

Модель аэропорта

Эта презентация является частью
стандартной программы обучения



Модель аэропорта

- Постройте имитационную модель движения пассажиров по небольшому аэропорту.
- Пассажиры прибывают в аэропорт, регистрируются на рейс, проходят досмотр и направляются в область ожидания у гейта. При объявлении начала посадки на рейс пассажиры проходят проверку билетов и затем проходят на борт самолета.
- В последней фазе мы смоделируем ситуацию распространения инфекции в аэропорту. Время от времени зараженные люди будут входить в аэропорт, и в случае длинных очередей они будут заражать пассажиров, которые будут находиться рядом с ними достаточно времени.



Модель аэропорта. Фаза 1

- На первой фазе мы создадим простую модель, в которой пассажиры прибывают в аэропорт и движутся по его помещению к гейту.

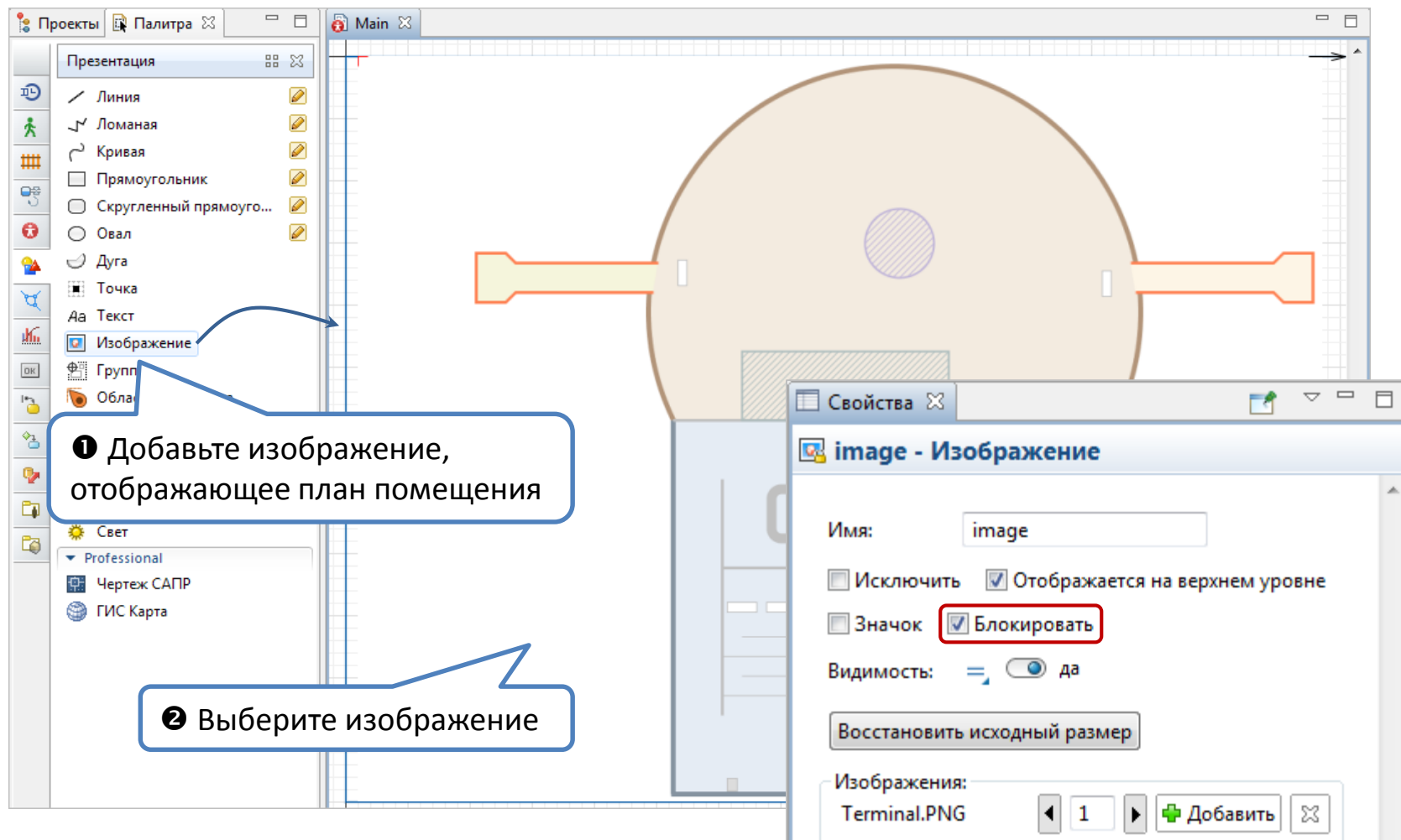


Пешеходная библиотека

- Традиционное (дискретно-событийное, очередное) моделирование может дать некорректные результаты в областях плотного пешеходного движения.
- **Пешеходная библиотека** AnyLogic является высокоуровневой библиотекой моделирования движения пешеходов в физическом пространстве. Она позволяет моделировать здания, в которых движутся пешеходы (станции метро, стадионы, музеи), а также улицы и другие места большого скопления людей.
- В моделях, созданных с помощью объектов Пешеходной библиотеки, пешеходы движутся в непрерывном пространстве, реагируя на различные виды препятствий в виде стен, различных областей и других пешеходов.
- Вы можете создать впечатляющую визуализацию для презентации и валидации Вашего проекта, собрать статистику плотности пешеходов в различных областях модели для того, чтобы убедиться, что сервисы смогут справиться с потенциальным ростом нагрузки, вычислить время пребывания пешеходов в каких-то определенных участках модели, выявить возможные проблемы, которые могут возникнуть при перепланировке интерьера здания, и т.д.



Аэропорт. Фаза 1. Шаг 1



Создайте новую модель. Назовите ее *Airport*.

Обычно создание моделей пешеходной динамики начинается с добавления плана моделируемого пространства и рисования стен поверх стен, нарисованных на плане.

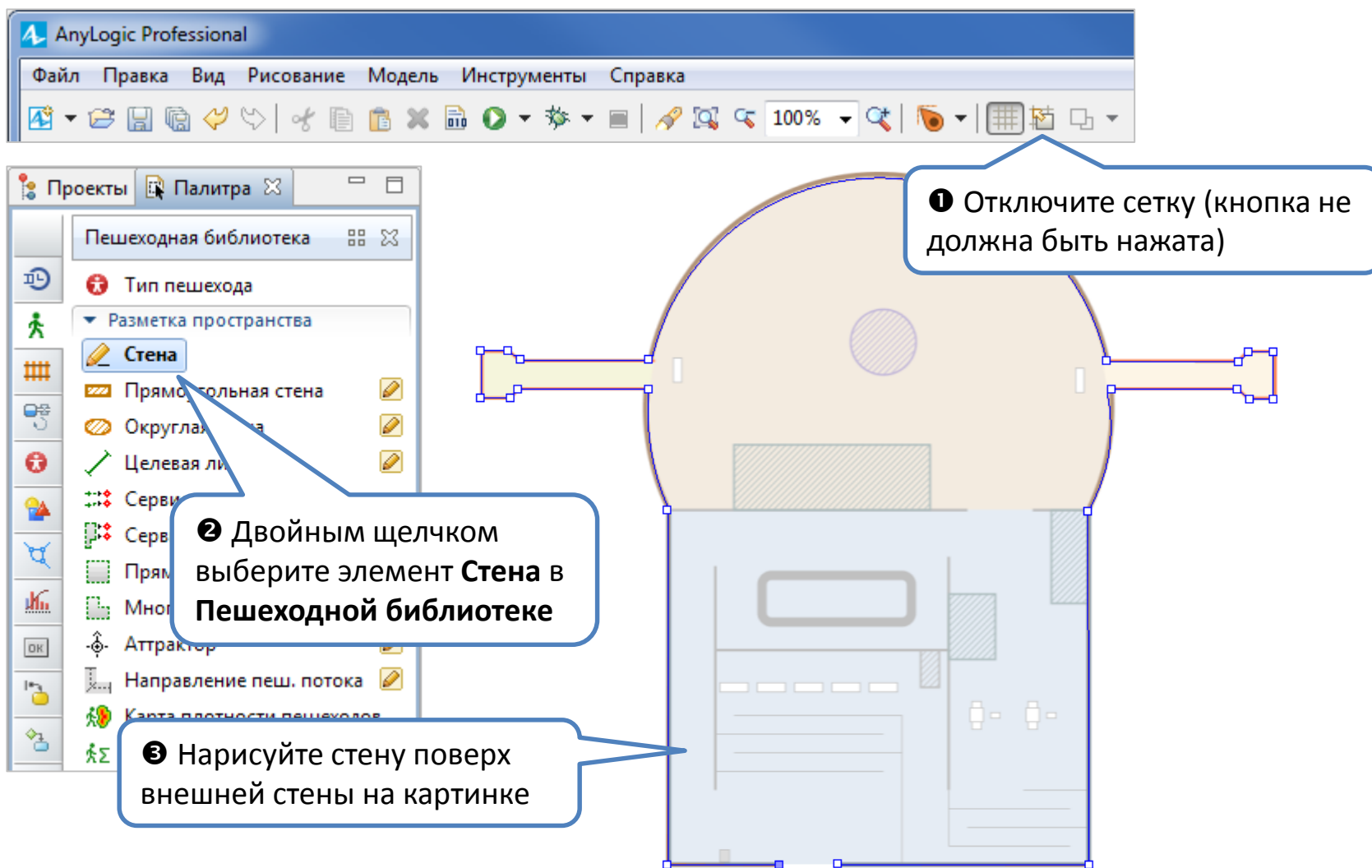
- ❶ Добавьте *Изображение* из палитры **Презентация**.
- ❷ Выберите файл изображения, которое будет отображаться этим элементом: [terminal.png](#) из папки [Models\Airport](#) на Вашем USB.
Выберите флажок **Блокировать**, чтобы заблокировать эту фигуру.

Блокировка фигур презентации

- Вы можете *заблокировать* фигуры презентации. Заблокированные фигуры не будут реагировать на щелчки мыши, пока Вы не снимете с них блокировку.
- Обычно блокировка требуется, когда на презентации есть фоновый рисунок (например, план моделируемого Вами предприятия), используемый как подложка для анимации. В этом случае при редактировании фигуры, нарисованной поверх этой подложки, Вы можете случайно выделить и отредактировать саму подложку. Если же Вы заблокируете фоновый рисунок, то Вы больше не сможете выделить этот рисунок ошибочным щелчком мыши.



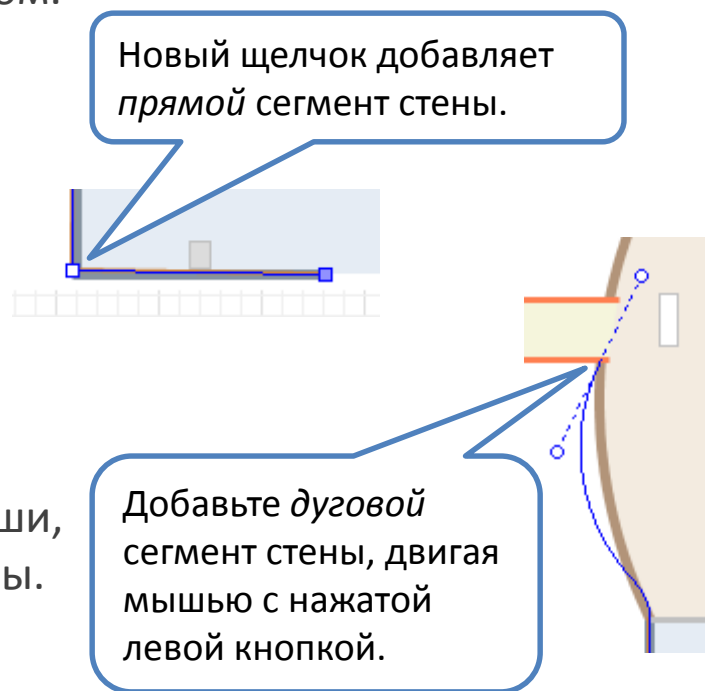
Аэропорт. Фаза 1. Шаг 2



Теперь давайте зададим внешнюю стену с помощью элемента разметки пространства **Стена**.

