

Лабораторная работа

Модель перекрестка при организации дорожного движения

Цель работы: научиться создавать модели дорожного трафика с помощью *Библиотеки дорожного движения*. Показать, как промоделировать движение автомобилей на перекрестке дорог Tapiolavagen и Menninkaisentie в районе Tapiola рядом с Хельсинки.


Краткие теоретические сведения

Будет создана модель, в которой за основу берется показанный ниже спутниковый снимок местности. На нем отчетливо видны все нюансы данного участка дорожной сети. Обе дороги - с двусторонним движением, и имеют по одной полосе для движения в каждом направлении. На обочине Tapiolavagen можно увидеть автобусную остановку, а на обочине Menninkaisentie - небольшую парковку на пять мест.

Последовательно учтем в будущей модели все эти нюансы, попутно продемонстрировав большинство элементов библиотеки дорожного движения.



Содержание работы:

Шаг 1. Добавление спутникового снимка местности, рисование дороги Tapiolavagen. Создание простейшей диаграммы процесса из блоков  Библиотеки дорожного движения **CarSource**, **CarMoveTo**, **CarDispose** для моделирования движения машин по Tapiolavagen на север.

Шаг 2. Добавление трехмерной анимации (добавление 3D окна и создание типа автомобиля Car с выбранной вами фигурой анимации, которая будет использоваться для отображения автомобилей).

Шаг 3. Рисование дороги Menninkaisentie, образующей перекресток с Tapiolavagen. Добавление новых блоков в диаграмму процесса для моделирования движения машин по обоим дорогам в обоих направлениях (дополнительные блоки **CarSource** и **CarMoveTo** для задания новых точек появления машин и новых целей движения машин, а также блоки Библиотеки моделирования процессов **SelectOutput** для разделения потоков машин по разным направлениям).

Шаг 4. Моделирование парковки. Изменение формы въездов с Menninkaisentie на Tapiolavagen. Рисование парковки на обочине дороги Menninkaisentie и добавление блоков в диаграмму процесса (очередные **CarSource** и **CarMoveTo**, а также блок Библиотеки моделирования процессов **Delay** для моделирования ожидания машин на парковке).

Шаг 5. Моделирование движения автобусов. Создание нового типа автомобиля Bus с заданной анимацией автобуса. Рисование автобусной остановки на обочине дороги Tapiolavagen и изменение диаграммы процесса для моделирования движения автобусов на север по Tapiolavagen и их заезда на автобусную остановку.


Шаг 6. Добавление светофоров для регулировки движения на перекрестке с помощью блока **Traffic Light**. Экспериментирование с различными режимами работы светофора для нахождения оптимальных длительностей красной и зеленой фазы светофора на перекрестке.

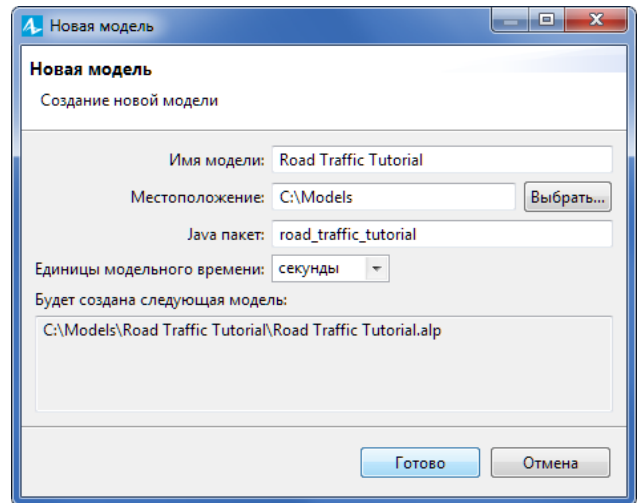
Порядок выполнения работы

Шаг 1. Создание дороги

Вначале мы создадим простейшую модель, которая будет моделировать движение автомобилей по Tapiolavagen в северном направлении.

1. Создайте новую модель

1. Щелкните мышью по кнопке панели инструментов **Создать** . Появится диалоговое окно **Новая модель**.
2. Задайте имя новой модели. В поле **Имя модели** введите *Road Traffic Tutorial*.
3. Выберите каталог, в котором будут сохранены файлы модели.
4. Выберите **секунды** в качестве **Единиц модельного времени**.
5. Щелкните мышью по кнопке **Готово**, чтобы завершить процесс.



Теперь мы можем настроить нашу модель. Стандартный сценарий создания модели:

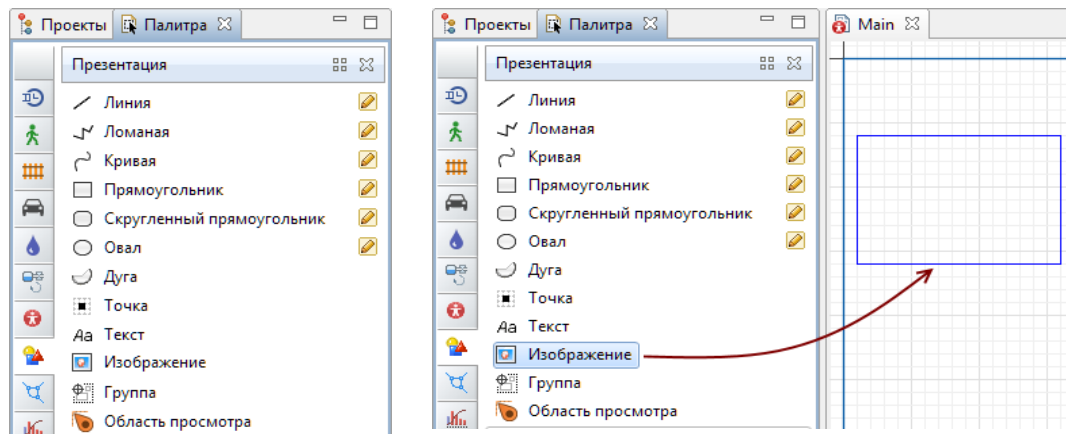
1. Добавьте спутниковый снимок местности, в которой находится интересующий нас перекресток.
2. Нарисуйте дороги поверх спутникового снимка.
3. Задайте движение автомобилей с помощью блоков *Библиотеки дорожного движения*.

Давайте создадим модель по вышеописанному сценарию.

1. Добавьте спутниковый снимок на графическую диаграмму модели

Место расположения файла со спутниковым снимком: Компьютер/Документы/ИВТ 151/Спутниковый снимок.jpg.

1. Сначала откройте палитру **Презентация** в панели **Палитра**.

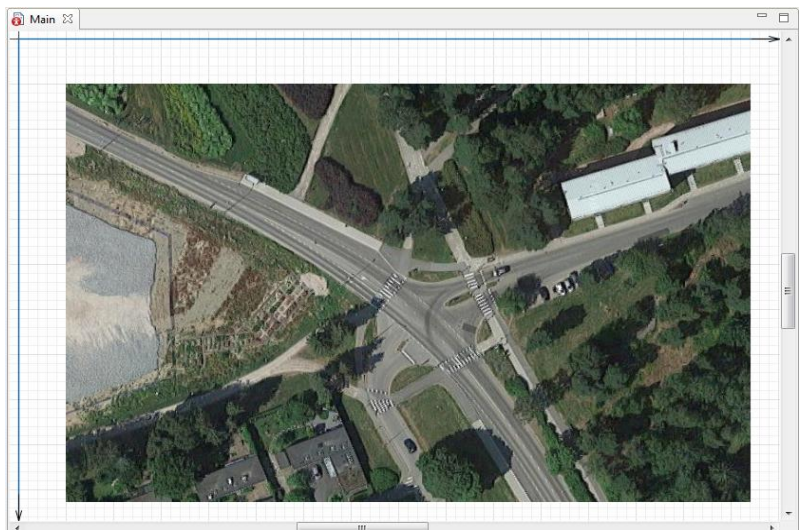


2. Перетащите элемент **Изображение** из палитры **Презентация** на графическую диаграмму *Main*.

4. Выберите изображение. Диалог для выбора файла появится автоматически. Откройте папку, в которую сохранится файл изображения, выберите его и нажмите **Открыть**. В графическом редакторе изображение будет выглядеть следующим образом:

1. Создайте дорогу

Нарисуем дорогу (Tapiolavagen), разместив ее строго поверх той, что на снимке. Дорожные сети в AnyLogic задаются следующим образом: рисуем




дороги, перекрестки и при необходимости добавляете другие элементы разметки пространства (автобусную остановку, стоп-линию и парковку).

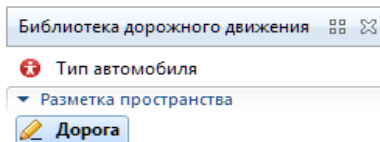
1. Прежде чем нарисовать дорогу, следует отключить сетку в графическом редакторе. Щелкните по кнопке панели инструментов **Включить/Отключить сетку**. Она должна принять вид не нажатой кнопки:



2. В панели **Палитра** выберите палитру **Библиотека дорожного движения**.

3. Сделайте двойной щелчок по элементу **Дорога** палитры **Библиотека дорожного движения**. Иконка элемента изменится на . Это означает, что активирован *режим рисования*, теперь можно рисовать дорогу в графическом редакторе.

4. Щелкните мышью в графическом редакторе, чтобы поставить первую точку дороги.



5. Продолжайте рисовать дорогу сегмент за сегментом. Наша дорога начинается с прямого сегмента. Чтобы нарисовать прямой сегмент, щелкните мышью в том месте, где хотите разместить конечную точку сегмента (см. рис. ниже).

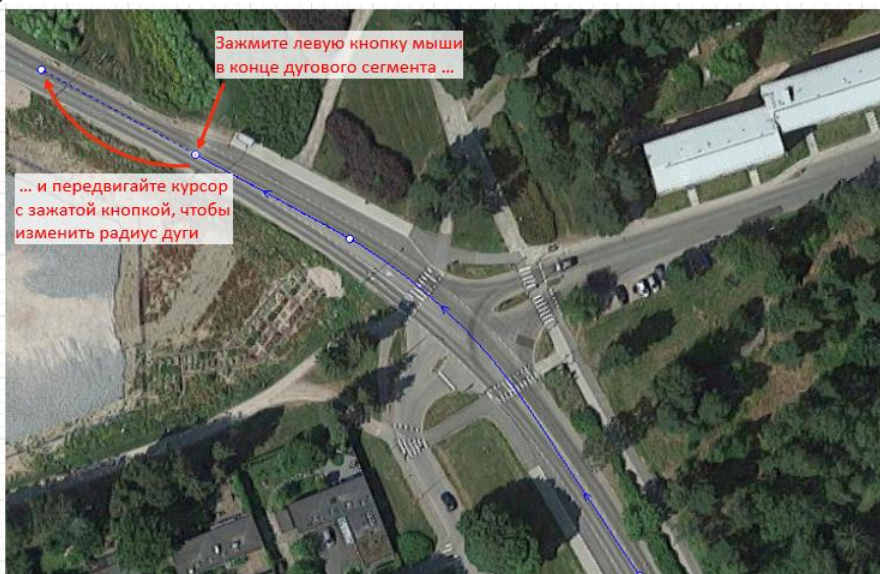


Чтобы начать рисовать дорогу, щелкните здесь

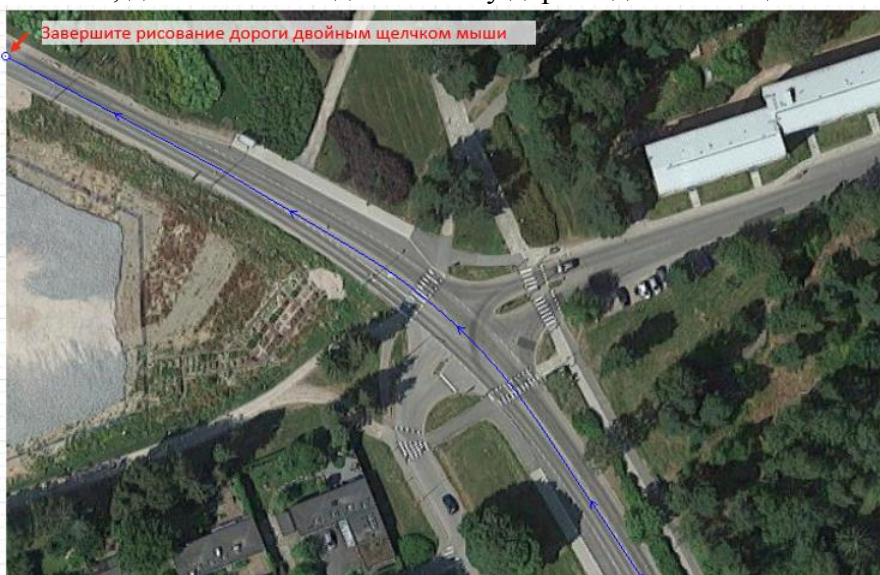


Щелкните, чтобы нарисовать прямой сегмент

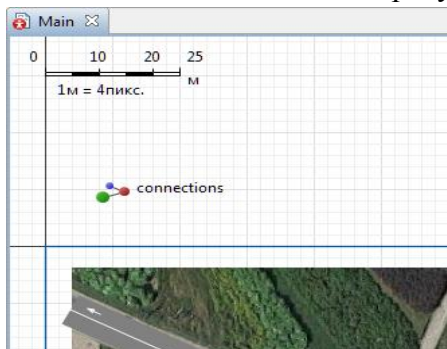
6. Следующий сегмент дороги имеет форму дуги. Чтобы нарисовать дуговой сегмент дороги, зажмите левую кнопку мыши в точке окончания дугового сегмента и начала следующего прямого сегмента и перемещайте курсор с нажатой левой кнопкой мыши до тех пор, пока сегмент не приобретет необходимую форму.



7. Чтобы завершить рисование, добавьте последнюю точку дороги двойным щелчком мыши.



