function*), 23
 UpdateEntryPointInflow (C++ *function*), 2
 UpdateVehicles (C++ *function*), 18
 UpdateWindow (C++ *function*), 13

V

Vehicle (C++ *function*), 23
 VehicleCreationFailedException (C++ *func-*
 tion), 10
 VehicleEnqueuedEventArgs (C++ *function*), 9
 VehicleFactory (C++ *function*), 24
 VehicleManager (C++ *function*), 25
 VehicleStateEventArgs (C++ *function*), 9
 VerticalTiles (C++ *member*), 3

X

X (C++ *member*), 6