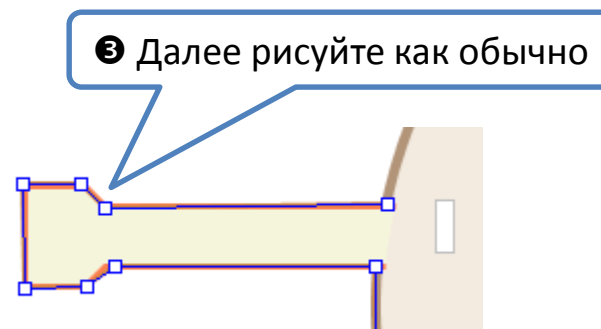
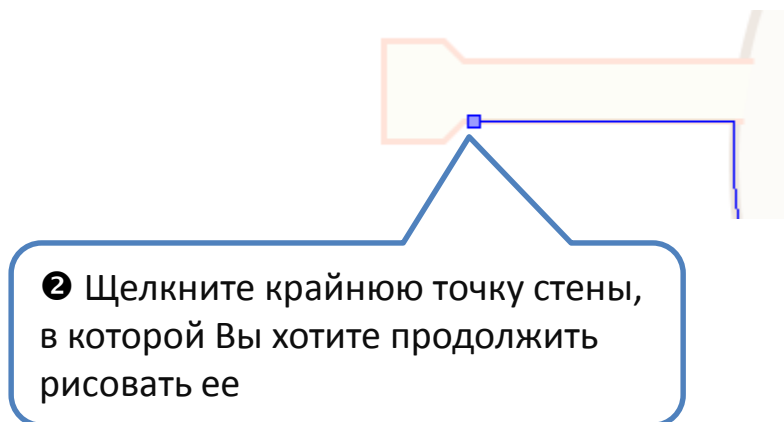
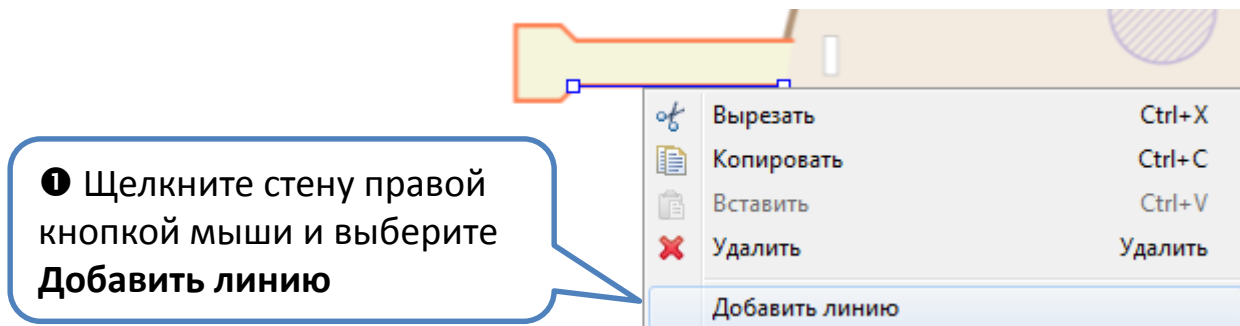
Рисование стен

- При рисовании стен используйте режим рисования: вначале сделайте двойной щелчок по элементу **Стена** в палитре, а затем нарисуйте стену на диаграмме, последовательно щелкнув мышью в точках изгиба линии и добавив конечную точку двойным щелчком.
- Чтобы добавить *прямую* стену, просто сделайте щелчок мышью там, где заканчивается стена.
- Чтобы добавить *дуговой* сегмент, не отпускайте кнопку мыши после щелчка и двигайте курсором. Отпустите кнопку мыши, когда этот сегмент будет требуемой формы.



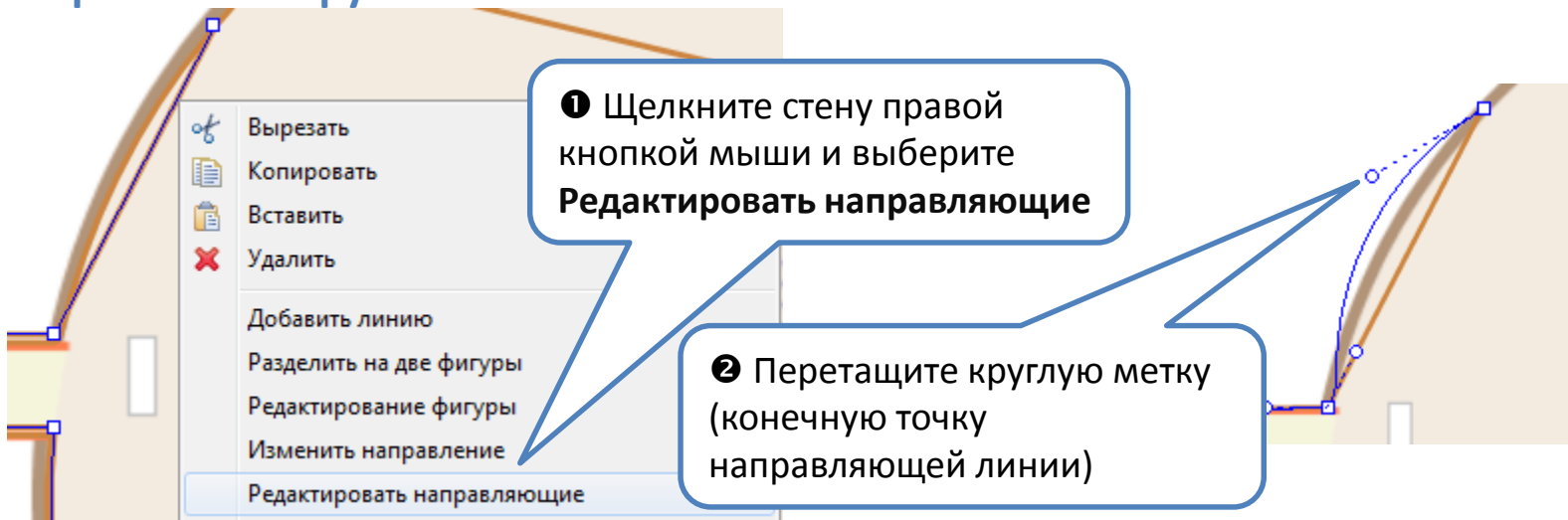
Как продолжить рисовать стену

- Вы можете завершить рисование стены на какое-то время (двойным щелчком) и продолжить позже, пользуясь указаниями ниже:

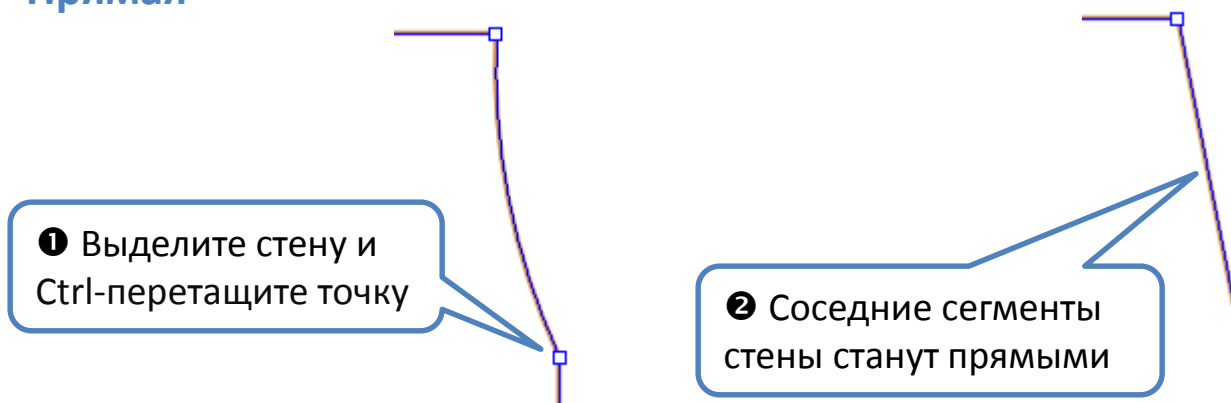


Как изменить прямую стену в округлую (и наоборот)

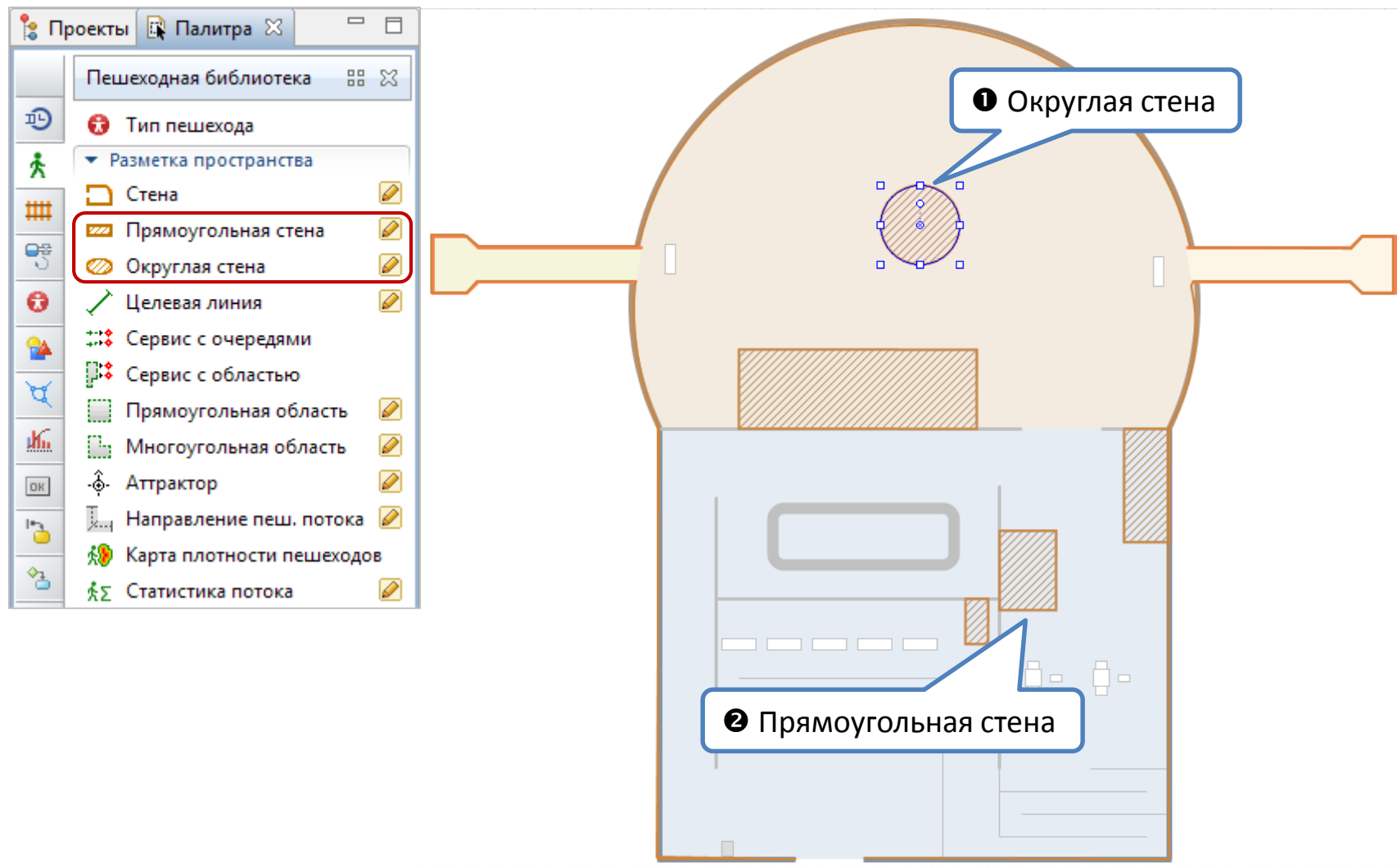
Прямая -> Округлая



Округлая -> Прямая



Аэропорт. Фаза 1. Шаг 3



Давайте продолжим рисовать стены и препятствия в здании аэропорта.


- ❶ Нарисуйте колонну с помощью элемента **Округлая стена**.
- ❷ Нарисуйте зоны рабочего пространства с помощью элемента **Прямоугольная стена**.

Стены

Вы можете использовать следующие три элемента разметки, чтобы задать стены на своем рисунке:

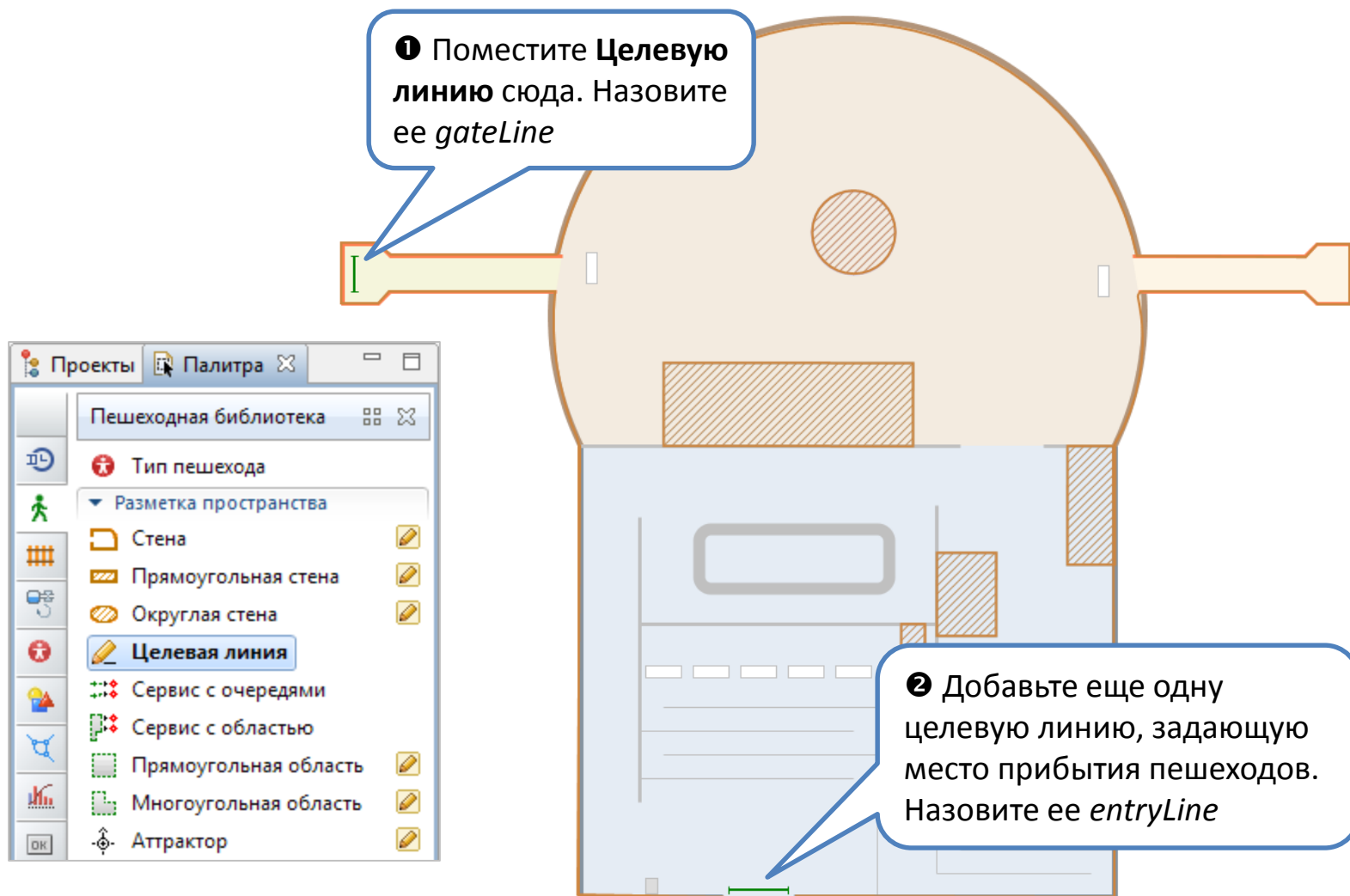
 **Стена** - Используется для рисования стен сложной формы (внешние стены)