8. Если это первая дорога в модели, появится сообщение с предложением сменить масштаб модели на: 4 пикселя в 1 метре. Оставляем предлагаемый масштаб. После этого увидим нарисованную дорогу. Она должна быть похожа на дорогу, которая изображена на рисунке ниже:

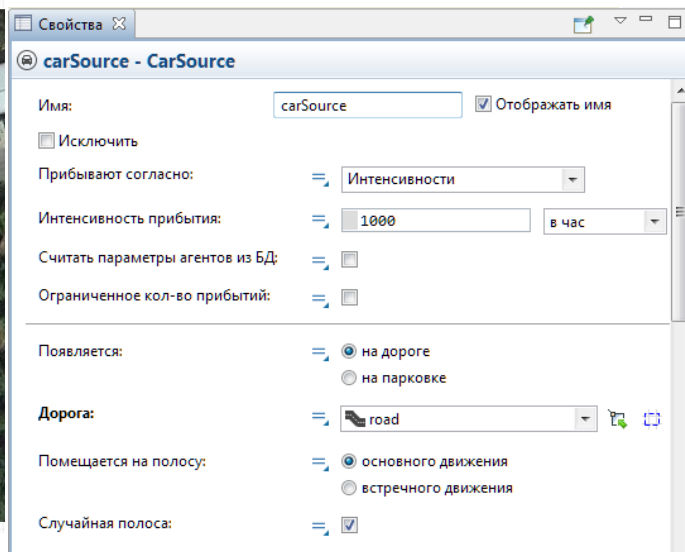
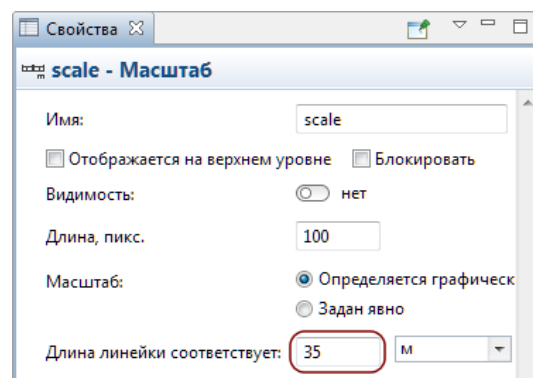
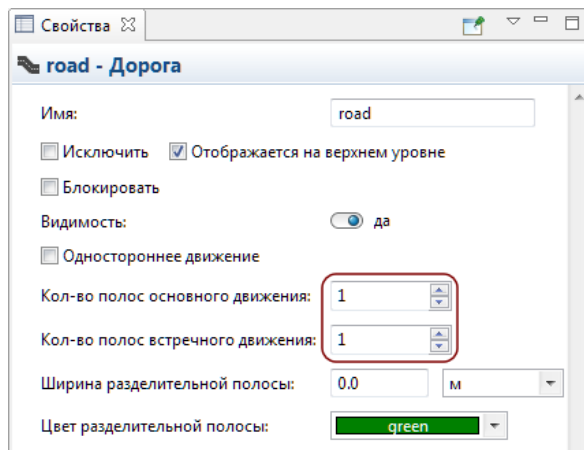


9. Настройте атрибуты дороги в панели **Свойства**. По умолчанию дороги создаются с двусторонним движением, при этом каждое направление дороги (основное и встречное) содержит по две полосы движения. Наша дорога (Tapiolavagen) содержит одну полосу основного движения и одну полосу встречного движения, поэтому необходимо задать

значение параметров **Кол-во полос основного движения** и **Кол-во полос основного движения** равным единице. Количество полос автоматически уменьшится до двух.

Сделаем дорогу полупрозрачной, чтобы можно было убедиться, что она точно накладывается на дорогу на карте. Настройки прозрачности (как и настройки направления движения, ширины полосы) находятся не в свойствах дороги, а в свойствах дорожной сети, которой принадлежит эта дорога.


10. Выделите дорожную сеть, сделав два последовательных щелчка по дороге (первый щелчок выделит саму дорогу, а уже второй - дорожную сеть). После этого откройте секцию свойств
11. Теперь можно увидеть, как созданная дорога накладывается на дорогу на карте. Если она шире или уже, нужно настроить масштаб модели. При необходимости можете перетащить холст немного ниже, чтобы над осью X появился элемент Масштаб, в свойствах которого можно будет указать масштаб.
12. В этом примере линейка масштаба соответствует 35 метрам.
13. Ширина дороги изменится в соответствии с новым значением масштаба модели. В итоге должна получиться дорога, подобная той, что на рисунке ниже:

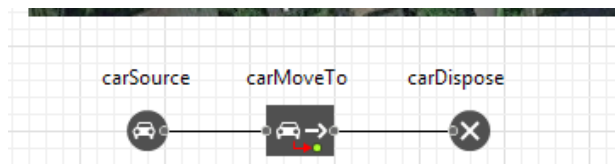


Создайте диаграмму процесса

Зададим процесс движения транспорта, создав диаграмму из блоков *Библиотеки дорожного движения*. Каждый блок задает определенную операцию, которая будет совершаться над проходящими по диаграмме процессами автомобилями.

Диаграмма процесса в AnyLogic создается путем добавления объектов библиотеки из палитры на графическую диаграмму, соединения их портов и изменения значений свойств блоков в соответствии с требованиями вашей модели.

1. Откройте палитру  **Библиотека дорожного движения** и добавьте блоки из этой палитры на диаграмму, соединив их, как показано на рисунке ниже:



- Когда блоки располагаются рядом друг с другом, между ними появляются соединительные линии. Будьте внимательны, эти линии должны соединять только порты, находящиеся с правой или левой стороны блоков. Скажем пару слов об этих блоках.

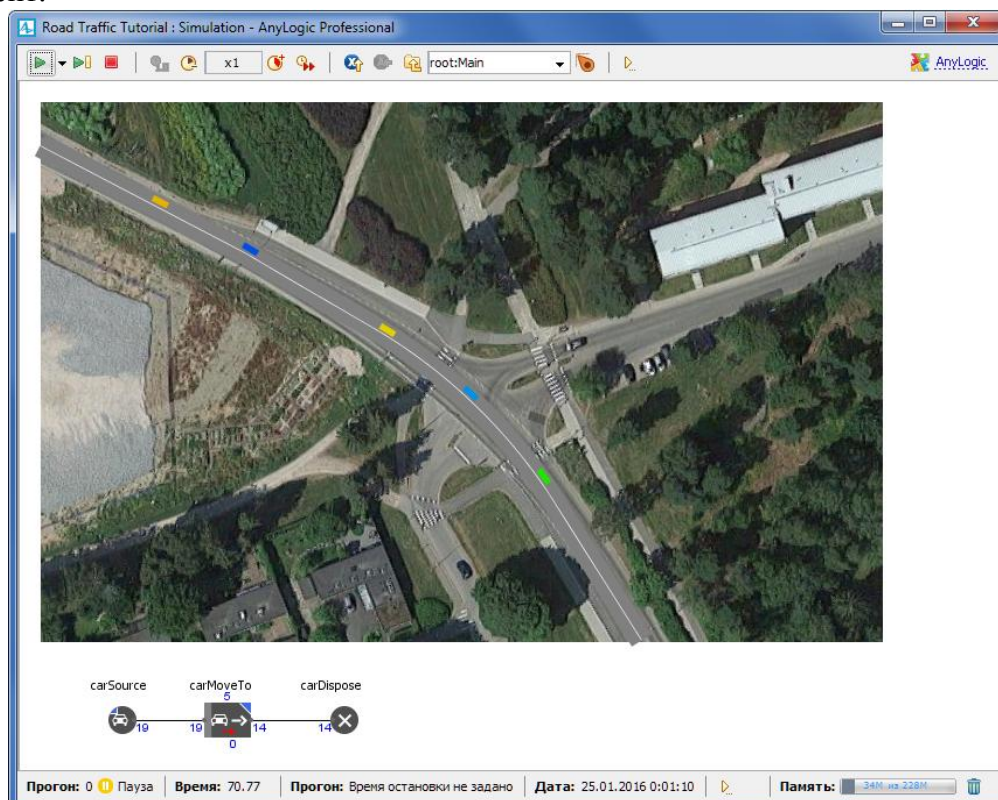
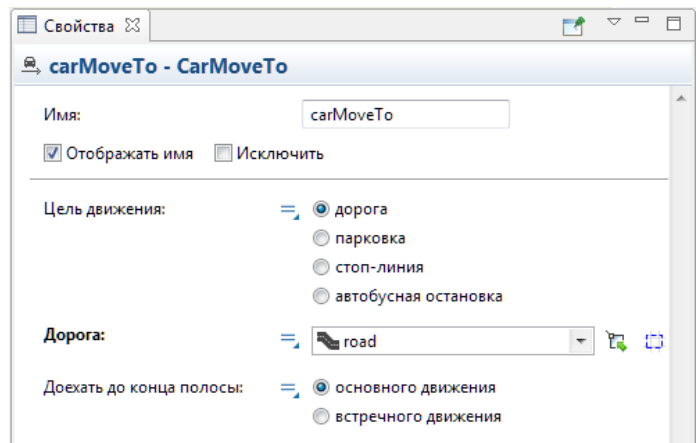
CarSource создает автомобили. Обычно используется как начальная точка автомобильной диаграммы процесса.

CarMoveTo моделирует движение автомобиля к цели движения.

CarDispose удаляет поступающие на вход этого блока автомобили из модели. Обычно используется как конечная точка автомобильной диаграммы процесса.

Настройте блоки диаграммы

- Выделите блок *carSource*. В раскрывающемся списке **Дорога** выберите *road*, это дорога, которую мы только что создали. Таким образом мы указываем блоку *carSource* дорогу, на которую стоит поместить создаваемые автомобили.
- Внесите изменения в свойства блока *carMoveTo*. Оставьте заданную по умолчанию опцию **Цель движения: дорога** и выберите имя нашей дороги *road* в раскрывающемся списке **Дорога**. Этот блок будет моделировать движение автомобилей по указанной дороге до конца полосы основного движения (задается параметром **Доехать до конца полосы**).
- В блоке *carDispose* оставьте настройки по умолчанию.
- Запустим модель, предварительно построив ее с помощью кнопки **Построить модель** . При необходимости, измените скорость выполнения модели, используя кнопки панели инструментов **Замедлить** и **Ускорить**. Для каждой модели, созданной с помощью объектов **Библиотеки дорожного движения**, автоматически создается блок-схема с наглядной визуализацией процесса. Она отображает количество автомобилей, которые прошли через блок или находятся в блоке в данный момент.

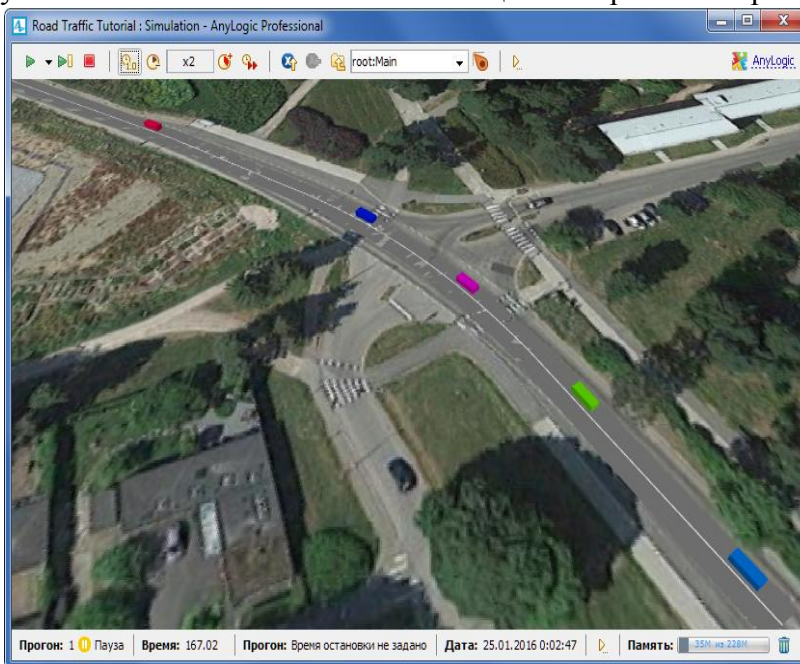
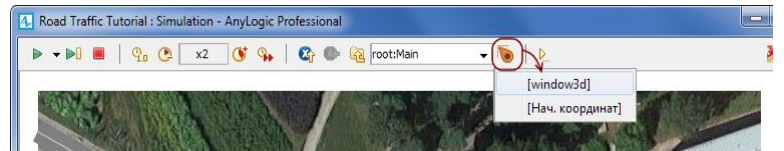
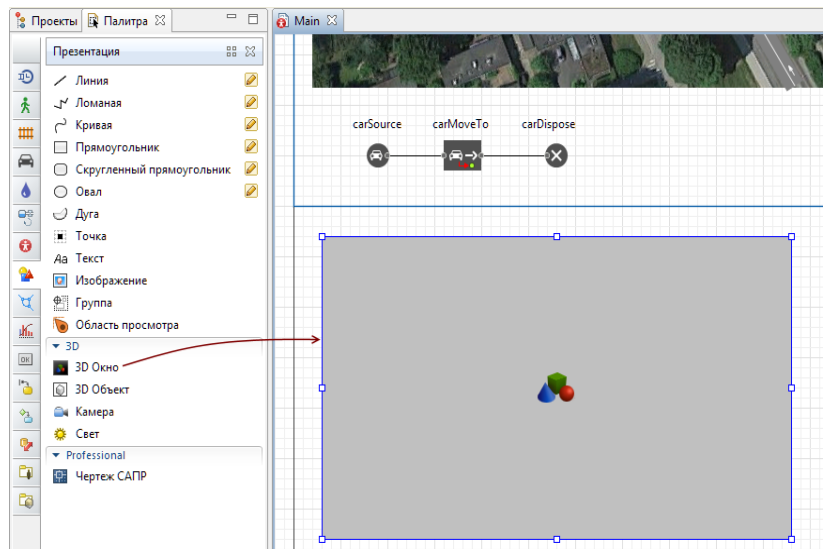


Шаг 2. Создание 3D анимации

Сначала необходимо добавить на графическую диаграмму **3D окно**. 3D окно используется для задания на диаграмме агента области, в которой во время запуска модели будет отображаться трехмерная анимация этой модели.

Добавьте 3D окно

1. Перетащите элемент **3D Окно** из секции **3D** палитры **Презентация** в графический редактор.
2. В графическом редакторе появится закрашенная серым цветом область. Поместите эту область туда, где вы хотите видеть 3D анимацию во время запуска модели. В нашей модели спутниковый снимок занимает большую часть пространства окна модели (прямоугольная рамка синего цвета в графическом редакторе обозначает границы окна модели), поэтому мы расположим 3D окно под снимком. Теперь можно запустить модель и наблюдать простую 3D анимацию.
3. Запустите модель.
4. Щелкните кнопку панели инструментов **Показать область...** и выберите **[window3D]**.
5. Вы увидите 3D анимацию. По умолчанию автомобили в 3D анимации отображаются разноцветными параллелепипедами.