 **Прямоугольная стена** - Обычно используется для задания прямоугольных помещений внутри заданного стеной пространства, закрытых для пешеходов (служебные помещения и т.д.)

 **Округлая стена** - Используется для задания округлых пространств внутри заданного стеной пространства, закрытых для пешеходов (колонны, фонтаны и т.д.)



Аэропорт. Фаза 1. Шаг 4



- ① Нарисуйте линию, к которой пассажиры будут двигаться, войдя в аэропорт. Поместите ее, например, в области гейта. Назовите эту линию *gateLine*.
- ② Обозначьте вход в аэропорт с помощью линии. Назовите ее *entryLine*.

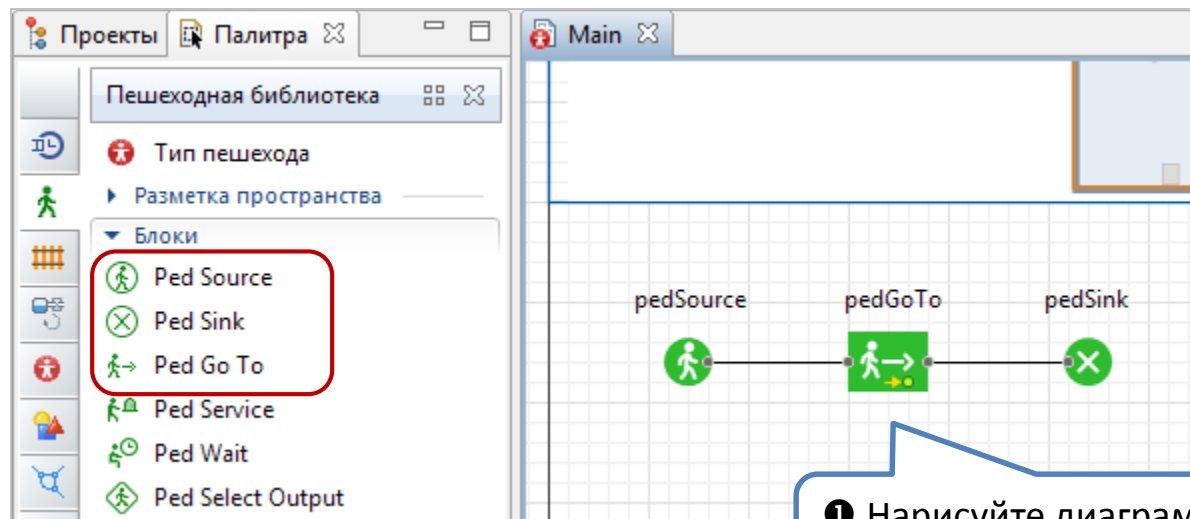
Как линия, на которой будут появляться пешеходы, так и линии, задающие сервисы, очереди к ним и т.д. – все они должны быть помещены внутрь стен, ограничивающих моделируемое пространство, так, чтобы они были достижимы пешеходами.

Целевая линия

- **Целевая линия** используется, чтобы графически задавать в пешеходных моделях следующее:
 - Место, где пешеходы появляются в моделируемом пространстве (указывается в блоках *PedSource* или *PedEnter*)
 - Место назначения движения пешеходов (указывается в блоке *PedGoTo*)
 - Место ожидания пешеходов (указывается в блоке *PedWait*)
 - При моделировании передвижения пешеходов с одного этажа на другой, Вам нужно нарисовать две целевые линии: одну для места, где пешеходы покидают текущий этаж и вторую для места, где пешеходы входят на другой этаж. В этом случае Вы указываете обе целевые линии в блоке диаграммы процесса *PedChangeGround*.



Аэропорт. Фаза 1. Шаг 5



❶ Нарисуйте диаграмму
PedSource - PedGoTo - PedSink



- ❶ Создайте диаграмму простого процесса. Начните с блока создания пешеходов **PedSource**, следующим разместите блок **PedGoTo**, который моделирует движение пешеходов к определенному месту, и в конце расположите блок **PedSink**, который уничтожает поступающих в него пешеходов. Каждая пешеходная диаграмма процесса обычно начинается с блока **PedSource** и заканчивается блоком **PedSink**.

Задание процессов в моделях пешеходной динамики

- Процессы в моделях пешеходной динамики задаются с помощью диаграмм процессов, так же, как и в традиционных дискретно-событийных моделях, создаваемых с помощью Библиотеки моделирования процессов. Так же, как и агенты в дискретно-событийных моделях, пешеходы движутся по диаграмме процесса, последовательно выполняя операции, заданные блоками диаграммы.



Аэропорт. Фаза 1. Шаг 6

1 PedSource
Целевая линия: *entryLine*
Интенсивность: 100 в час

2 PedGoTo
Целевая линия: *gateLine*

Свойства X

pedSource - PedSource

Имя: ☒ Отображать имя

☐ Исключить

Место появления: ☒ линия
☐ точка (x, y)
☐ область

Целевая линия:
entryLine
gateLine

Прибывают согласно:

Интенсивность: в час

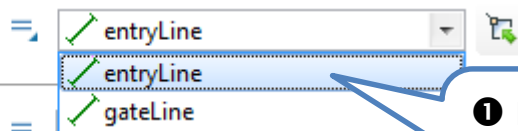
Количество прибытий ограничено: ☐



- 1 Мы указываем в блоке *pedSource*, что генерировать нужно в среднем 100 пассажиров в час и располагать их на целевой линии *entryLine*.
- 2 Блок *pedGoTo* моделирует движение пассажиров к гейту (пункт назначения задается целевой линией *gateLine*).

Выбор элемента из выпадающего списка

Целевая линия:

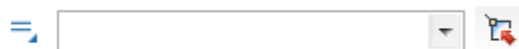


Прибывают согласно:

- 1 Выберите элемент из выпадающего списка, который содержит только элементы подходящего типа

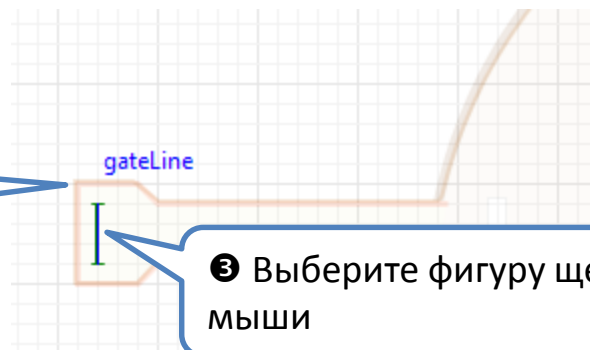
Выбор элемента в графическом редакторе

Целевая линия:



- 1 Щелкните эту кнопку

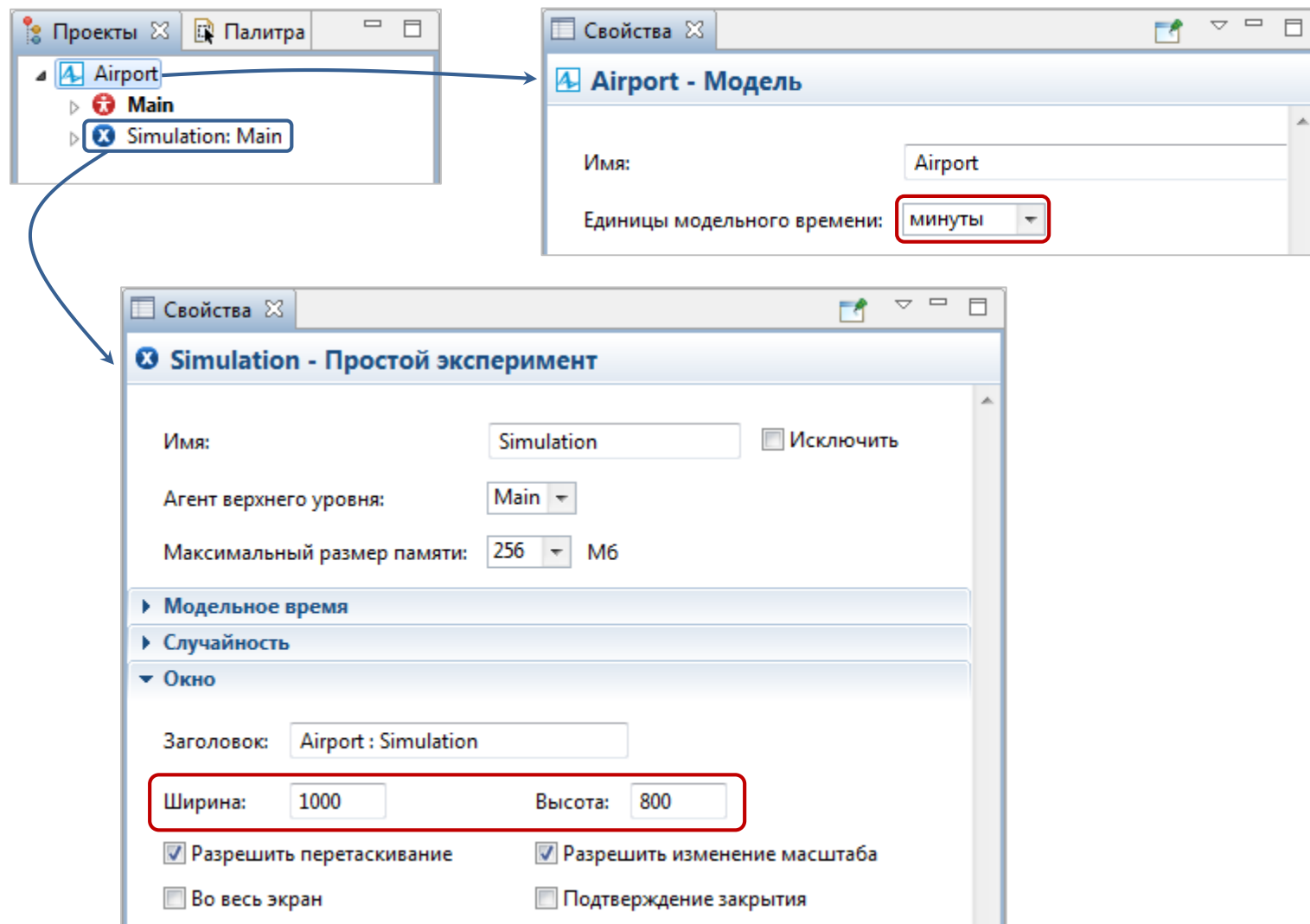
- 2 Вы увидите только элементы подходящего типа, все остальные элементы будут обесцвечены



- 3 Выберите фигуру щелчком мыши



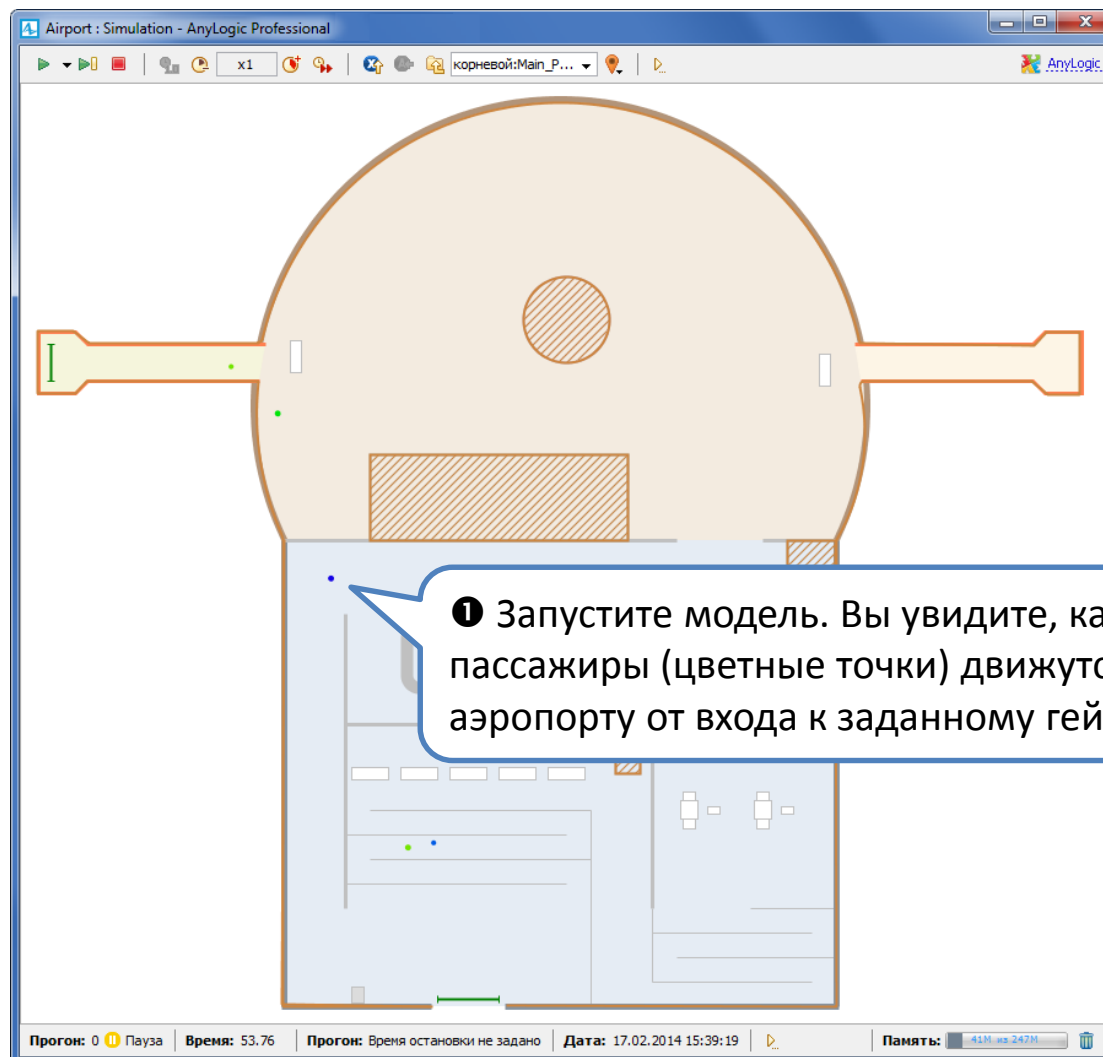
Аэропорт. Фаза 1. Шаг 7



- ❶ Выберите *минуты* как **Единицы модельного времени** (в свойствах модели).
- ❷ Измените размер окна модели как вам угодно (например, на 1000*800).
Если Вы хотите, чтобы окно Вашей модели было на весь экран, выберите опцию **На весь экран**.



Аэропорт. Фаза 1. Шаг 8

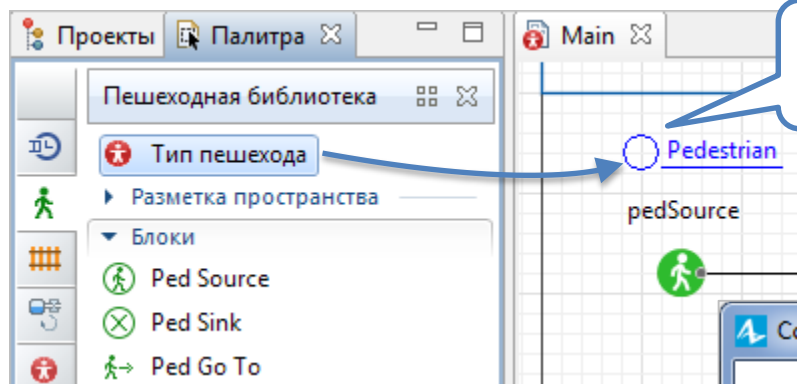


Аэропорт. Фаза 2

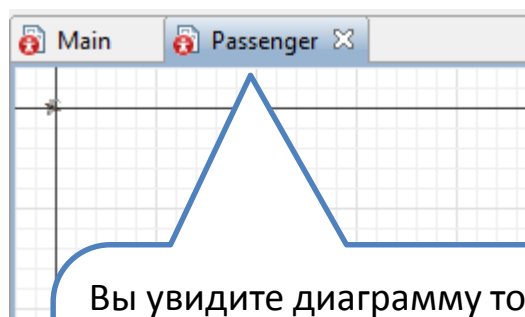
- В этой фазе мы добавим стойки регистрации пассажиров на рейсы.
- Также мы добавим 3D анимацию.



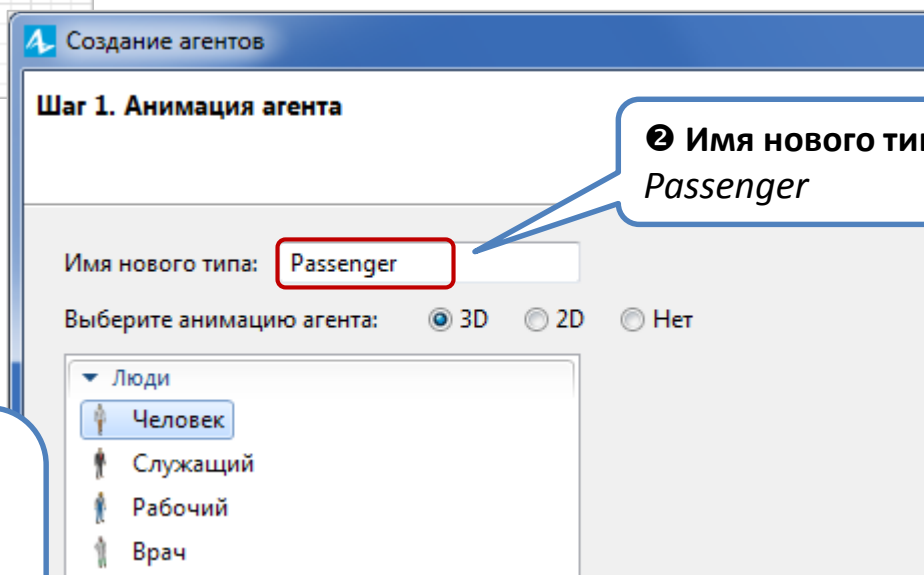
Аэропорт. Фаза 2. Шаг 1



❶ Перетащите Тип пешехода в графический редактор



Вы увидите диаграмму только что созданного типа пешеходов с 3D фигурой анимации в точке пересечения осей. Откройте диаграмму *Main*, чтобы продолжить строить модель.



❷ Имя нового типа: *Passenger*



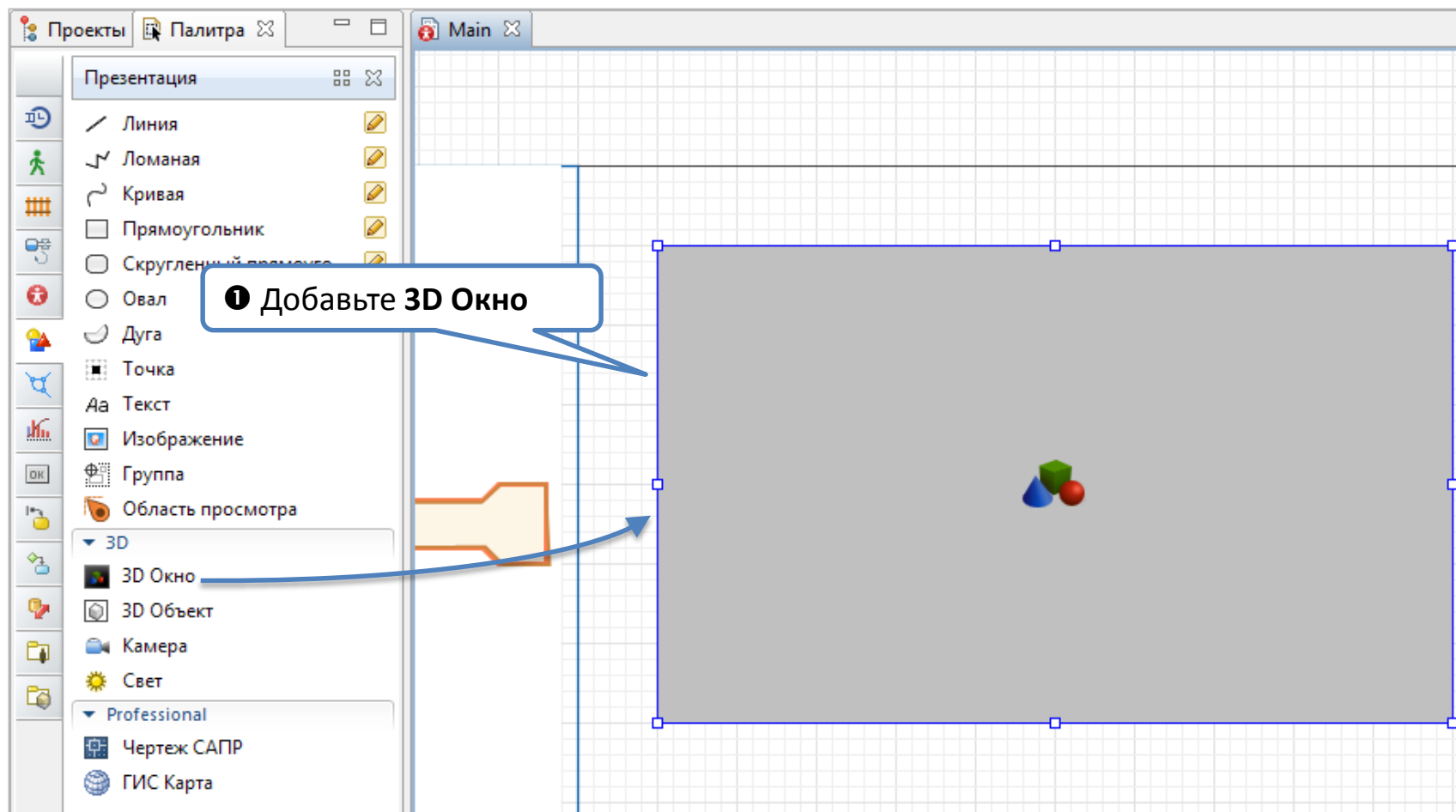
- ❶ Создайте новый тип пешеходов. Нам нужно задать фигуру анимации для наших пешеходов (в этом случае 3D фигуры вместо точек по умолчанию, которые в 3D выглядят просто как цветные цилиндры).
- ❷ Назовите этот тип пешеходов *Passenger*.

Создание нестандартных типов пешеходов

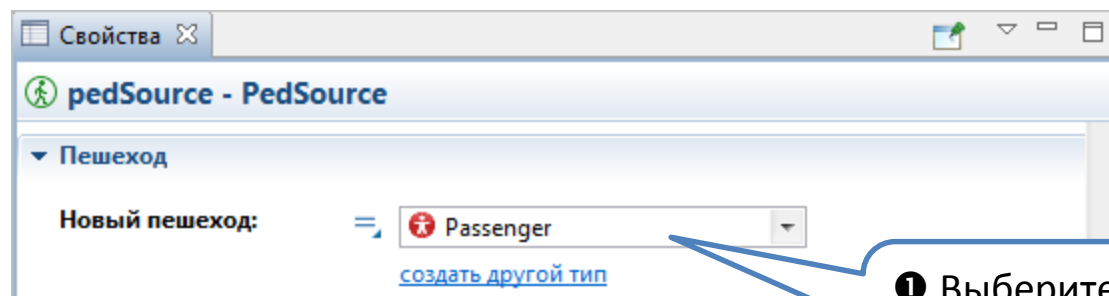
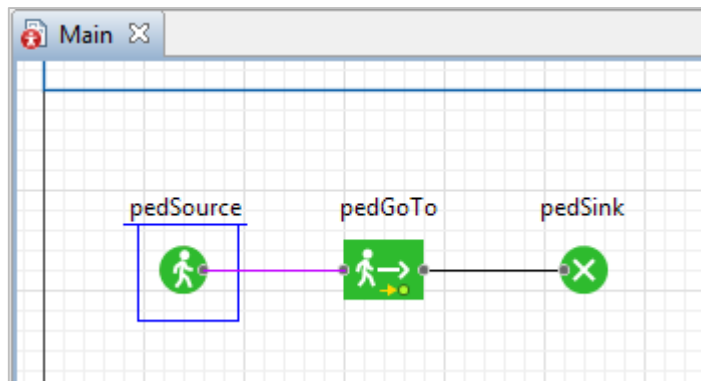
- Если Вам необходимо задать нестандартную анимацию, добавить какие-то свойства или задать поведение пешеходам, Вам нужно создать свой собственный тип пешеходов.
- Так как пешеходы являются агентами и поддерживают всю их функциональность, то Вы можете добавить любые элементы модели в графическую диаграмму типа пешехода так же, как с агентами: задать поведение диаграммой состояния, добавить переменные и параметры, задать системную динамику, создать расписание событий и т.д.