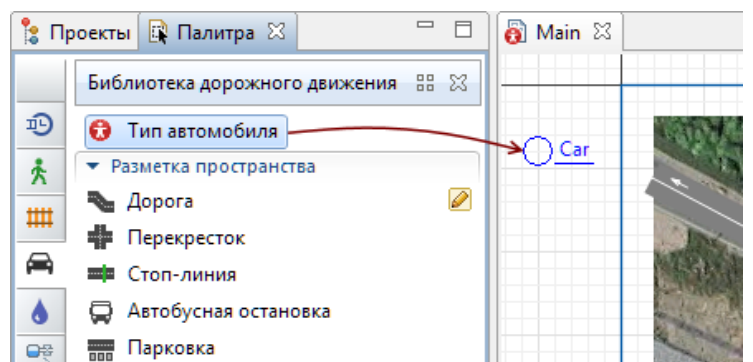
Чтобы	Выполните следующие действия
Переместить сцену	<ol style="list-style-type: none"> 1. Нажмите левую кнопку мыши в области 3D окна и держите ее нажатой. 2. Передвиньте мышшь в направлении перемещения.
Повернуть сцену	<ol style="list-style-type: none"> 1. Нажмите клавишу Alt и держите ее нажатой. 2. Нажмите левую кнопку мыши в области 3D окна и держите ее нажатой. 3. Передвиньте мышшь в направлении вращения.
Приблизить/отдалить сцену	<ol style="list-style-type: none"> 1. Покрутите колесо мыши от/на себя в области 3D окна.

4. Попробуйте перемещаться по трехмерной сцене с помощью описанных в таблице команд навигации.

Теперь необходимо задать автомобилям определенную фигуру анимации вместо используемого по умолчанию параллелепипеда. Это потребует создания нового типа автомобиля.

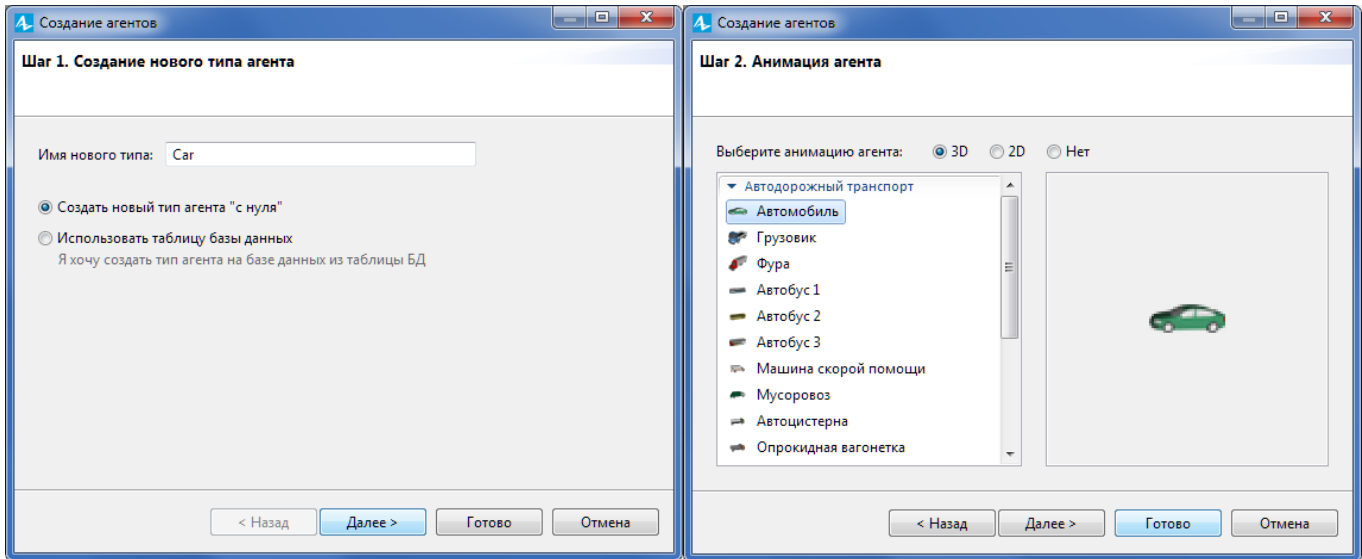
Создайте новый тип агента

1. Откройте палитру **Библиотека дорожного движения**.
2. Перетащите элемент **Тип автомобиля** в графический редактор.
3. Откроется диалоговое окно Мастера создания



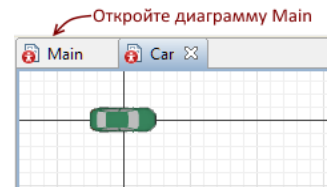
агентов на шаге **Создание нового типа агента**. Оставьте заданное по умолчанию имя нового типа **Car** в поле **Имя нового типа**. Нажмите **Далее**.

- Вы увидите следующую страницу Мастера создания типа агента, на которой предлагается выбрать фигуру анимации для этого агента. Поскольку мы хотим выбрать 3D фигуру, оставьте опцию **3D** выбранной. В списке ниже, выберите фигуру анимации **Автомобиль**.

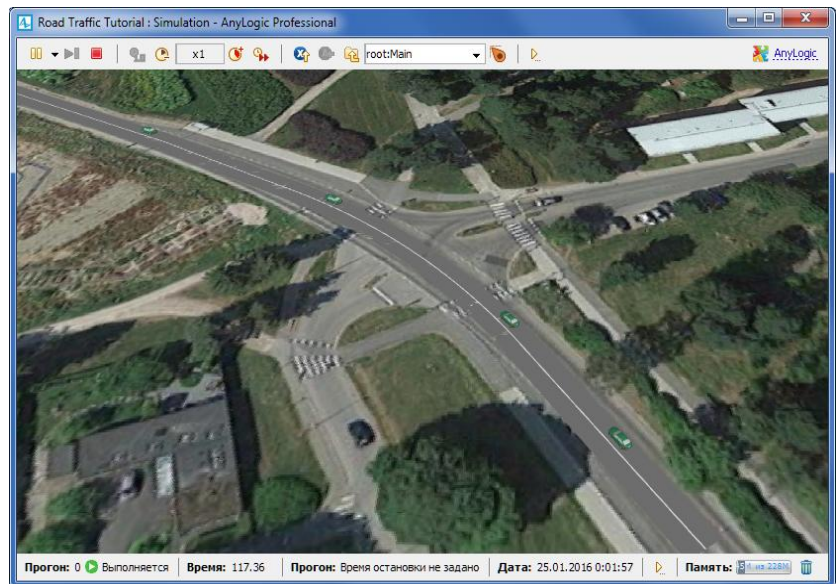
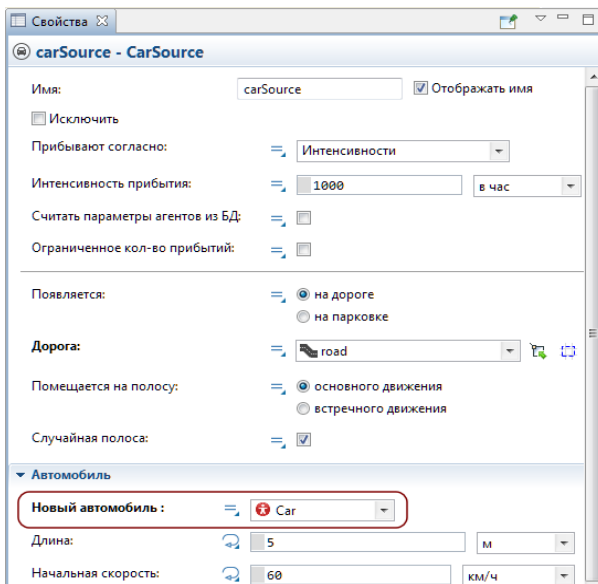


- Щелкните по кнопке **Готово**. Автоматически откроется диаграмма нового агента **Car**. В начале координат этой диаграммы вы увидите выбранную нами 3D фигуру автомобиля. Переключитесь обратно на диаграмму **Main**.

1/2 Настройте диаграмму процесса, чтобы использовать новый тип автомобиля



- На диаграмме **Main**, выделите блок **carSource** в графическом редакторе.
- Раскройте секцию **Автомобиль** в панели **Свойства** и выберите **Car** из раскрывающегося списка **Новый автомобиль**.



- Запустите модель и переключитесь в режим **3D**, чтобы увидеть 3D анимацию автомобилей, движущихся по дороге.

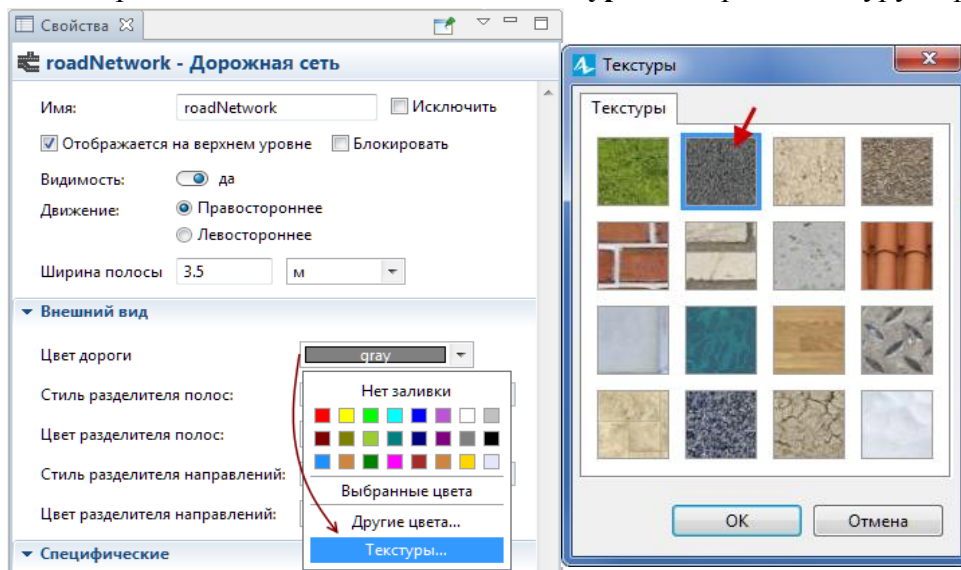
Иногда нарисованная вами дорога и снимок со спутника накладываются друг на друга, что приводит к искажению изображения. Кроме того, цвет нашей дороги очень похож на цвет дороги на спутниковом снимке, поэтому эти дороги сливаются.

Давайте внесем буквально пару изменений для того, чтобы сделать анимацию нашей модели более красивой.

1/2 Измените цвет дорожного покрытия нарисованной дорожной сети

- Выделите дорожную сеть, сделав два последовательных щелчка по дороге (первый щелчок выделит саму дорогу, а уже второй - дорожную сеть).

- Откройте секцию свойств дорожной сети **Внешний вид**. Щелкните по элементу управления **Цвет дороги** и выберите **Текстуры...** из раскрывшегося списка.
- В открывшемся диалоговом окне **Текстуры** выберите текстуру асфальта (см. рисунок ниже).



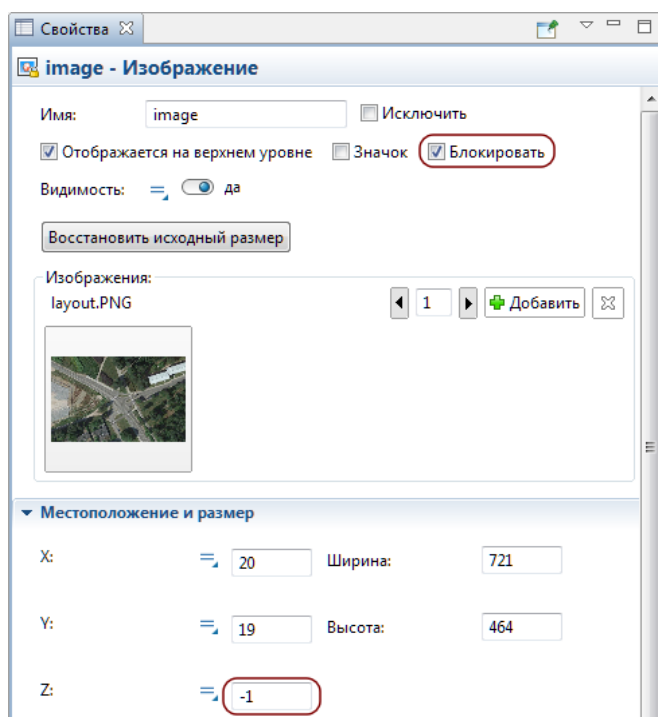
В графическом редакторе нарисованная дорога станет отличаться по цвету от дороги на спутниковом снимке:



Теперь надо решить проблему с чересполосицей в трехмерной анимации. Причина проблемы в том, что и дорожная сеть, и спутниковый снимок отображаются на одном Z-уровне (0), и поэтому в какие-то моменты на верхний уровень вылезает рисунок, а в какие-то - дорога.

1. 2. Измените Z-уровень рисунка

- В графическом редакторе, щелкните по спутниковому снимку моделируемой местности.
- Откройте секцию свойств изображения **Местоположение и размер** и введите **-1** в поле **Z**. Тем самым, мы поместим рисунок в трехмерной анимации чуть ниже, чем нашу дорожную сеть.
- Выберите в свойствах изображения флажок **Блокировать**. Теперь изображение будет считаться заблокированным и не будет реагировать на щелчки мыши. С помощью блокировки можно избежать случайного редактирования изображения во время рисования элементов поверх него.



На следующем шаге нарисуем еще одну дорогу и промоделируем движение автомобилей по образовавшемуся перекрестку.