Аэропорт. Фаза 2. Шаг 2



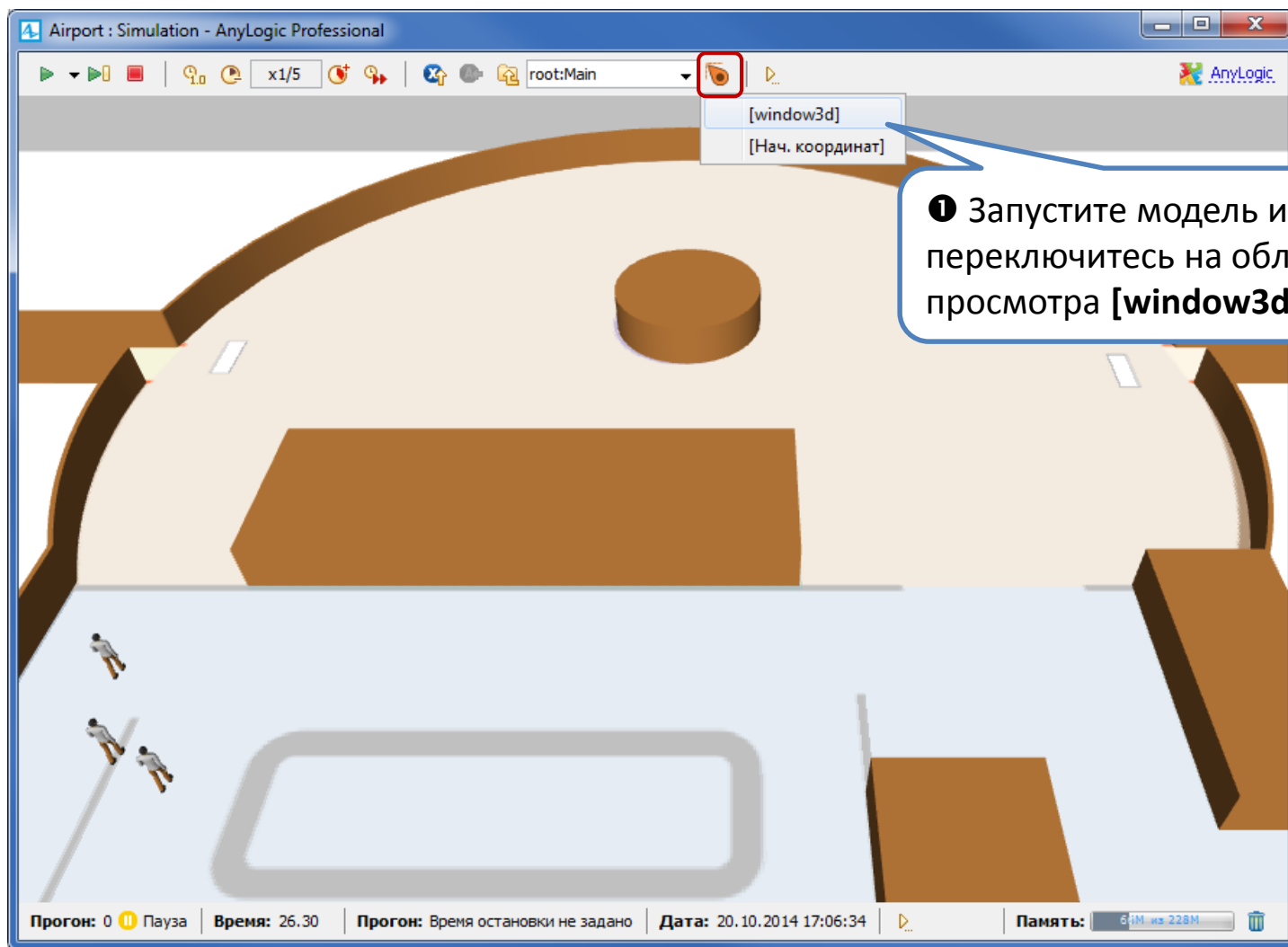
Аэропорт. Фаза 2. Шаг 3



❶ Выберите *Passenger* как тип пешеходов, которые создает блок *pedSource*.



Аэропорт. Фаза 2. Шаг 4



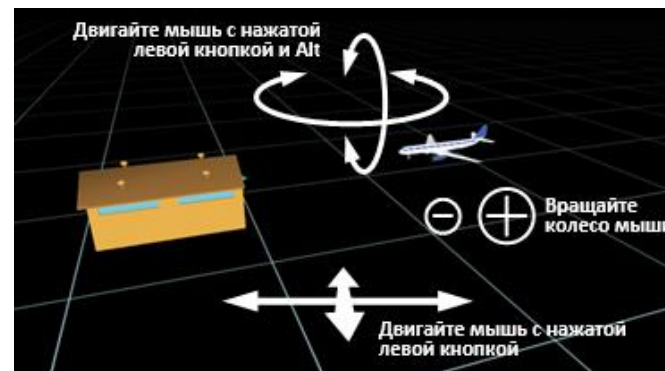
Чтобы создать 3D анимацию для моделируемого процесса, мы добавили элемент **3D Окно** из палитры **Презентация**.

- 1 Теперь мы можем запустить модель и наблюдать процесс в 3D анимации. Переключитесь на сцену 3D анимации с помощью области просмотра **[window3d]**, созданной автоматически.

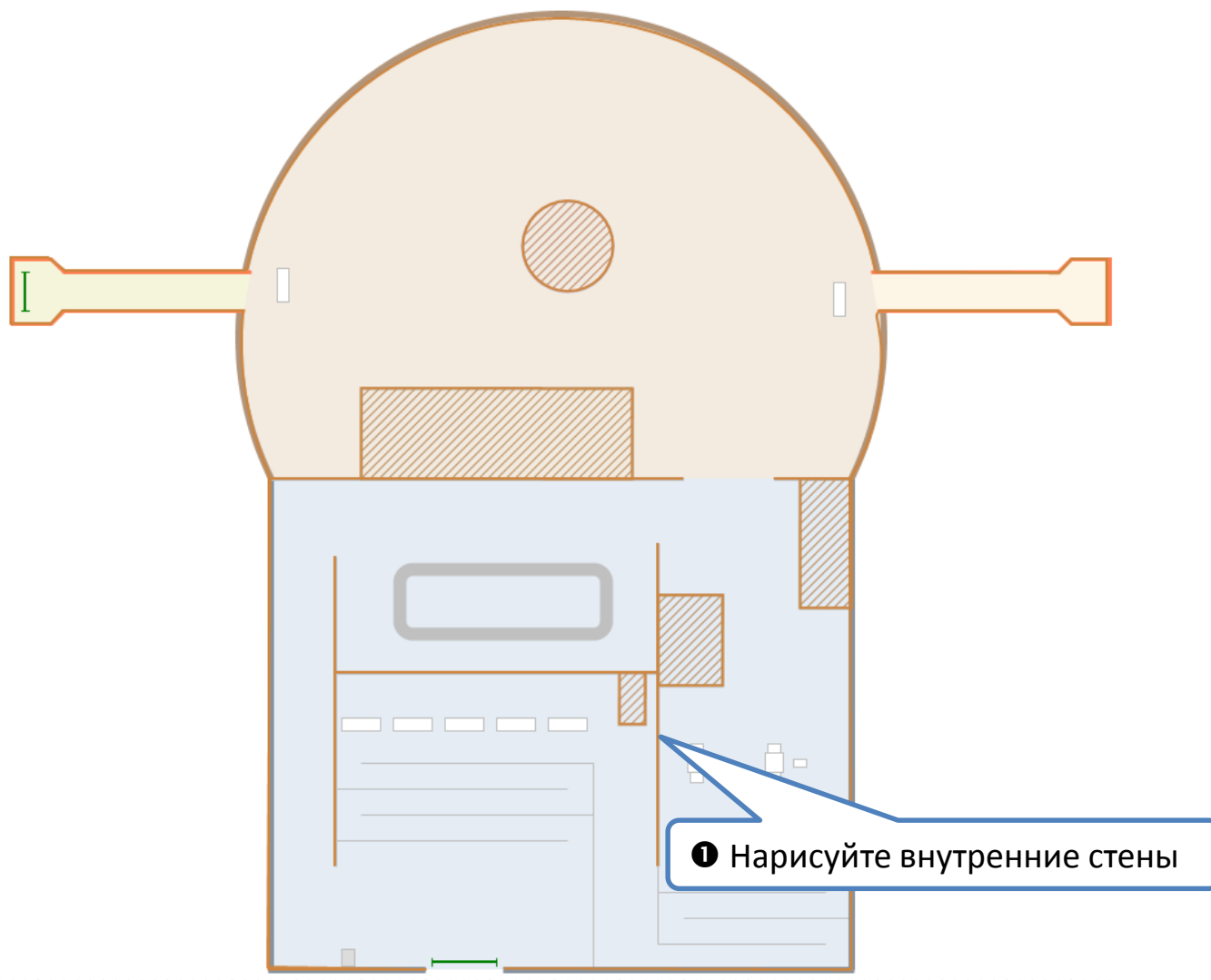
Вы увидите, как пассажиры движутся по аэропорту от входа к заданному гейту.

Навигация по 3D сцене

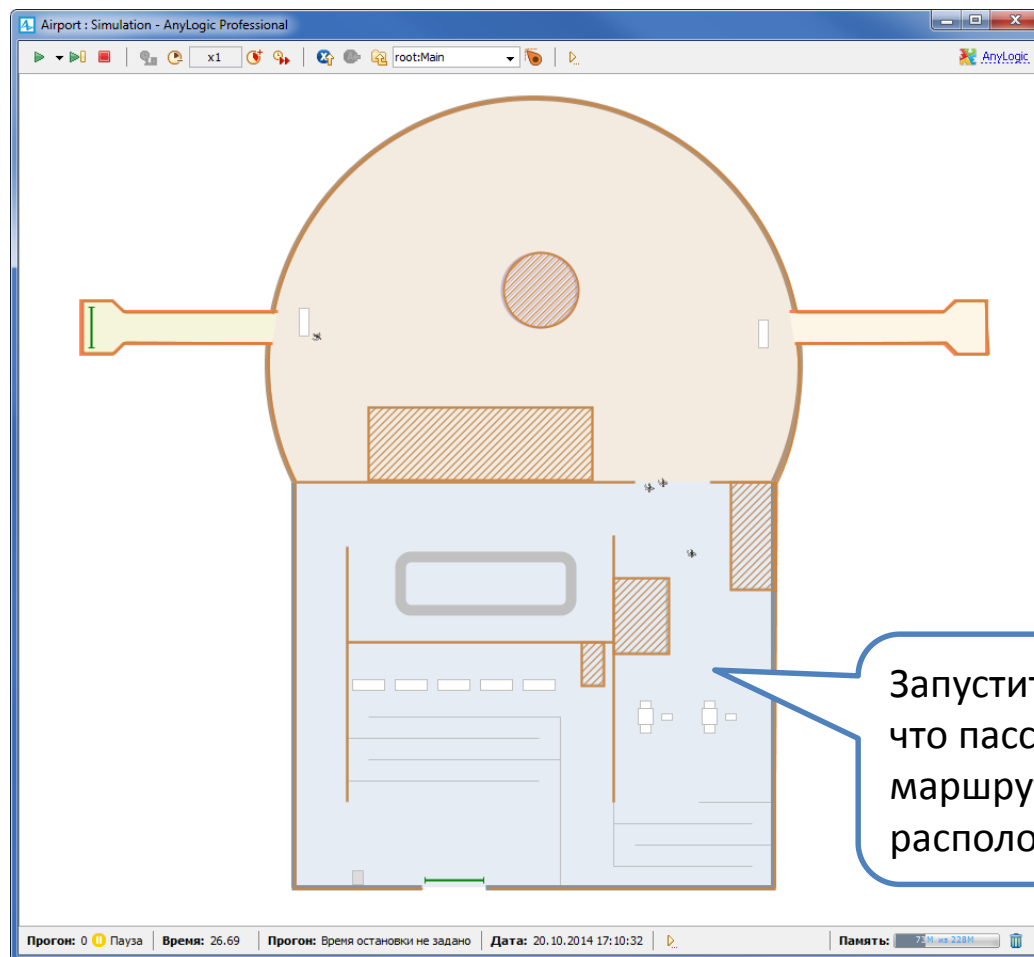
- **Перетаскиванием мыши с нажатой правой кнопкой** Вы перемещаете камеру вправо, влево вверх или вниз на той же высоте.
- **Вращением колесика мыши** Вы перемещаете камеру ближе или дальше от текущего центра сцены.
- **Перетаскиванием мыши с нажатой левой кнопкой и клавишей Alt** Вы вращаете сцену относительно камеры.



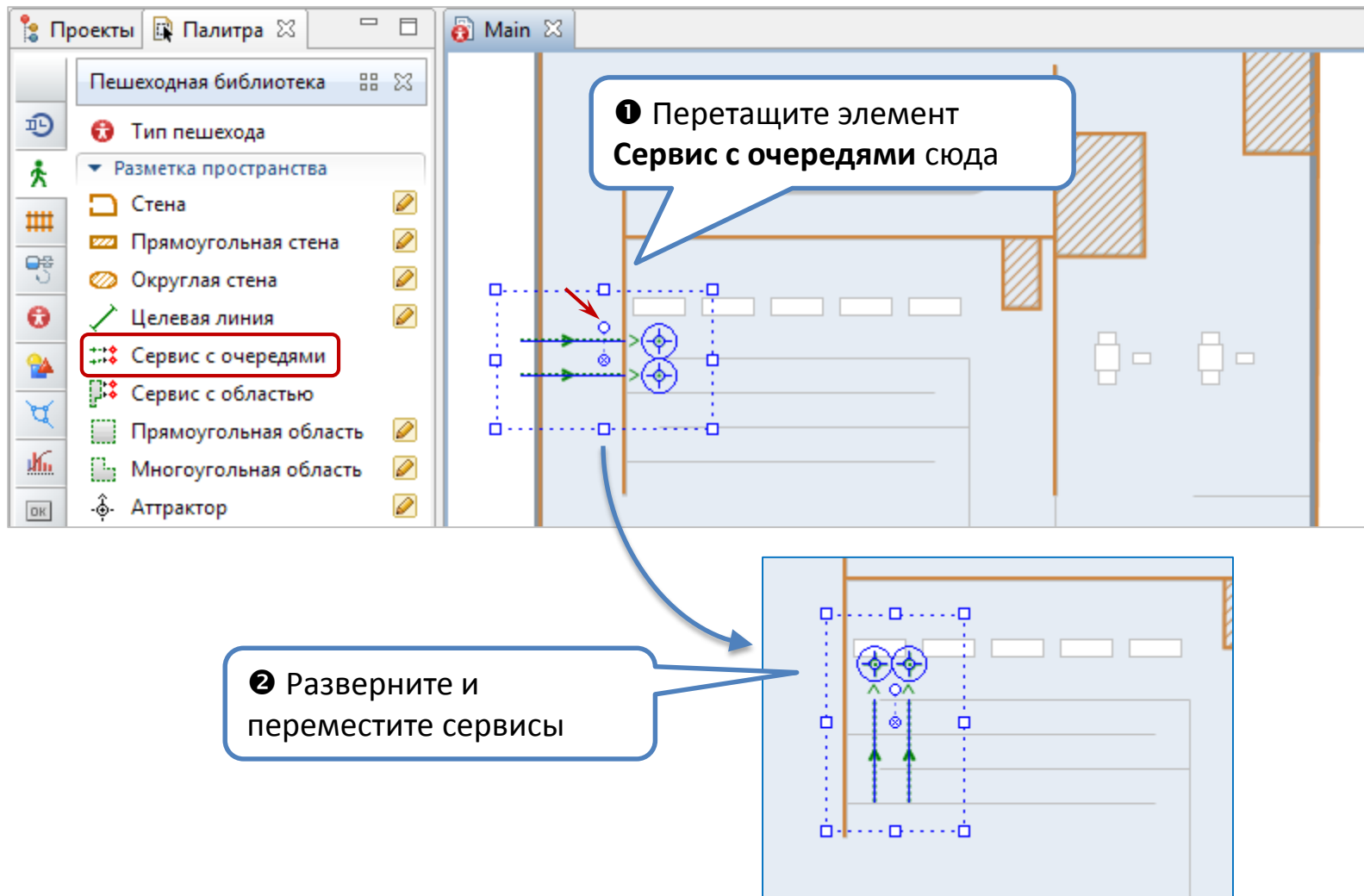
Аэропорт. Фаза 2. Шаг 5



Давайте зададим внутренние стены. Используйте тот же элемент разметки пространства **Стена**, как и раньше. Пожалуйста, убедитесь, что между отрезками стен нет промежутков.




Аэропорт. Фаза 2. Шаг 6




Теперь мы можем начать моделировать процессы происходящего в аэропорту. Давайте сначала смоделируем стойки регистрации на рейс. В действительности это сервис: пассажиры должны там обслуживаться, и если сервис занят пассажиром, другие ждут в очереди, пока сервис освободится.

Сервисы в пешеходных моделях

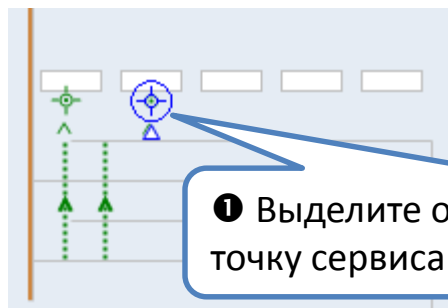
 **Сервис с очередями** – Используется, чтобы задавать сервис(ы) с линией очереди (например, турникеты, касса, паспортный контроль).



 **Сервис с областью** – Для задания сервисов с электронной очередью, когда пешеходы ждут в располагающейся рядом области, пока сервис не станет доступен (например, билетная касса, отделение банка, стойка информации).



Аэропорт. Фаза 2. Шаг 7



❶ Выделите отдельным щелчком эту точку сервиса и переместите ее сюда

Свойства ✕

checkInServices - Сервис с очередями

Имя:

☐ Исключить ☒ Отображается на верхнем уровне

☐ Блокировать

Видимость: ☒ да

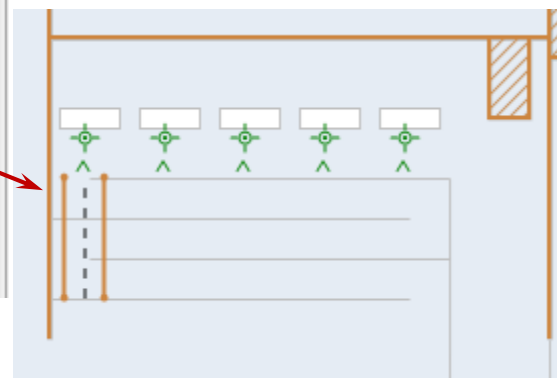
Этаж:

Кол-во сервисов:

Кол-во очередей:

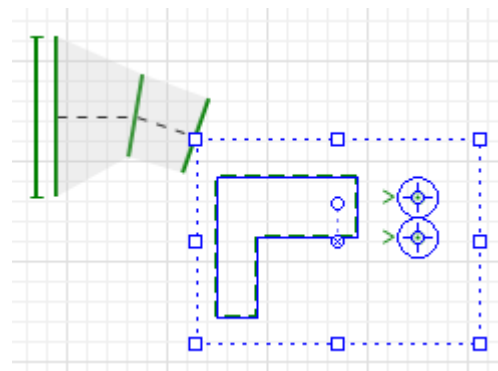
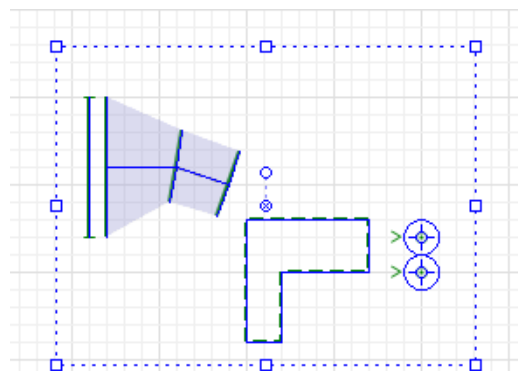
Тип очереди: ☐ Линия ☒ Змейка

Тип сервиса: ☒ Точечный ☐ Линейный



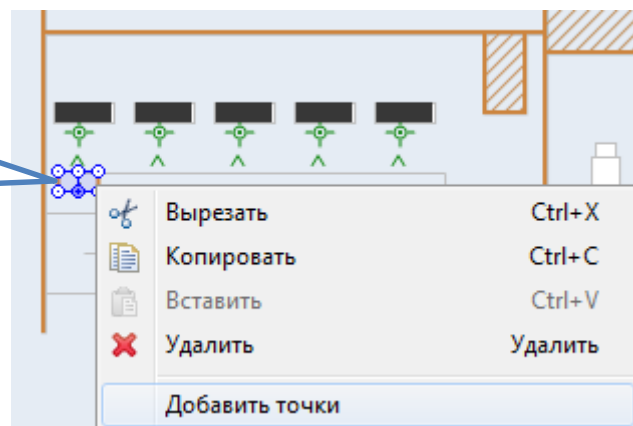
При работе со сложными элементами разметки

- Так как некоторые элементы разметки пространства могут состоять из нескольких примитивных фигур и все пешеходные фигуры разметки составляют этаж(и), важно понимать, как выделять фигуры разметки так, чтобы Вам было удобно редактировать их графически.

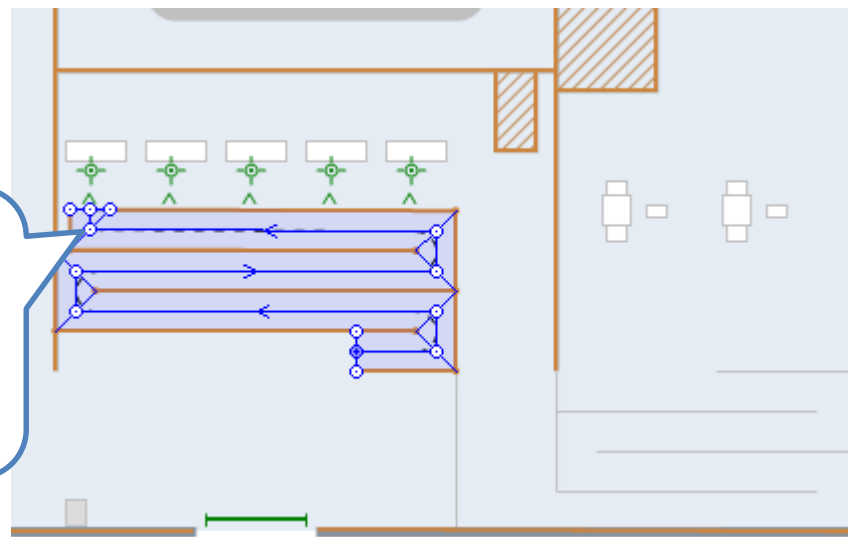


Аэропорт. Фаза 2. Шаг 8

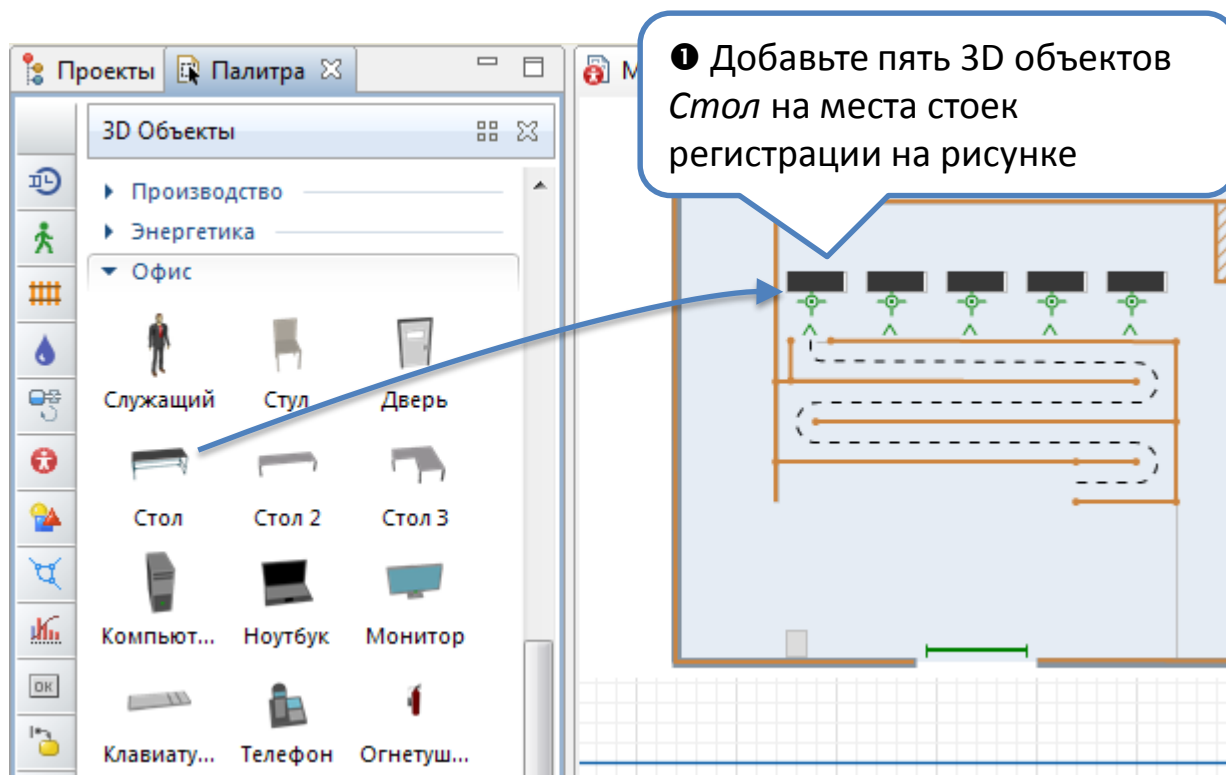
❶ Уменьшите размер линии очереди, затем щелкните ее правой кнопкой мыши и выберите **Добавить точки**