Шаг 3. Моделирование перекрестка

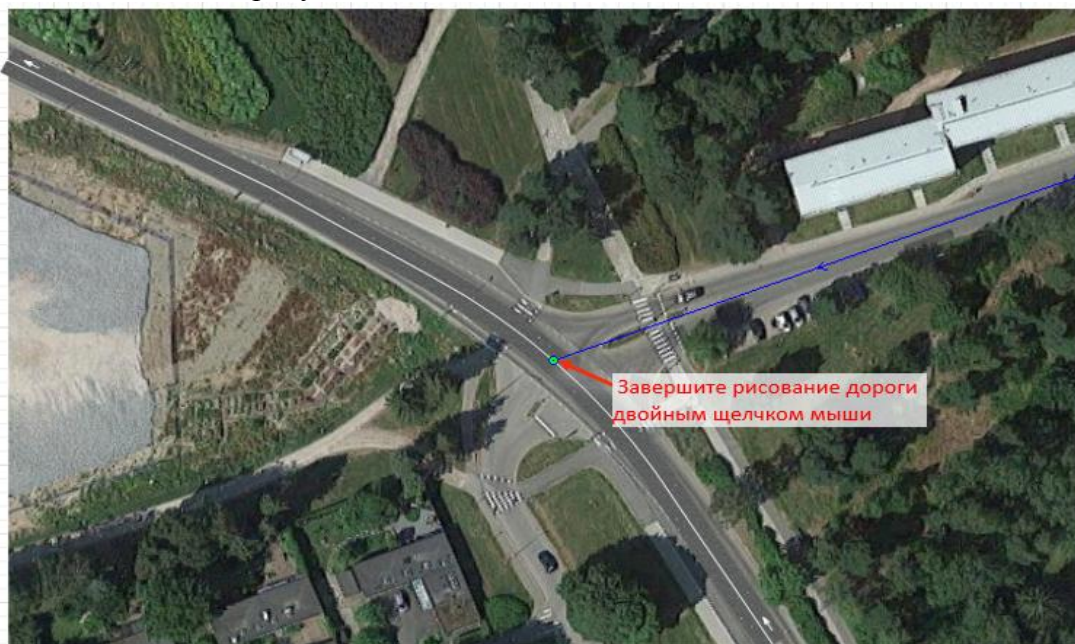
Сейчас машины движутся в модели по дороге Tapiolavagen, причем только в одном направлении, с юга на север.

На этом шаге создадим перекресток, нарисовав улицу Menninkaisentie, примыкающую к улице Tapiolavagen с востока. После этого добавим новые блоки в диаграмму процесса, которые будут моделировать движение машин по обеим дорогам во всех направлениях.

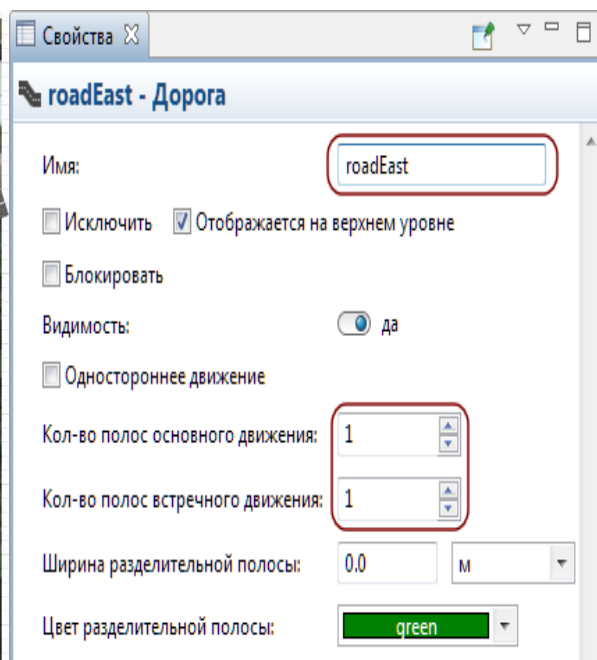
Перекрестки в AnyLogic создаются автоматически при соединении дорог. Нарисуйте улицу Menninkaisentie, примыкающую к Tapiolavagen с востока.



1. Начните рисование дороги с двойного щелчка по элементу  **Дорога** в палитре **Библиотека дорожного движения**.
2. Добавьте начальную точку дороги, щелкнув по разделительной полосе Menninkaisentie в правой части рисунка.
3. Добавьте конечную точку дороги. Чтобы правильно соединить дороги в Т-образный перекресток, сделайте двойной щелчок мыши, когда увидите зеленую точку на разделительной полосе улицы Tapiolavagen, как показано на рисунке ниже.



4. Будет создан трехсторонний перекресток.
5. У нарисованной дороги по две полосы в каждом направлении, у реальной же улицы Menninkaisentie одна полоса основного движения и одна полоса встречного движения, поэтому нужно изменить настройки этой дороги.
6. Откройте свойства только что нарисованной дороги, щелкнув по ней мышью, и сделайте **Кол-во полос основного движения** и **Кол-во полос встречного движения** этой дороги равным 1.
7. Чтобы в дальнейшем нам было легче ссылаться на эту дорогу в параметрах блоков диаграммы процесса, измените **Имя** этой дороги на **roadEast**.



7. После внесенных изменений дорожная сеть должна будет выглядеть следующим образом:

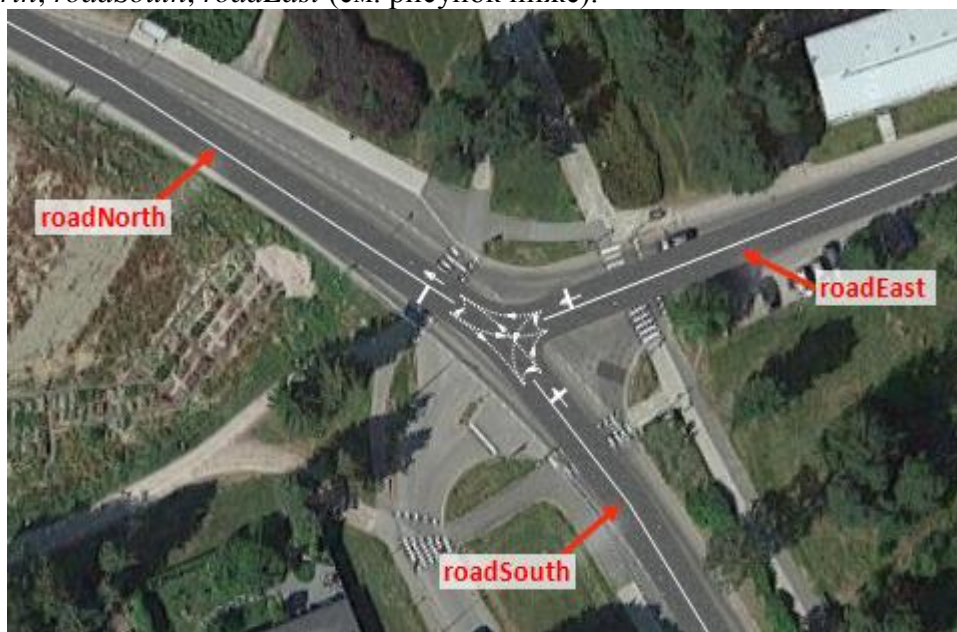


Если вы сейчас запустите модель, то увидите, что теперь машины едут не до конца улицы Tärölavägen, а лишь доезжают до перекрестка и исчезают. Это вызвано тем, что при присоединении новой дороги к уже существующей дороге создается новый элемент разметки пространства AnyLogic - [перекресток](#), который делит существующую дорогу на две отдельные дороги.

В нашей диаграмме процесса блок *carSource* создает машины в начале дороги *road*, после чего блок *carMoveTo* моделирует движение машины до конца той же самой дороги, которая теперь заканчивается у перекрестка (после перекрестка следует уже другая дорога, названная *road2*).


Переименуем обе эти дороги, чтобы в дальнейшем нам было проще ссылаться на них в блоках диаграммы процесса.

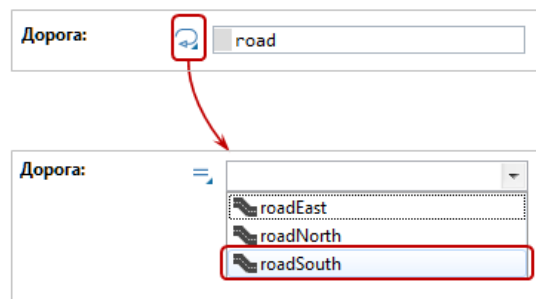
1. Поочередно выделите дороги и измените их имена в свойствах. Пусть дороги теперь называются *roadNorth*, *roadSouth*, *roadEast* (см. рисунок ниже).



Поскольку блоки диаграммы процесса *carSource* и *carMoveTo* все еще ссылаются на старое название дороги, нужно внести соответствующие изменения и в параметры этих блоков.

1. Выделите блок *carSource*. Измените его **Имя** на *carSourceS* (чтобы подчеркнуть, что этот блок генерирует машины, появляющиеся с юга).

2. В поле **Дорога** этого блока вы увидите старое название дороги. Чтобы заменить его на новое, лучше переключиться из режима задания динамического значения в статический режим выбора элемента, щелкнув по кнопке  слева от поля **Дорога**. Затем выберите имя дороги *roadSouth* из раскрывающегося списка (см. рисунок справа).




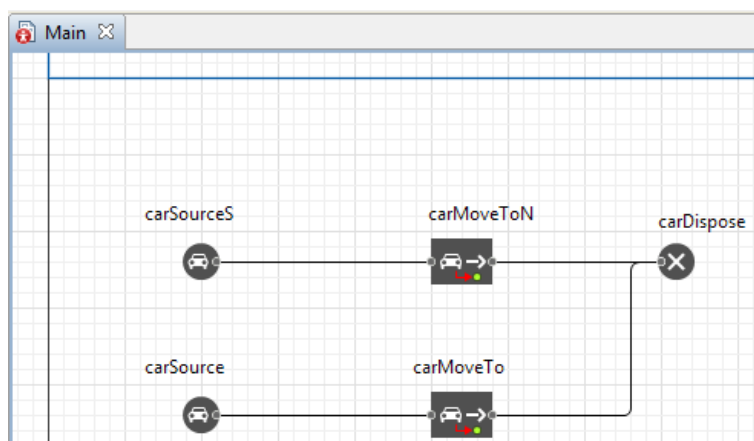
3. Аналогичным образом измените свойства блока *carMoveTo*. Назовите его *carMoveToN* (он будет моделировать движение машин до конца уходящей на север дороги) и выберите в поле **Дорога** *roadNorth*. Если сейчас запустить модель, то видно, что машины снова движутся по дороге Tapiolavagen с юга на север.


Следующим шагом будет добавление новых блоков в диаграмму процесса, чтобы машины начали ездить по всем дорогам, причем в каждом разрешенном направлении.

Добавьте новые блоки в диаграмму процесса

1. Вначале переместим существующую диаграмму процесса ниже, под синюю рамку окна модели, поскольку наша диаграмма процесса станет больше, и уже не поместится в ту область под спутниковым снимком, которая отображается в окне модели при ее запуске.

2. Теперь добавим блоки, моделирующие движение по той же дороге, но с севера на юг (по полосе встречного движения улицы Tapiolavagen). Добавьте еще один блок 



CarSource и блок  **CarMoveTo**. Соедините их, как показано на рисунке справа.

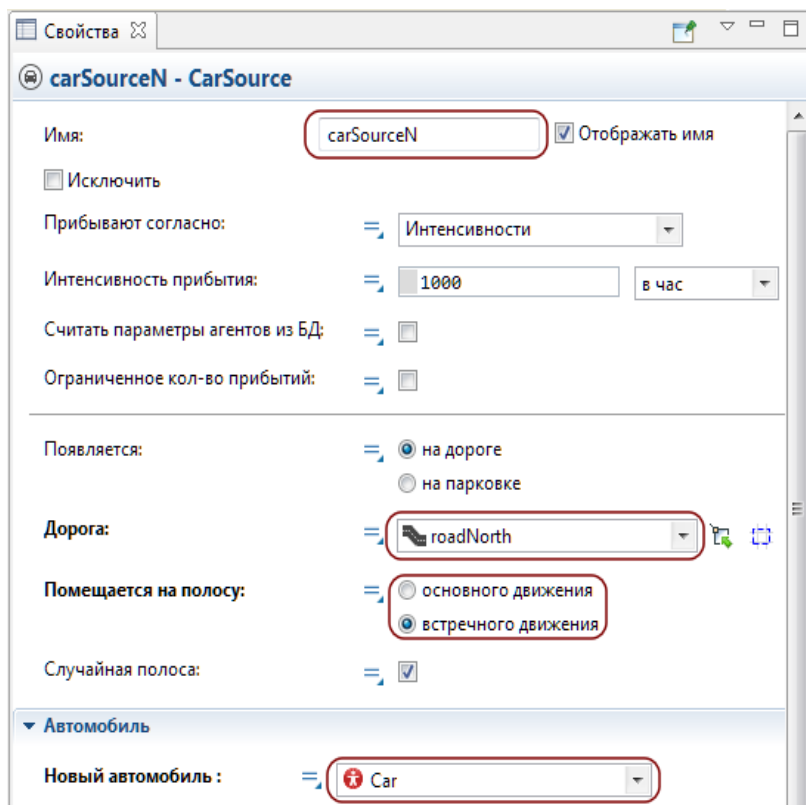
Как видно из рисунка, соединители блоков имеют сложную форму. Чтобы соединить порты двух блоков соединителем нестандартной формы, нужно начать рисование с двойного щелчка мышью по первому соединяемому порту, затем (если нужно) добавить точки изгиба соединителя, щелкнув в соответствующих местах холста, и завершить рисование, сделав щелчок внутри второго соединяемого порта.

1. Переименуйте блок *carSource* в *carSourceN* (этот блок генерирует машины, появляющиеся с севера).

2. В поле **Дорога**, выберите дорогу, на которой будут появляться создаваемые этим блоком автомобили: *roadNorth*.

3. Поскольку нам нужно, чтобы автомобили появлялись в начале полосы встречного движения этой дороги (у ее северного конца), выберите в параметре **Помещается на полосу:** **встречного движения**.

4. Разверните секцию свойств **Автомобиль** и выберите в свойстве **Новый автомобиль** *Car*. Теперь блок *carSourceN* будет генерировать машины созданного нами типа *Car*, и они будут отображаться в трехмерной анимации фигурками зеленых автомобилей.

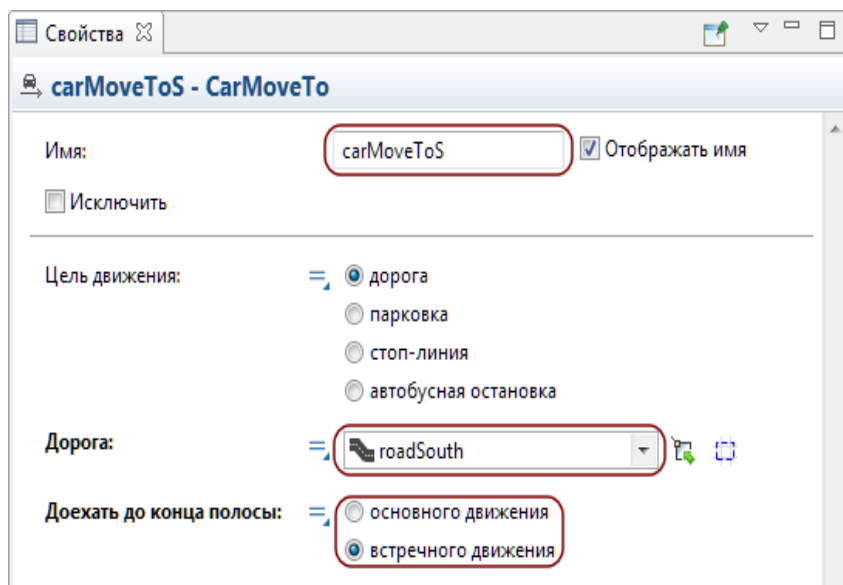


- Аналогичным образом измените свойства блока *carMoveTo*. Назовите его *carMoveToS* (он будет моделировать движение машин на юг), выберите в поле **Дорога** *roadSouth*, а также выберите опцию **Доехать до конца полосы:**

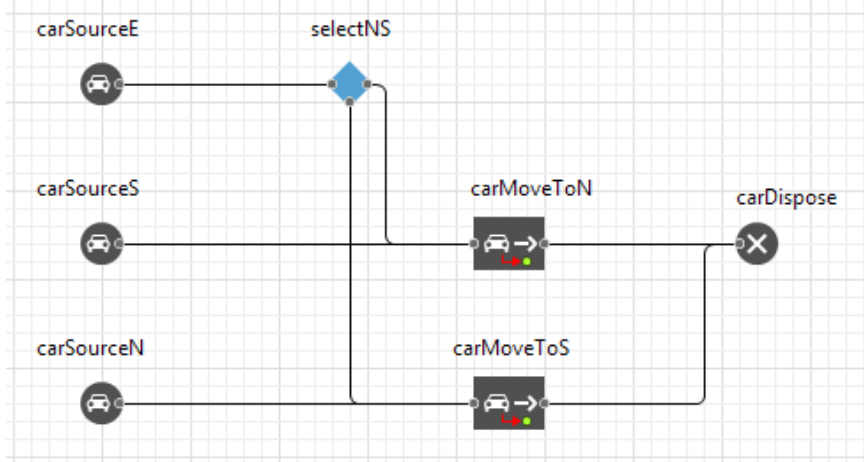
встречного движения.

Запустите модель, видно, что теперь машины движутся по Tapiolavagen в обоих направлениях.

Промоделируем движение машин, едущих по Menninkaisentie с востока. На перекрестке часть этих машин (примерно половина) будет следовать на юг, а оставшаяся часть - на север.



- Добавьте еще один блок **CarSource** (он будет генерировать машины у начала дороги Menninkaisentie).
- Назовите этот блок *carSourceE*, а в поле **Дорога** выберите *roadEast*. Разверните секцию свойств **Автомобиль** и выберите в свойстве **Новый автомобиль** *Car*.
- Теперь нужно добавить блок, который будет направлять часть машин по дороге Tapiolavagen на север, а часть – на юг. Для этого понадобится блок **SelectOutput** Библиотеки моделирования процессов. Переключитесь на палитру **Библиотека моделирования процессов** и перетащите блок **SelectOutput** на диаграмму так, чтобы он соединился с только что созданным блоком *carSourceE*.
- Назовите этот блок *selectNS*.
- Два выходных порта блока **SelectOutput** соедините с входными портами блоков *carMoveToN* и *carMoveToS*. Диаграмма процесса должна в итоге выглядеть следующим образом:




Блок **SelectOutput** перенаправляет поступающих на его вход агентов (в нашем случае - автомобили) на один из двух его выходов согласно заданным вероятностям или в зависимости от выполнения заданного условия. По умолчанию блок работает согласно вероятностям, направляя в верхний порт примерно 50 процентов агентов, что нас устраивает, поэтому менять свойства этого блока нет смысла.

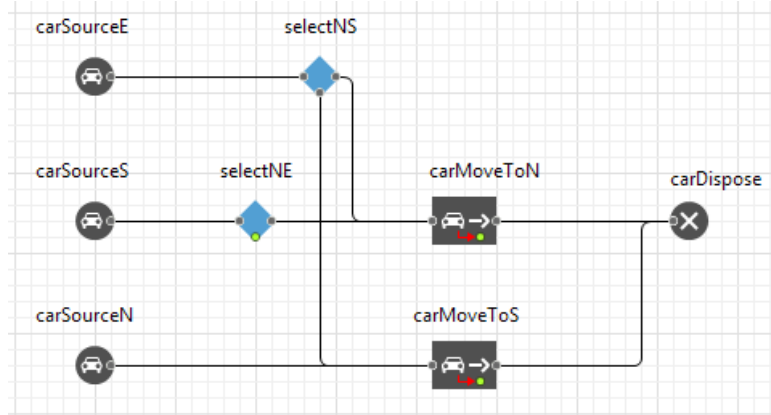
Чаще всего диаграммы процесса моделей дорожного движения состояются из блоков **Библиотеки дорожного движения** **CarSource**, **CarMoveTo** и **CarDispose**, а также блоков **Библиотеки моделирования процессов** **SelectOutput** или **SelectOutput5** (для маршрутизации машин по разным направлениям) и **Delay** (для моделирования пребывания машин на парковках и автобусов на остановках, мы рассмотрим эти случаи в данном учебном пособии позже).

Если вы запустите модель, то увидите, что теперь машины движутся в обоих направлениях по Tapiolavagen, а также приезжают с востока по Menninkaisentie, после чего направляются либо на юг, либо на север. Единственное, что осталось неучтенным в текущей модели - это поворот машин с Tapiolavagen на Menninkaisentie.


Давайте добавим еще несколько блоков в диаграмму процесса, которые позволят нам промоделировать и этот последний неучтенный сценарий движения машин.

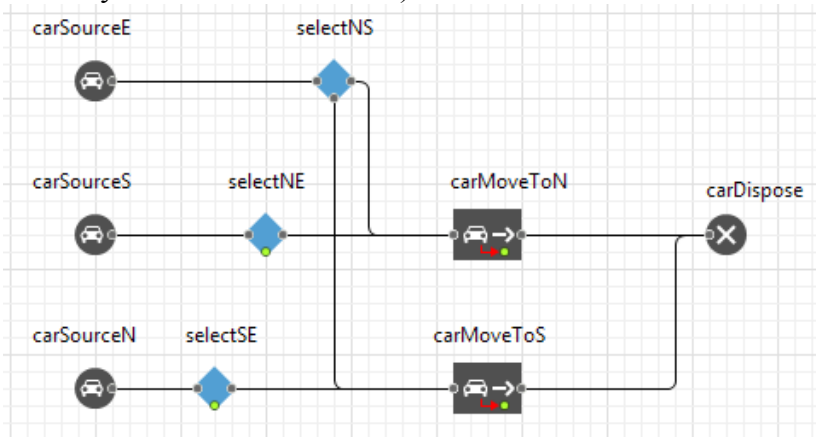
Измените диаграмму процесса


1. Добавьте еще один блок Библиотеки моделирования процессов  **SelectOutput**. Поместите его между блоками *carSourceS* и *carMoveToN* (так, чтобы он автоматически соединился с обоими этими блоками, см. рисунок ниже). Этот блок нужен нам, чтобы распределить поток автомобилей с юга между восточным и северным направлениями. Назовите этот блок *selectNE*.

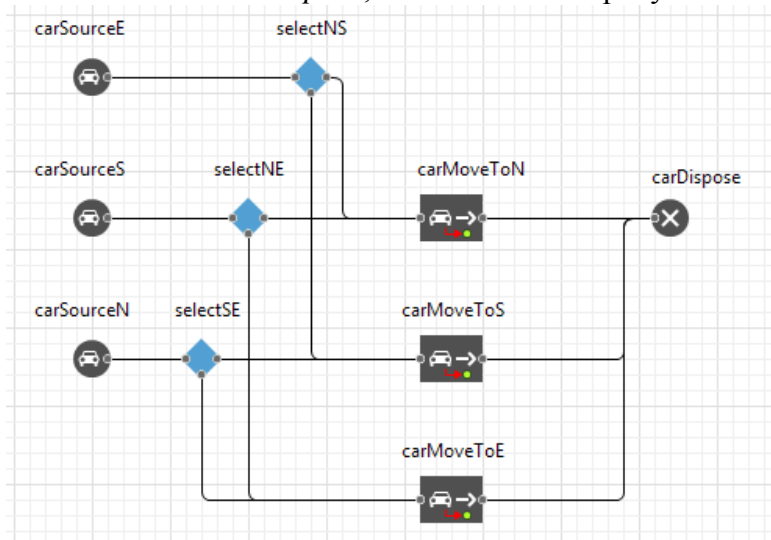


2. Откройте свойства этого блока. Настройте маршрутизацию автомобилей. Предположим, что большинство автомобилей, едущих по главной дороге (примерно 70%) будут продолжать ехать прямо по ней. В этом случае нам нужно изменить долю автомобилей, которые будут перенаправляться в первый выходной порт этого блока **SelectOutput**. В свойстве блока **Вероятность** введите 0.7. Теперь в среднем 70 процентов агентов будут направляться в верхний порт, а 30 - в нижний (мы соединим нижний порт с соответствующим блоком позже).

3. Добавьте еще один блок  **SelectOutput**. Назовите этот блок *selectSE*. Поместите его между блоками *carSourceN* и *carMoveToS* (см. рисунок ниже). Этот блок нужен для выбора пути автомобилей, едущих с севера. Аналогично измените долю автомобилей, едущих на перекрестке прямо, на 70 процентов, введя в поле **Вероятность** 0.7



4. Теперь добавьте еще один блок Библиотеки дорожного движения  **CarMoveTo**. Назовите его *carMoveToE* (он будет моделировать движение машин по Menninkaisentie на восток). Соедините входной порт с нижними портами двух (!) блоков, *selectNE* и *selectSE*. Выходной же порт соедините с блоком *carDispose*, как показано на рисунке ниже.




5. Откройте свойства этого блока. Выберите в поле **Дорога** *roadEast*, а также выберите опцию **Доехать до конца полосы: встречного движения**.
6. Запустите модель и пронаблюдайте за тем, как машины движутся по дорогам.

Шаг 4. Добавление парковки

На этом шаге добавим парковку на обочине Menninkaisentie (парковка хорошо видна на спутниковом снимке).

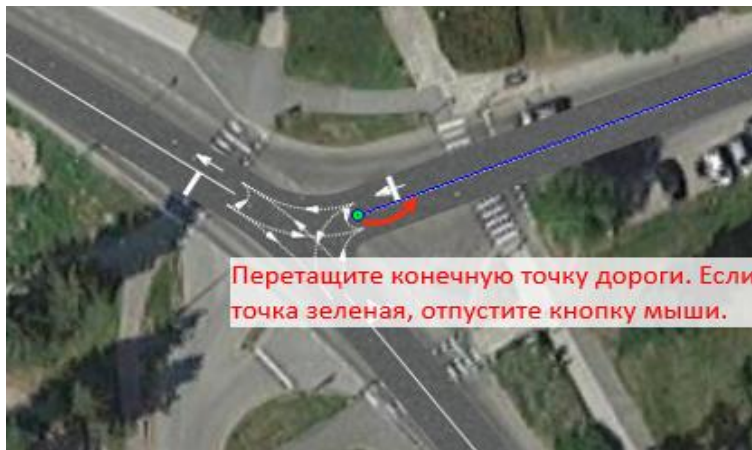
Улучшим текущую модель. Как видно, въезды на главную дорогу с Menninkaisentie не совсем точно соответствуют реальным дорогам на местности. Изменим перекресток в модели, чтобы добиться более точного соответствия.

1. Чтобы удобнее редактировать графические элементы перекрестка, увеличьте масштаб графической диаграммы .

2. Чтобы увеличить форму и длину въездов на перекрестке, нужно увеличить площадь самого перекрестка. Это можно сделать, отделив конечные точки примыкающих к перекрестку дорог. Для этого выделите в графическом редакторе дорогу *roadEast*. Вы увидите, что конечная точка дороги у перекрестка выделена зеленым цветом. Аккуратно перетащите примыкающую к перекрестку точку в сторону от перекрестка. При этом отпустить кнопку мыши нужно только в том случае, если точка все еще подсвечивается зеленым цветом - это означает наличие соединения с перекрестком. Если же точка станет белой, то это значит, что вы перетащили конец дороги слишком далеко от перекрестка. В таком случае отмените последнее действие и затем повторите действие.

3. Если необходимо, повторите перетаскивание несколько раз. В итоге, перемещая конечную точку дороги по чуть-чуть, нужно, чтобы точка оказалась примерно на том же расстоянии, которое показано на следующем рисунке (см. рисунок справа).


4. Аналогичным образом перетащите и конечные точки других двух дорог. В результате въезды на главную дорогу должны будут принять ту же форму, что и дороги на спутниковом снимке.

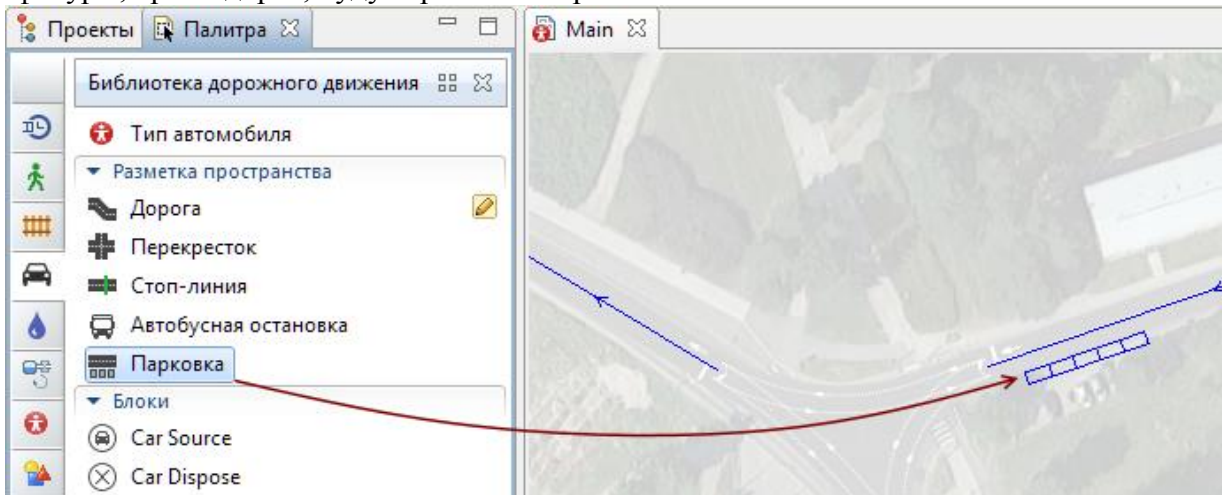


Запустите модель. Теперь движение машин в нашей модели будет абсолютно таким же, как и на моделируемом нами перекрестке.