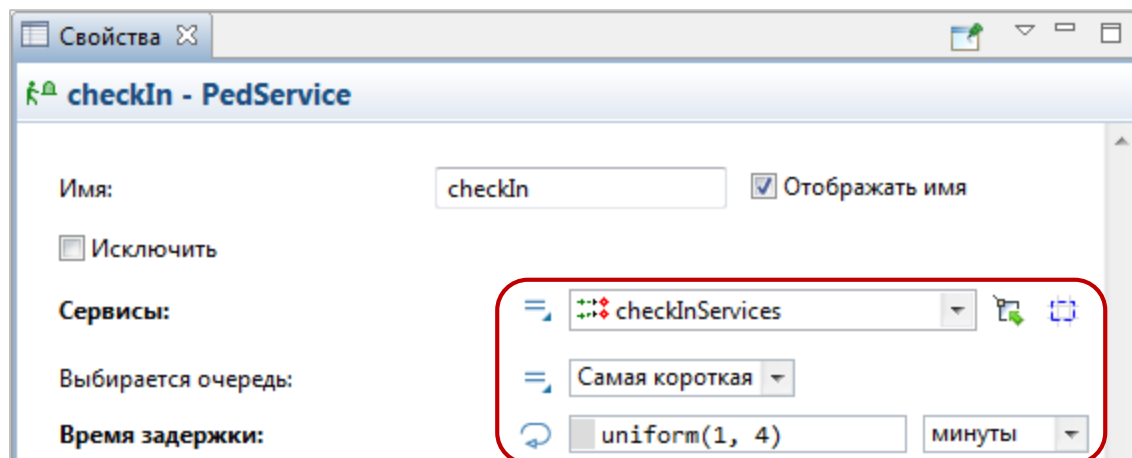
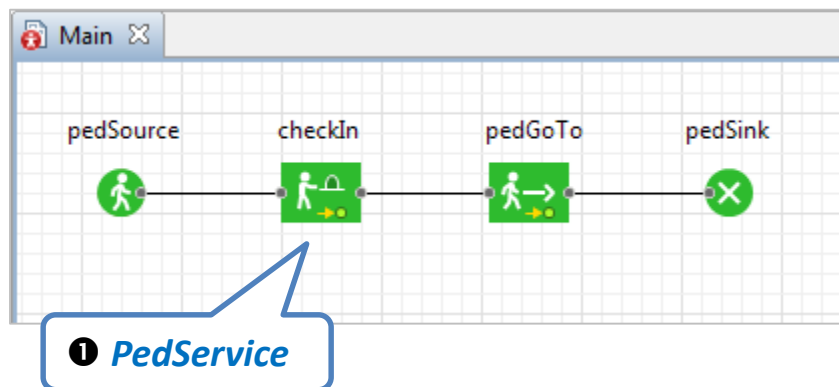
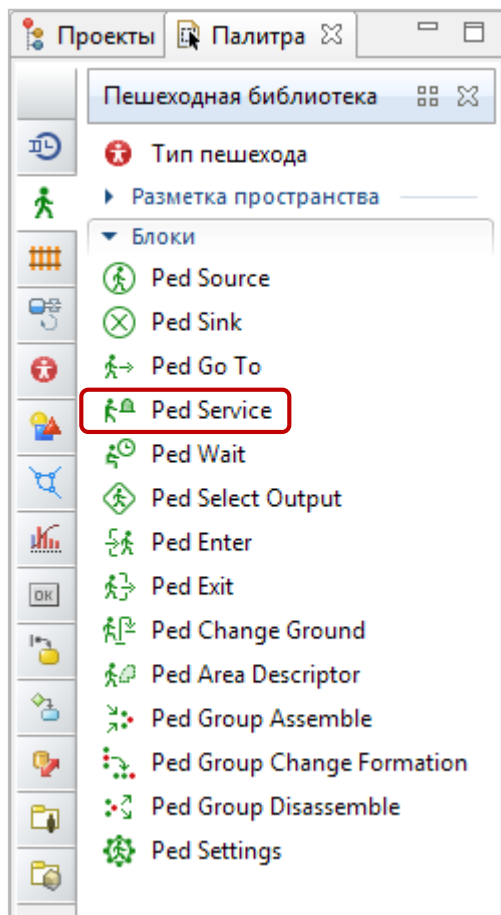
❷ Нарисуйте очередь указанной формы. Добавляйте точки поворота очереди, щелкая мышью посередине прохода. Последнюю точку добавьте двойным щелчком.



Аэропорт. Фаза 2. Шаг 9



Аэропорт. Фаза 2. Шаг 10

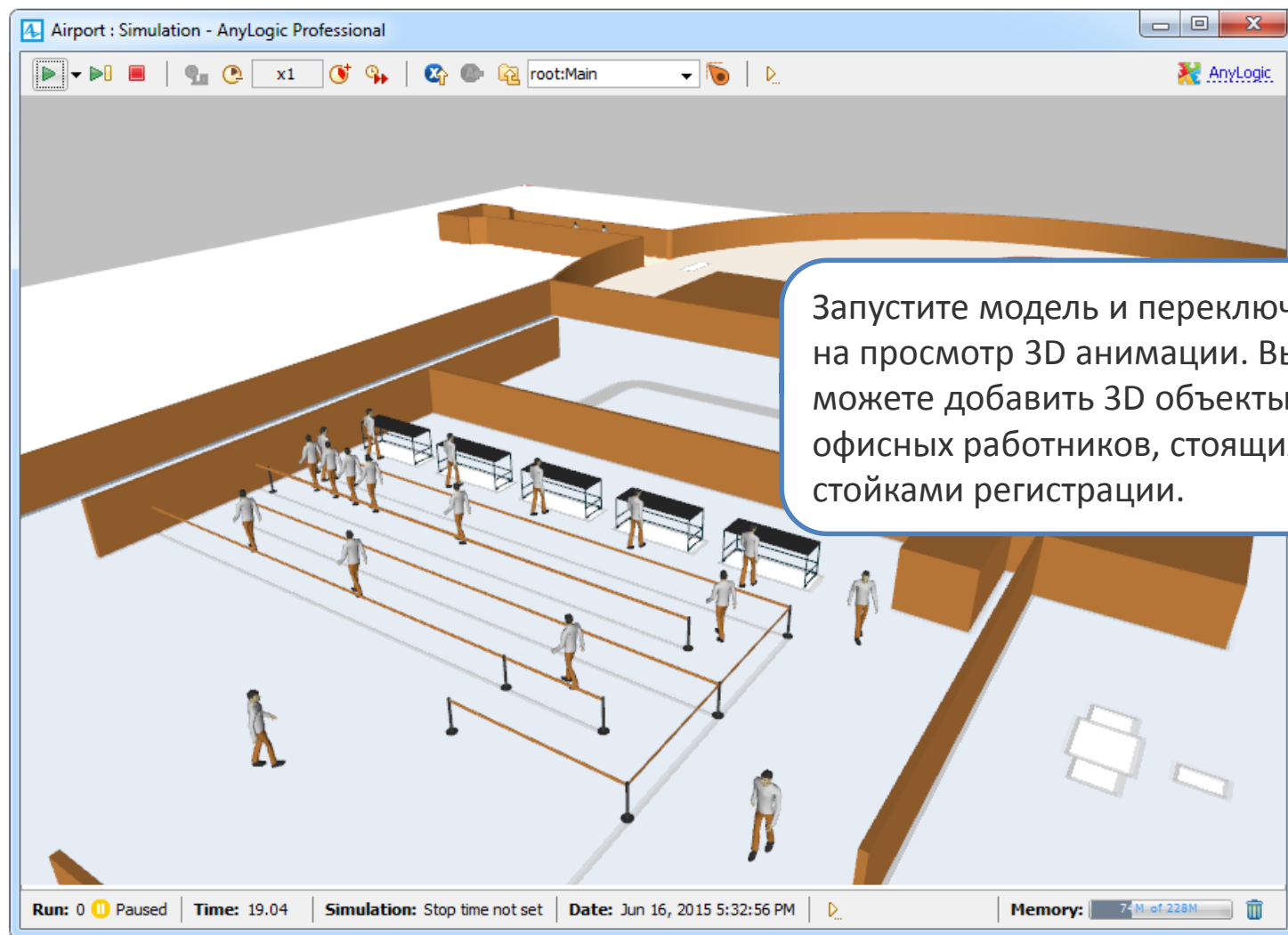


Теперь давайте добавим блок, моделирующий процесс регистрации пассажиров на рейс. Поместите блок *PedService* на диаграмму процесса, чтобы заставить пешеходов проходить через сервис, заданный указанными фигурами разметки.

Соедините блоки, как показано на рисунке выше.



Аэропорт. Фаза 2. Шаг 11

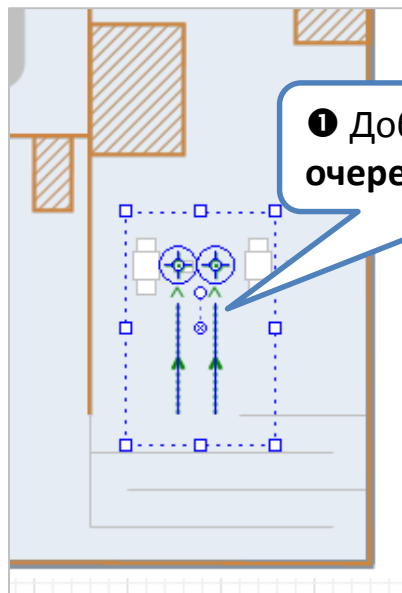


Аэропорт. Фаза 3

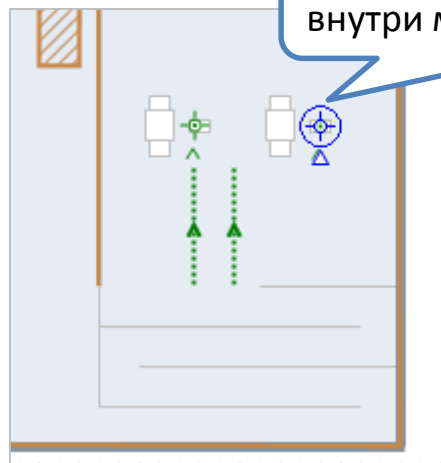
- В этой фазе мы добавим пункты досмотра.
- Пройдя досмотр, пассажиры ждут в области ожидания и через какое-то время проходят на посадку.