Промоделировать, как время от времени машины заезжают на парковку.

Нарисуйте парковку

1. Перетащите элемент **Парковка**  из секции **Разметка пространства** палитры **Библиотека дорожного движения** на графическую диаграмму. Перетаскивая парковку по диаграмме, видно, что все фигуры, кроме дорог, будут временно скрыты.




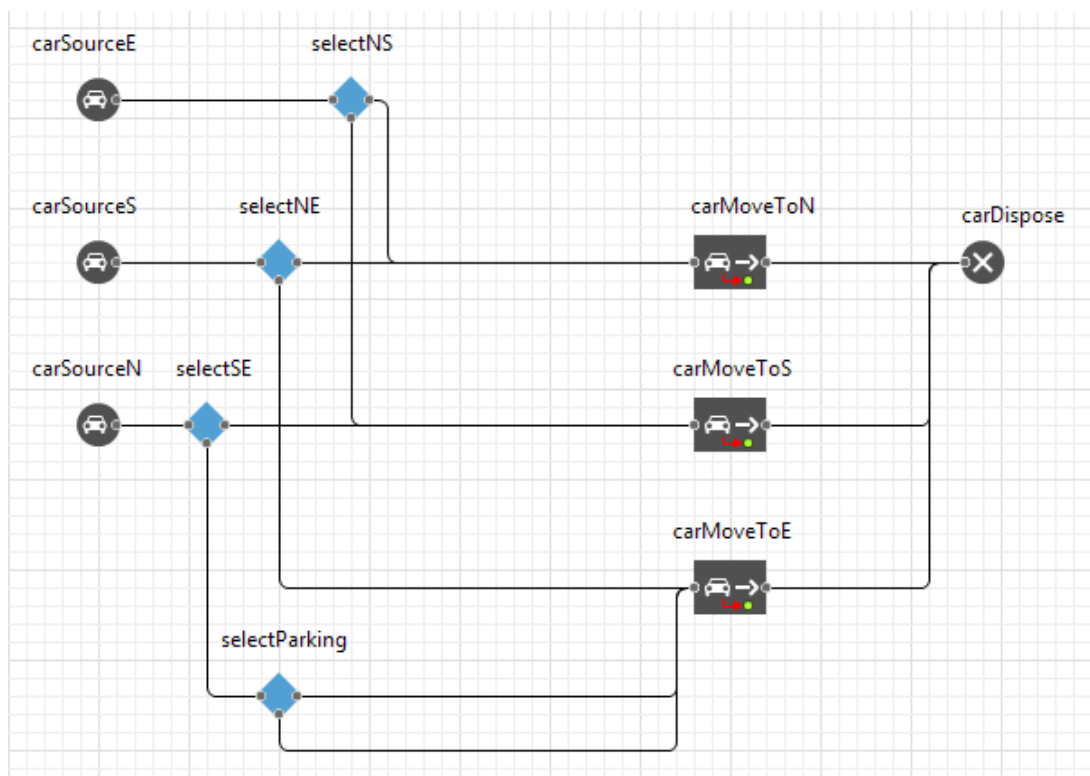
2. Поместите курсор у дороги в том месте улицы Menninkaisentie, где на спутниковом снимке видны припаркованные машины. Парковка прикрепится к дороге.
3. Перейдите в панель **Свойства**, и измените **Тип парковки** с параллельной на перпендикулярную. В итоге парковка должна выглядеть следующим образом (см. рисунок справа).
4. По умолчанию парковка создается с пятью парковочными местами. **Количество парковочных мест** можно изменить в свойствах элемента, но в нашем случае нам нужна парковка именно на пять машин.
5. Если нужно, то длину парковочного места можно задать, изменив значение свойства **Длина парковочного места**. Ширина же определяется свойством **Ширина полосы дорожной сети**, которое влияет на все элементы этой сети: дороги, перекрестки, и т.д.



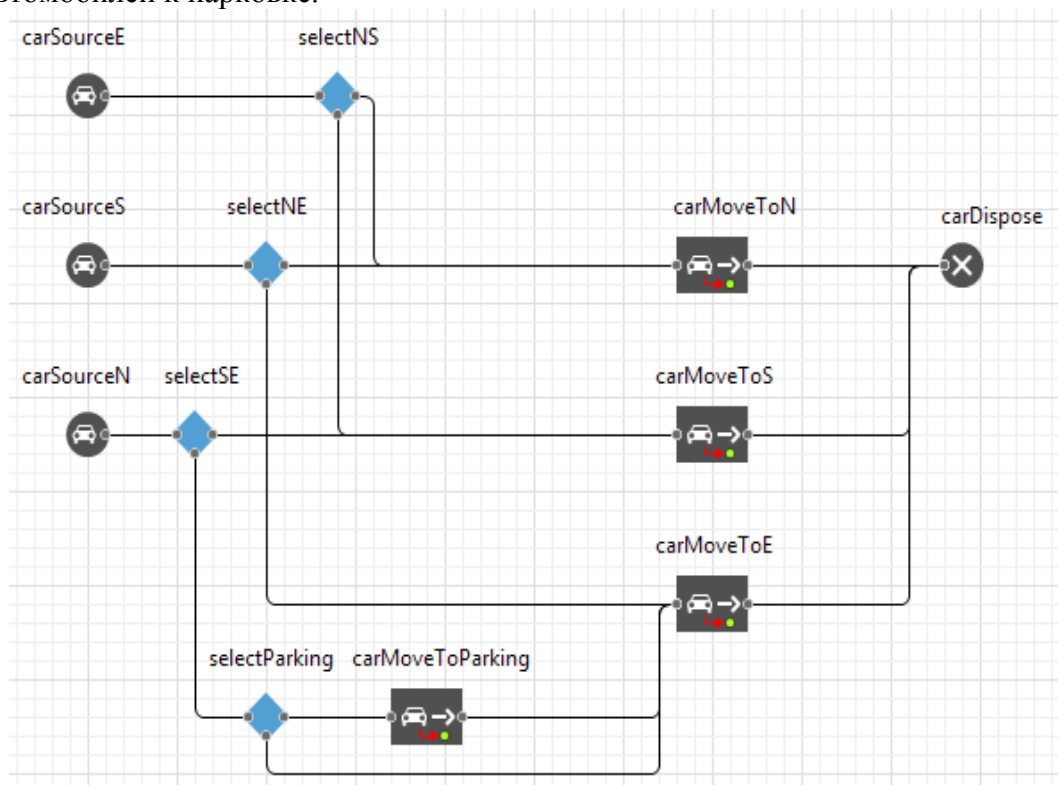
Теперь нужно добавить в диаграмму процесса блоки, которые будут моделировать заезд автомобилей на парковку и пребывание там в течение определенного времени. Чтобы не перегружать диаграмму процесса множеством аналогичных блоков, промоделируем заезд на парковку только для части тех машин, которые движутся по улице Tapiolavagen с севера, и затем поворачивают на Menninkaisentie.

Измените диаграмму процесса

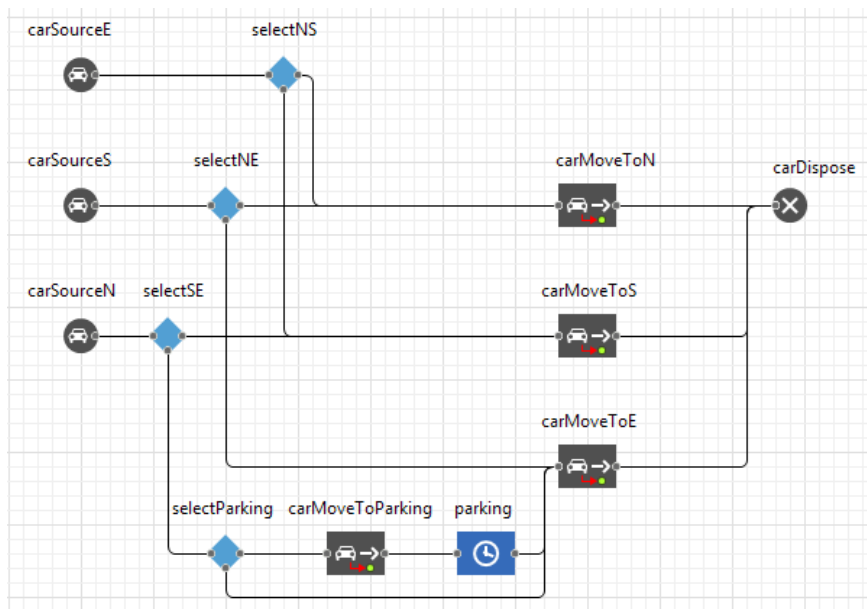
1. Необходимо, чтобы теперь часть машин, едущих с севера и поворачивающих на Menninkaisentie, заезжала на парковку, расположенную на обочине этой дороги. Первым делом нужно удалить соединитель, ведущий из блока *selectNE* в блок *carMoveToE*, поскольку в этом месте диаграммы нужно будет произвести еще одно деление потока машин - часть машин поедет на парковку, в то время, как остальные - до конца дороги Menninkaisentie.
2. Для того, чтобы произвести это деление потока машин, добавьте еще один блок  **SelectOutput**. Назовите этот блок *selectParking* и соедините его порты с портами других блоков, как это показано на рисунке ниже.



3. Необходимо направлять на парковку только небольшую часть потока машин (скажем, 10 процентов), поэтому в поле **Вероятность**: 0.1. Таким образом задаем долю автомобилей, направляемую блоком в свой верхний выходной порт.
4. Добавьте еще один блок **CarMoveTo** (в ветвь процесса, ведущую из верхнего порта блока *selectParking*, см. рисунок ниже). Назовите его *carMoveToParking*. Этот блок будет моделировать движение автомобилей к парковке.

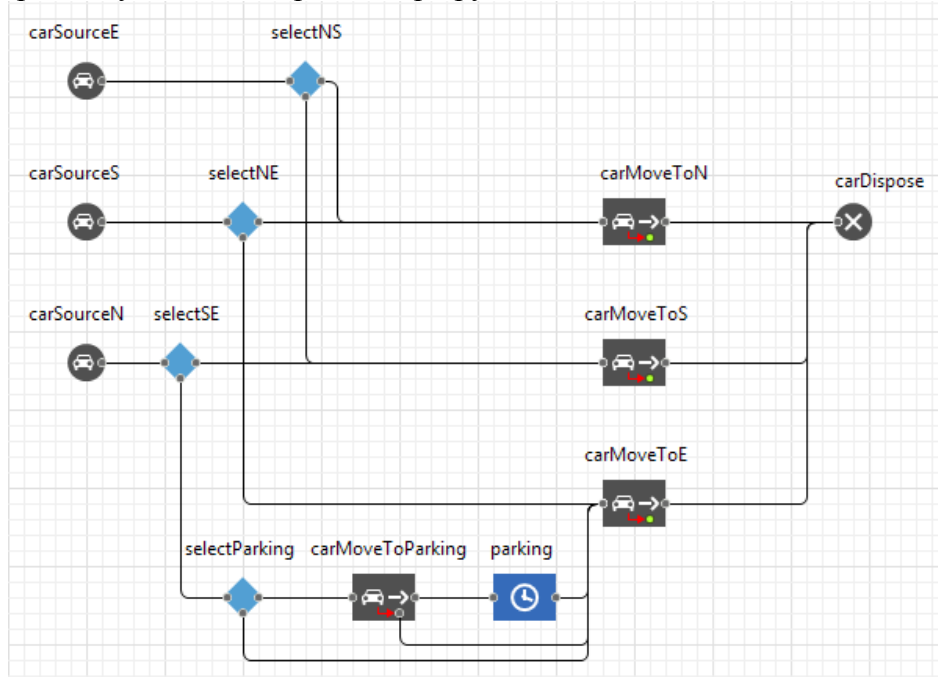


5. В свойствах блока выберите **Цель движения: парковка**, и затем выберите в поле **Парковка** имя созданной нами ранее парковки: *parkingLot*.
6. Теперь нужно промоделировать, как приехавшие на парковку автомобили пребывают на ней в течение определенного времени. Для этого добавьте блок Библиотеки моделирования процессов **Delay**. Назовите этот блок *parking* и соедините его так, как это показано на рисунке ниже.



7. Данный блок моделирует задержку, связанную с выполнением агентом определенной операции (в нашем случае - ожидание на парковке). Откройте свойства этого блока. Измените **Время задержки** на triangular (5, 15, 45) **минут**.
8. Поскольку на парковке могут одновременно находиться до 5 машин, введите в поле **Вместимость**: 5. Этот параметр задает максимальное количество агентов (в нашем случае - автомобилей), которое может одновременно находиться в данном блоке.
9. Запустите модель. Видно, как время от времени автомобили будут заезжать на парковку. Включите выполнение модели в режиме максимально возможной скорости, щелкнув по кнопке **Реальное/виртуальное время**. Тогда вы, скорее всего, увидите сообщение об ошибке. Причина этого в том, что в какой-то момент автомобиль будет направлен на парковку, а свободных мест на ней не окажется. Учтем этот случай, внося минимальные изменения в нашу диаграмму процесса.

Заметим, у блока **CarMoveTo** помимо двух стандартных портов на левой и на правой границе блока есть и еще один, помещенный на нижнюю границу значка. К нему ведет красная стрелочка. Этот порт называется *outWayNotFound* и представляет собой выходной порт, в который перенаправляются те автомобили, для которых не удалось построить маршрут к заданной в этом блоке цели движения.



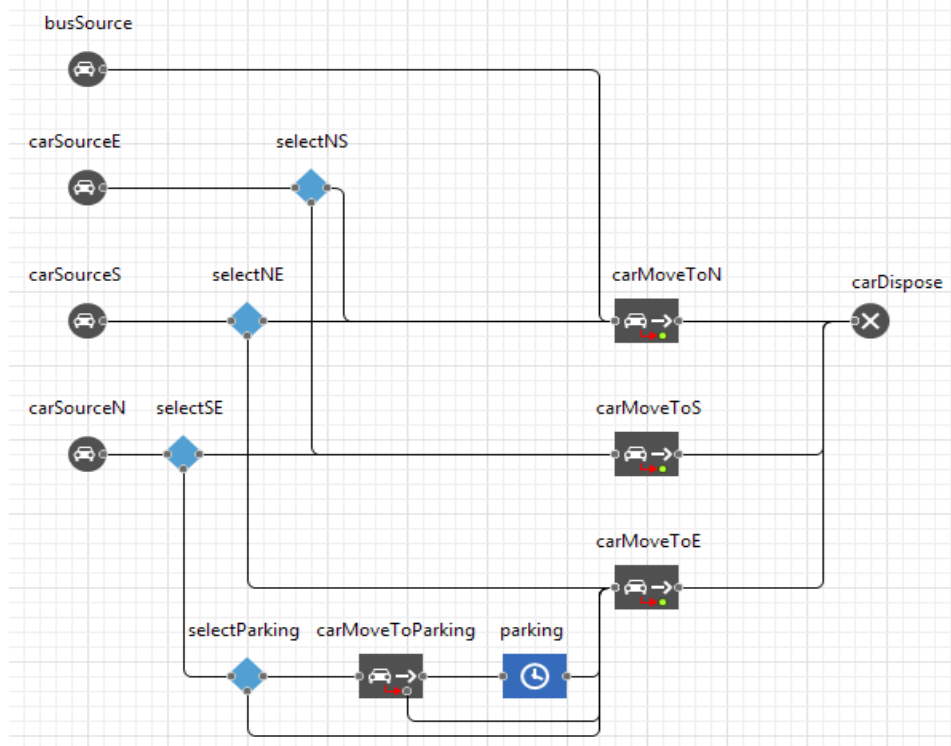
Соедините нижний выходной порт блока *carMoveToParking* с входным портом блока *carMoveToE*. В нашем случае автомобили будут направляться в этот порт, если на момент поступления автомобиля в блок *carMoveToParking* на парковке не окажется свободных мест. Таким образом, если парковка будет полностью занята, то автомобили будут просто продолжать движение дальше по дороге. Запустите модель и убедитесь в том, что модель работает корректно, и ошибок больше не возникает.

Шаг 5. Добавление автобусов

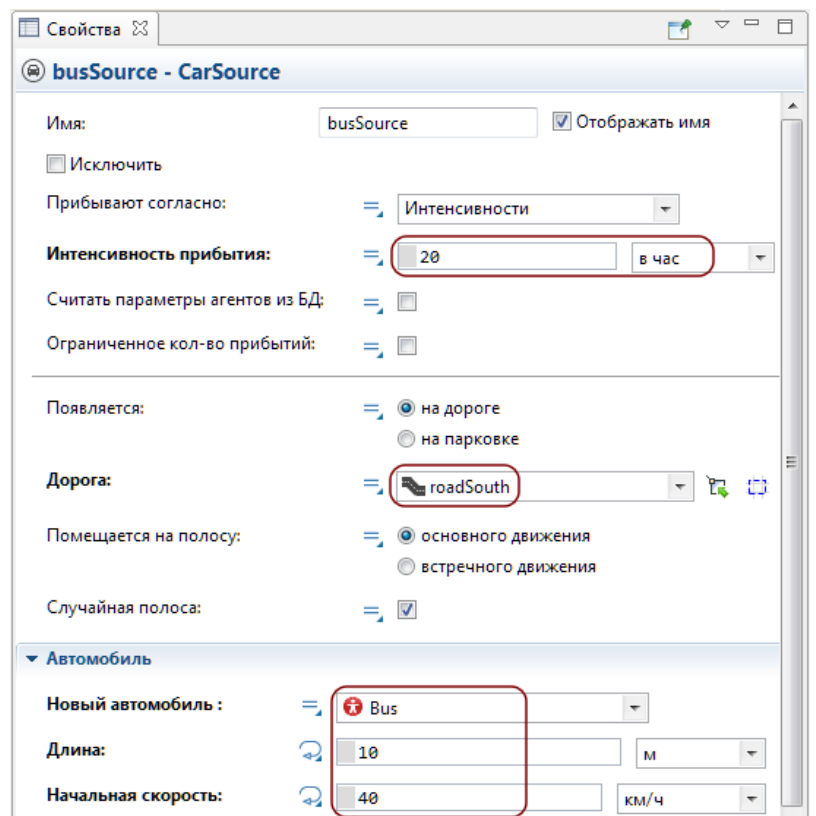
Сейчас в нашей модели присутствуют только легковые автомобили. Давайте сделаем следующий шаг по усовершенствованию модели, добавив в нее рейсовые автобусы. Автобусы будут проезжать по Tarpolavagen с юга на север и останавливаться на остановке (вы можете увидеть ее на спутниковом снимке, на обочине за перекрестком).

1. Добавьте блок-источник автобусов

1. Добавьте еще один блок Библиотеки дорожного движения **CarSource**. Этот блок будет моделировать появление автобусов. Назовите блок *busSource* и соедините его с блоком *carMoveToN*.

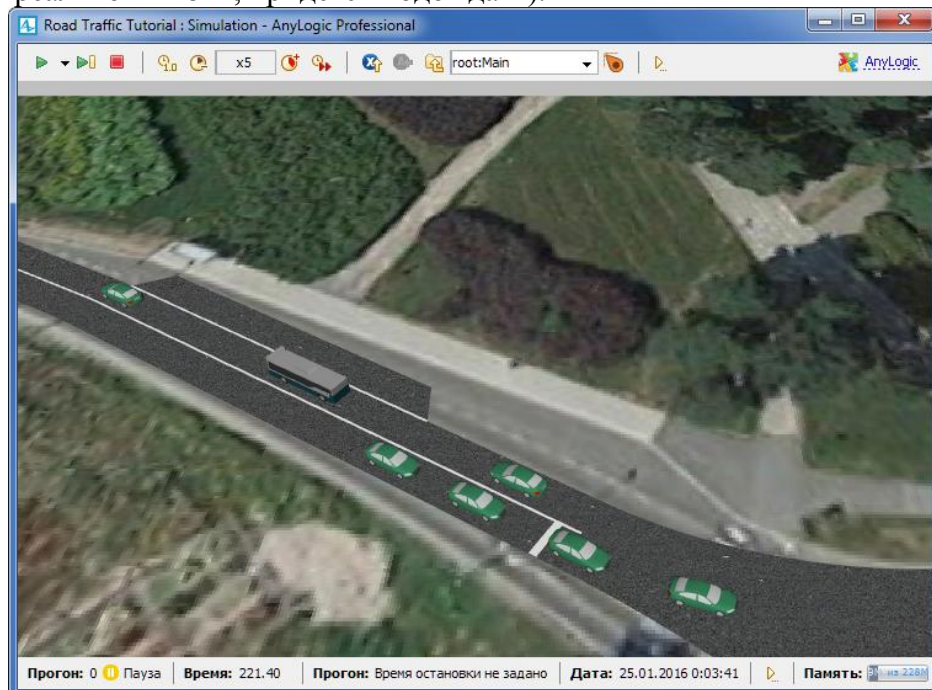


2. Измените свойства только что созданного блока. Автобусы будут появляться в модели существенно реже, чем легковые автомобили. Для начала зададим **Интенсивность** появления равной **20 в час**.
3. В свойстве **Дорога** выберите имя дороги *roadSouth*.
4. Раскройте секцию свойств **Автомобиль** и в поле **Длина** задайте длину создаваемого автобуса: **10 метров**.
5. В поле **Начальная скорость** введите **40 км/ч**.
6. Нужно, чтобы этот блок создавал не легковые автомобили, как это делают созданные нами ранее блоки **CarSource**, а автобусы. Для этого нам понадобится создать еще один тип агента - автобус. Раскройте секцию свойств **Автомобиль** и щелкните по ссылке **создать новый тип**.
7. При этом откроется диалоговое окно Мастера создания типа агента. Введите в поле **Имя нового типа** *Bus* и нажмите **Далее**.
8. На следующей странице Мастера оставьте выбранной опцию **3D** и выберите в списке ниже фигуру анимации **Автобус 1**. Щелкните по кнопке **Готово**.





Переключитесь обратно на диаграмму Main и сверьтесь со следующим рисунком, все ли свойства блока вы правильно задали.

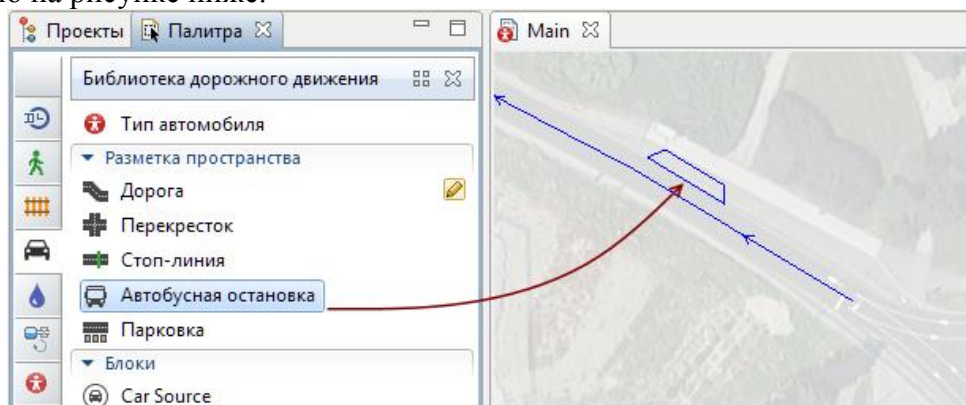
Запустите модель. Видно, теперь время от времени с юга на север проезжают автобусы (возможно, их появления, как и в реальной жизни, придется подождать).



На скриншоте сверху автобус как раз проезжает мимо автобусной остановки. Теперь изменим логику модели, чтобы автобусы начали заезжать на остановку.

 *Нарисуйте автобусную остановку*


1. Перетащите элемент **Автобусная остановка**  из секции **Разметка пространства** палитры **Библиотека дорожного движения** на графическую диаграмму. Поместите остановку на обочине, как это сделано на рисунке ниже.

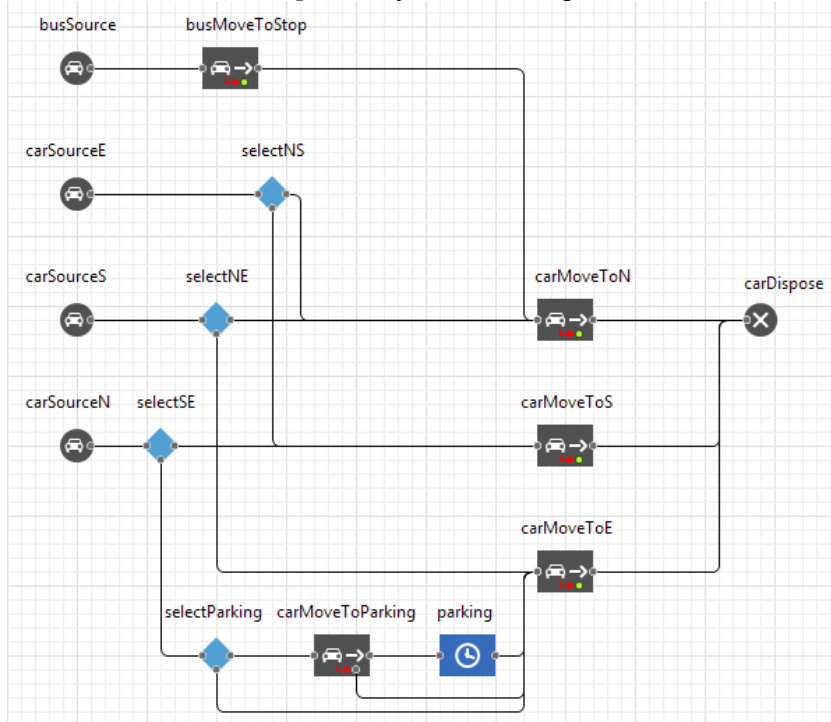



2. Удлините остановку. Для этого выделите ее щелчком мыши и затем перетащите квадратный маркер на ее краю.

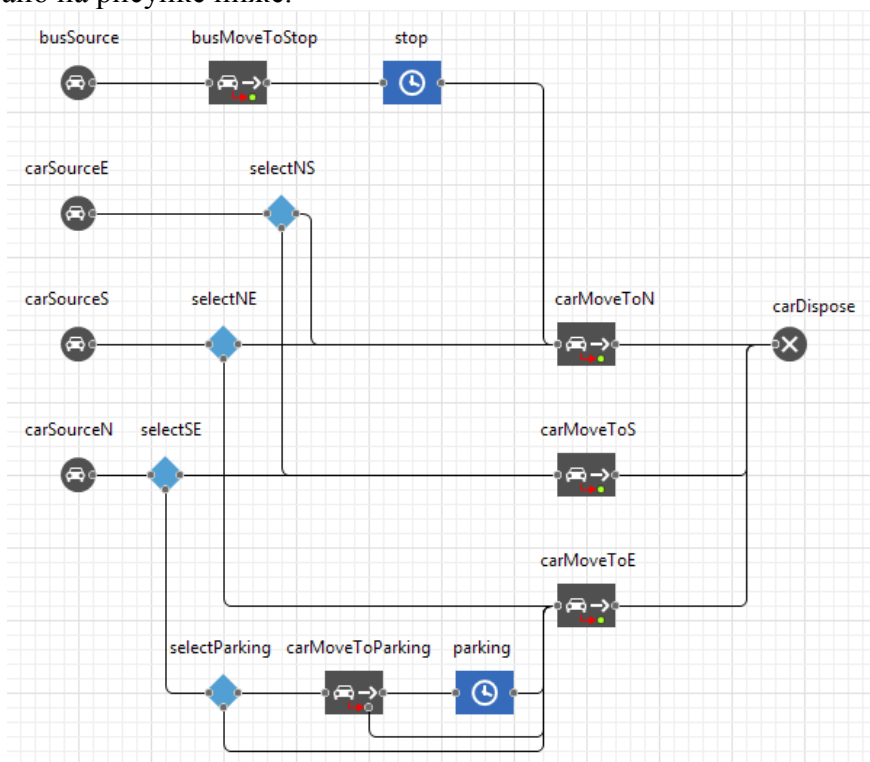


Теперь добавим новые блоки в диаграмму процесса.

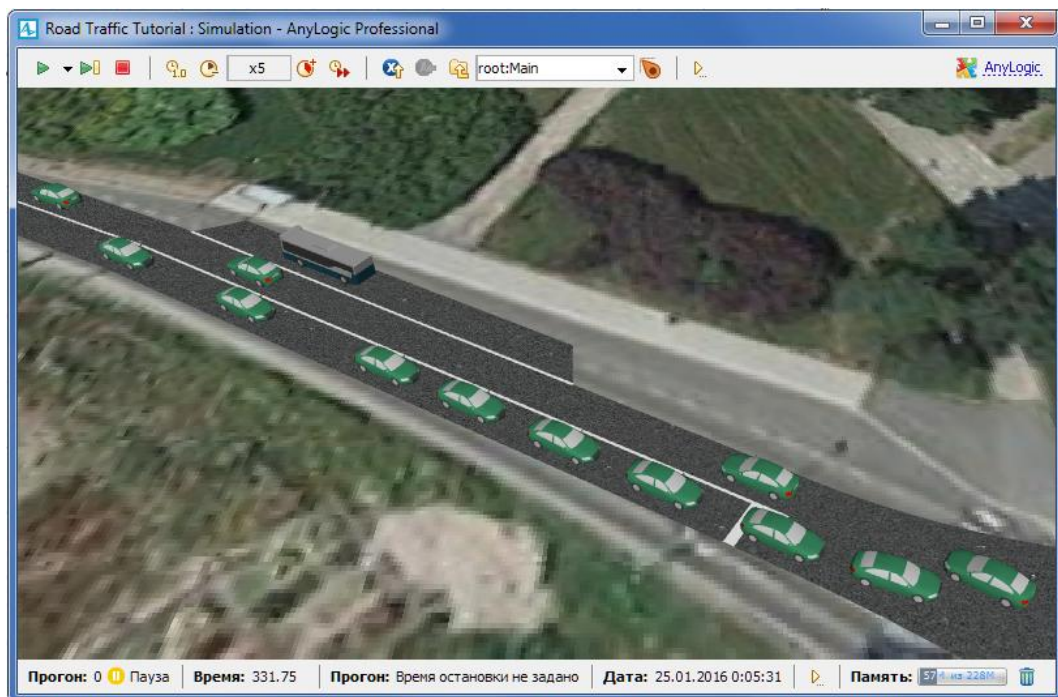
1. Добавьте еще один блок  **CarMoveTo** (в ветвь процесса, ведущую из блока *busSource*, см. рисунок ниже). Назовите этот блок *busMoveToStop*. Он будет моделировать движение автобусов к остановке.



2. В свойствах блока *busMoveToStop* выберите **Цель движения: автобусная остановка** и затем выберите в поле **Автобусная остановка** имя созданной нами ранее остановки: *busStop*.
3. Теперь нужно промоделировать, как приехавшие на остановку автобусы пребывают на ней в течение определенного времени. Для этого добавьте блок  **Delay**. Назовите этот блок *stop* и соедините его так, как это показано на рисунке ниже.



4. Откройте свойства этого блока и задайте там **Время задержки: triangular (15, 30, 90) секунд**. Запустите модель. Видно, как автобусы заезжают на остановку и ждут там, пока пешеходы не выйдут и не зайдут в автобус, после чего продолжают свое движение.




Шаг 6. Добавление светофоров

На этом шаге сделаем перекресток регулируемым, добавив светофоры.

Для этого понадобится блок Библиотеки дорожного движения **Traffic Light**. Блок **Traffic Light** задает логику работы одного или нескольких светофоров, регулирующих движение машин на перекрестке или на пешеходном переходе.

1.2 Задайте логику работы светофоров

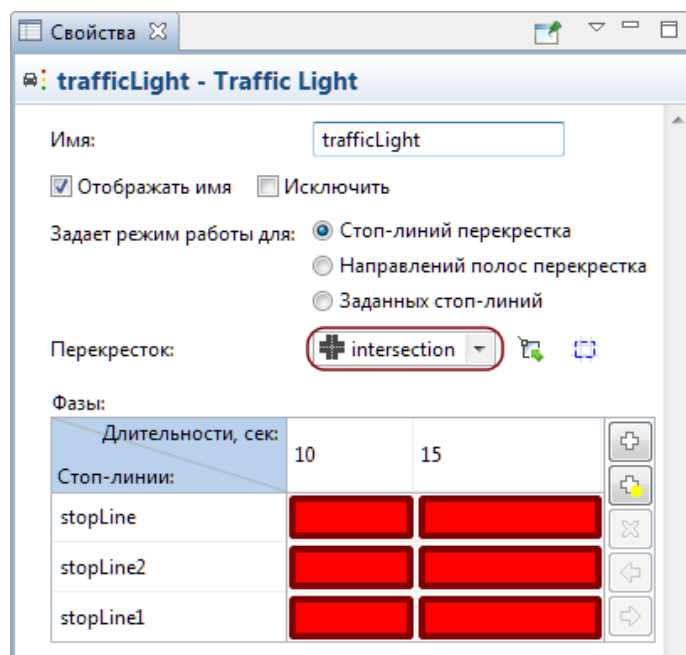
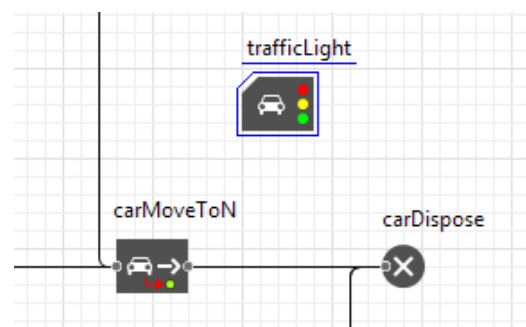
1. Добавьте на диаграмму блок **Traffic Light**  из секции **Блоки** палитры **Библиотека дорожного движения**. У этого блока нет портов, и его не нужно соединять ни с какими другими блоками диаграммы процесса. Его можно добавить в абсолютно любое место графического холста.

2. Откройте свойства блока. Обратите внимание на ключевой параметр блока: **Задает режим работы для**. У этого параметра три альтернативные опции.

Опция **Стоп-линий перекрестка** (выбрана по умолчанию). В нашем случае мы оставим выбранную по умолчанию опцию поскольку задаваемые этим блоком светофоры будут регулировать движение на перекрестке, причем каждая фаза светофора будет разрешать / останавливать движение сразу на всех полосах примыкающей к перекрестку дороги.

3. В свойстве **Перекресток** выберите имя нарисованного нами перекрестка: *intersection*.

4. После того, указан перекресток в свойствах блока **Traffic Light**, все стоп-линии этого перекрестка появятся ниже в таблице **Фазы**. Здесь можно задать режим работы светофора - как будут меняться красная, зеленая (а опционально и желтая) фазы для входящих в выбранный перекресток стоп-линий.

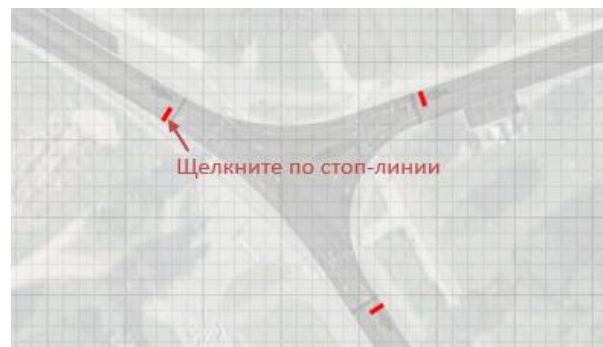


5. Давайте настроим фазы светофора. Пусть 30-секундная зеленая фаза светофора для движения по главной дороге Tarpilavagen (на время которой въезд на перекресток с улицы Menninkaisentie приостанавливается) сменяется 20-секундной красной фазой (зеленой для въезда с Menninkaisentie), и эти фазы бесконечно сменяют друг друга. По умолчанию таблица содержит два столбца, по одному для каждой фазы. Необходимо настроить зеленую и красную фазы для каждой стоп-линии. Начнем с первой фазы. Щелкните по заголовку столбца первой фазы и укажите длительность фазы равной 30 секундам.

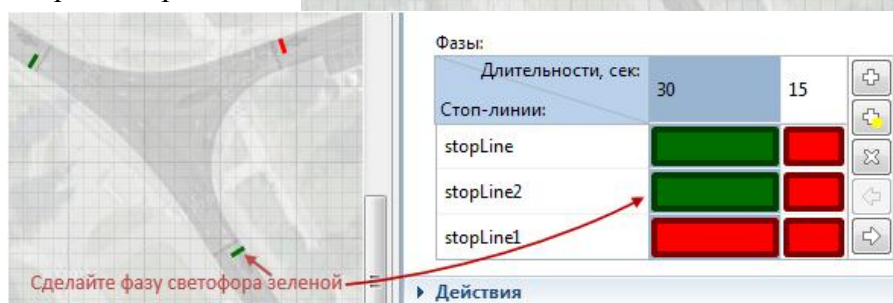
Фазы:

Длительности, сек:	30	15	
Стоп-линии:			
stopLine			
stopLine2			
stopLine1			

6. Поскольку мы выделили ячейку таблицы **Фазы**, соответствующую первой фазе светофора, видно, что в графическом редакторе все фигуры будут временно скрыты, а стоп-линии выделены красным цветом. Это означает, что мы находимся в режиме редактирования фазы, который позволяет менять состояние каждой стоп-линии в этой фазе прямо в графическом редакторе.



7. Щелкните по стоп-линии в графическом редакторе, как показано на рисунке ниже. Вы увидите, что цвет стоп-линии изменится с красного на зеленый.



Аналогичным образом изменится и цвет соответствующей ячейки в таблице **Фазы**.

8. Таким же образом сделайте первую фазу светофора зеленой и для другой стоп-линии, расположенной на той же дороге.

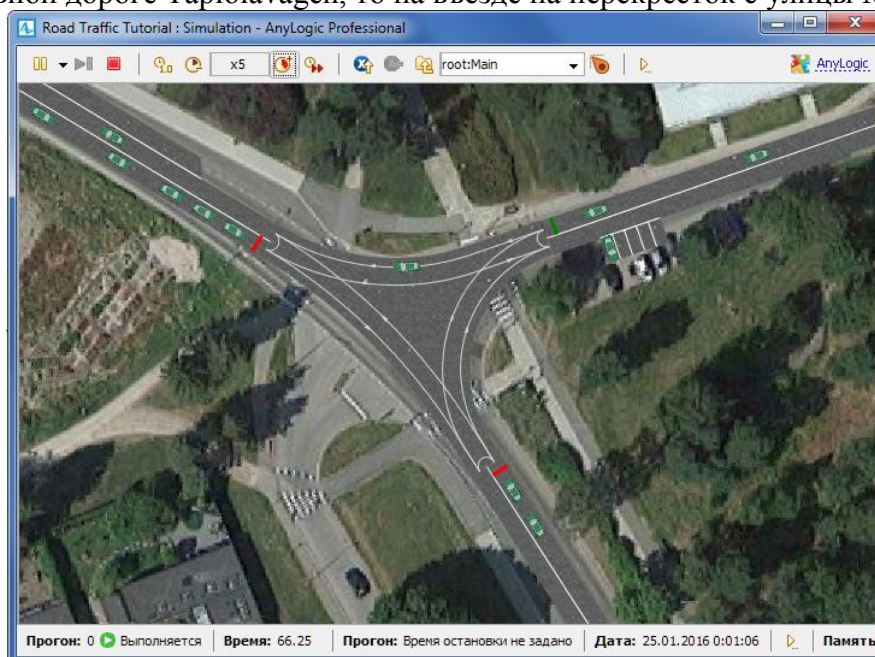
9. Теперь настроим вторую фазу. Щелкните по заголовку столбца второй фазы и укажите ее длительность равной 20 секундам, после чего щелкните по стоп-линии, расположенной на въезде с улицы Menninkaisentie.

Фазы:

Длительности, сек:	30	20	
Стоп-линии:			
stopLine			
stopLine2			
stopLine1			


10. В итоге таблица **Фазы** должна выглядеть следующим образом.

Запустите модель. Вы сможете увидеть, как поочередно меняются фазы светофоров, разрешающие движение то по главной дороге Tarpilavagen, то на въезде на перекресток с улицы Menninkaisentie.

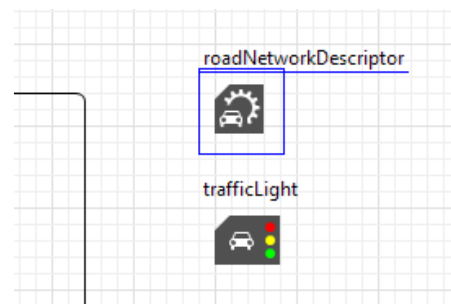


Чтобы лучше понимать, как загружены дороги при заданном режиме работы светофоров, можно включить *отображение пробок*.

 Включите отображение пробок

1. Добавьте на диаграмму блок **Road Network Descriptor**  из секции **Блоки** палитры **Библиотека дорожного движения**. Этот блок также не нужно соединять ни с какими другими блоками диаграммы процесса.
2. Измените свойства блока. Выберите имя нарисованной вами сети дорог *roadNetwork* в раскрывающемся списке **Дорожная сеть**.
3. Откройте секцию свойств **Карта пробок на дороге** и установите флажок **Показывать пробки**.








Запустите модель и изучите ситуацию на перекрестке.



Теперь вы можете попробовать изменить режим работы светофоров (например, задать фазы длительностью 30 / 20 или 30 / 30 секунд) и изучить, при каком режиме достигается наилучший для водителей результат.

Это один из примеров практического применения Библиотеки дорожного движения. Обычно модели дорожного трафика создаются для анализа и сравнения различных вариантов организации движения (как в нашем случае с регулировкой светофоров), моделирования проектируемых развязок и дорог с целью выявления узких мест, проверки целесообразности увеличения или уменьшения количества полос на определенном отрезке существующей дорожной сети и т.д. Кроме того, с помощью Библиотеки дорожного движения вы можете добавить (трехмерную) анимацию движения автотранспорта в ваших моделях (автобусы-шаттлы и такси в аэропорту, фуры в логистических центрах и т.д.), тем самым выведя анимацию вашего проекта на качественно иной уровень.

Контрольные вопросы:

1. Перечислите назначение следующих блоков:  **CarSource**,  **CarMoveTo**,  **CarDispose**.
2. Для чего используется 3D окно при создании модели в Anylogic.
3. Опишите назначение кнопки .
4. В перечень инструментов, какой библиотеки программы Anylogic входят блоки:  **SelectOutput**,  **SelectOutput5**,  **Delay**? Опишите их назначение?
5. Для чего применяется блок Библиотеки дорожного движения **Traffic Light**?