Аэропорт. Фаза 3. Шаг 1

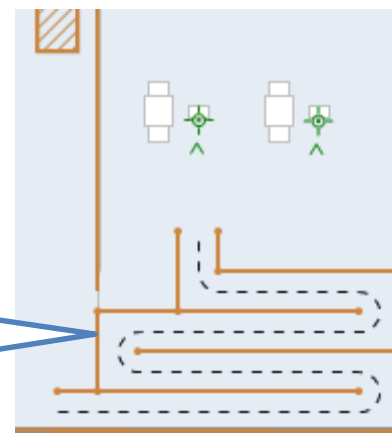


❶ Добавьте Сервис с очередями

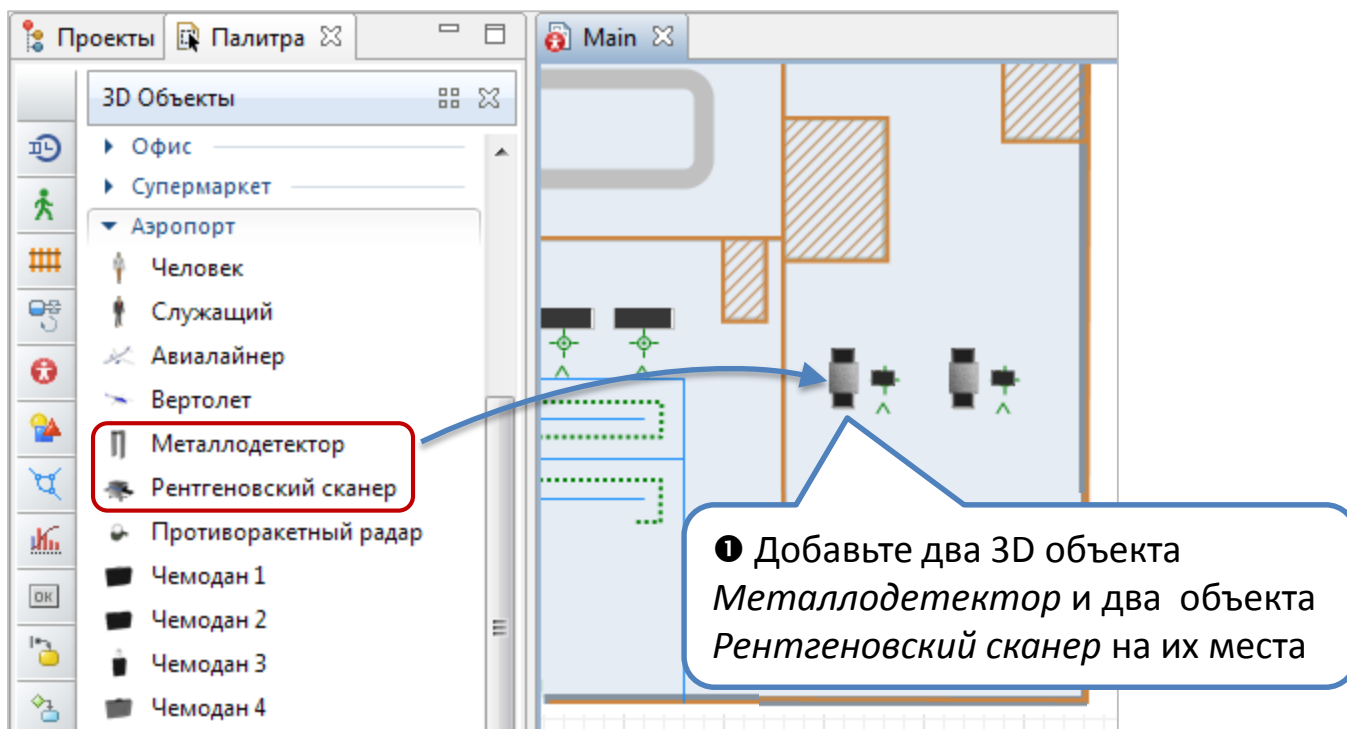


❷ Расположите точки сервиса внутри металлодетекторов

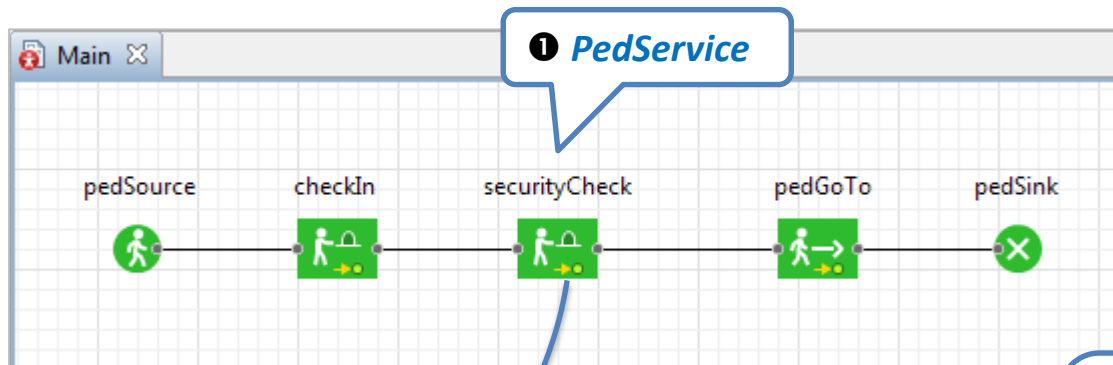
❸ Нарисуйте одну очередь-змейку такой формы



Аэропорт. Фаза 3. Шаг 2



Аэропорт. Фаза 3. Шаг 3



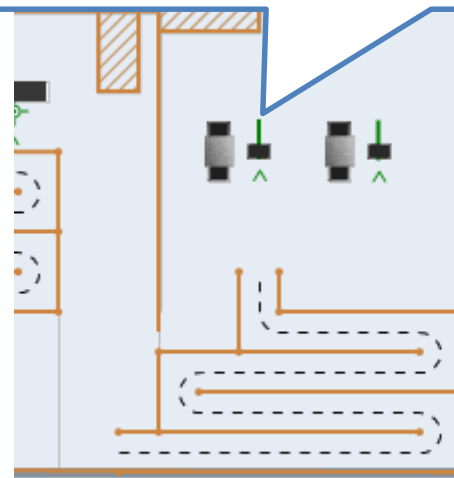
① **PedService**

The 'Свойства' (Properties) window for the 'securityCheck - PedService' service is shown. It contains the following fields and options:

- Имя: securityCheck ☒ Отображать имя
- ☐ Исключить
- Сервисы: securityCheckServices
- Выбирается очередь: Самая короткая
- Время задержки: uniform(15, 120) секунды
- Задержка на восстановление: 0.0 секунды

② Установите **Тип сервиса:**

Линейный и расположите линии сервиса так, чтобы они проходили через металлодетекторы



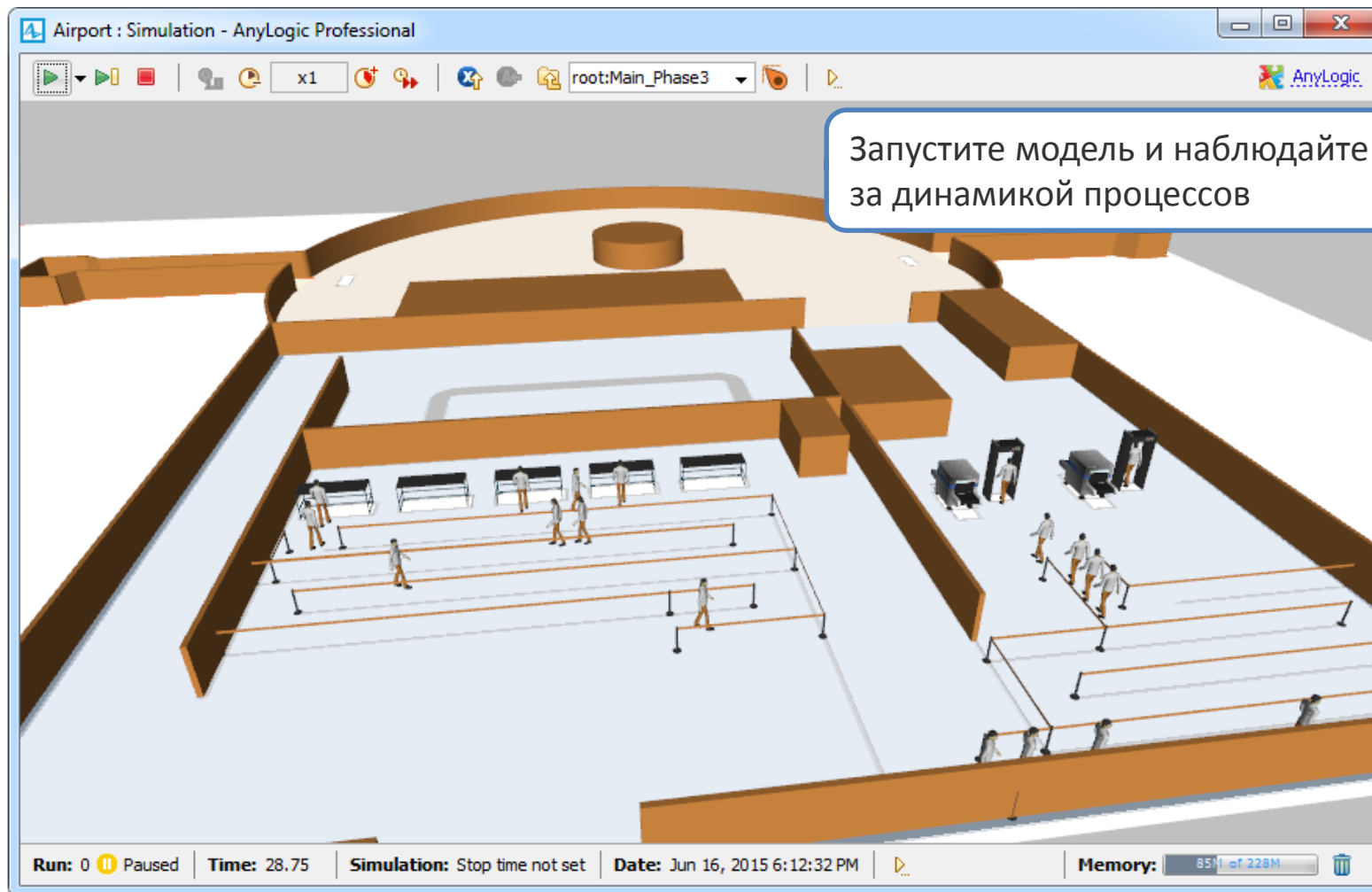
Точечные и линейные сервисы

Существует два типа сервисов: *Линейные* и *Точечные*. Мы использовали точечные сервисы, чтобы задать сервис регистрации, теперь нам нужны линейные сервисы, чтобы задать прохождение людей через металлодетектор при досмотре.

- *Линейный* сервис задается линией, вдоль которой должны передвигаться пешеходы. Пешеход начинает процедуру обслуживания в начальной точке линии и затем продвигается к ее конечной точке, из которой покидает сервис. Фаза восстановления сервиса начинается, когда пешеход заканчивает процедуру обслуживания (если ожидание выхода пешехода из сервиса отключено) или проходит конечную точку линии (если ожидание выхода пешехода из сервиса включено). После восстановления сервис становится свободным и готов принимать новых пешеходов.
- *Точечные* сервисы задаются точкой, на которой должен находиться пешеход в течение времени обслуживания.



Аэропорт. Фаза 3. Шаг 4



Аэропорт. Фаза 3. Шаг 5

1 Нарисуйте область ожидания возле гейта с помощью элемента **Многоугольная область**

2 **PedWait**

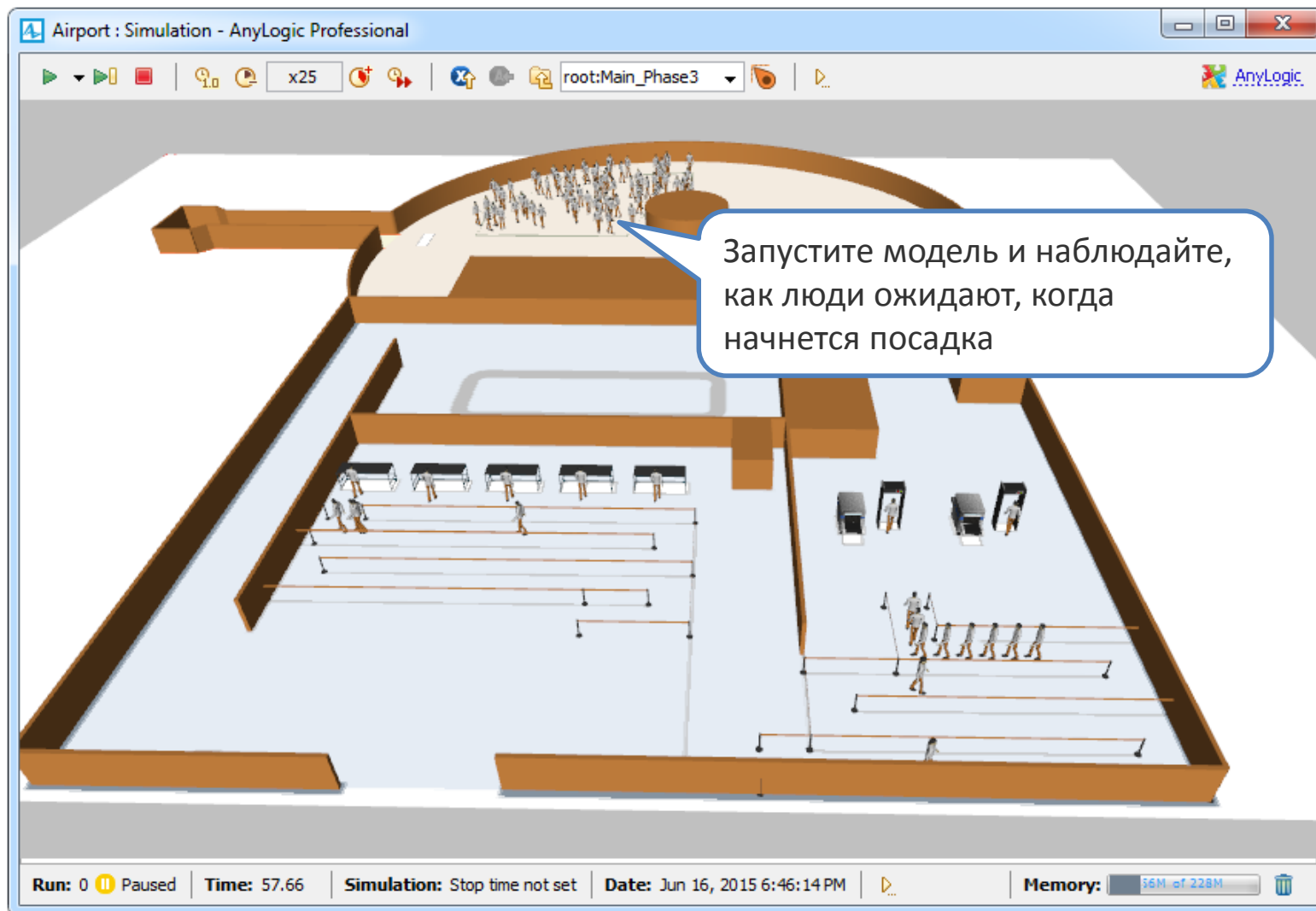
securityCheck → pedWait → pedGoTo → pedSink

Свойства pedWait - PedWait

- Имя: pedWait ☒ Отображать имя
- ☐ Исключить
- Место ожидания: ☒ область ☐ линия ☐ точка (x, y)
- Область: area
- Использовать аттрактор: Нет
- Ожидание заканчивается: ☒ По истечении заданного времени ☐ По вызову функции free()
- Время задержки: 2 часы



Аэропорт. Фаза 3. Шаг 6





Аэропорт. Фаза 4

- В этой фазе мы покажем, как извлечь выгоду из полноценной интеграции всех методов моделирования.
- Мы смоделируем то, как инфекция распространяется по аэропорту.
- Время от времени зараженные люди будут входить в аэропорт, и в случае длинных очередей они будут заражать тех пассажиров, которые будут стоять с ними рядом достаточно долго.



Аэропорт. Фаза 4. Шаг 1

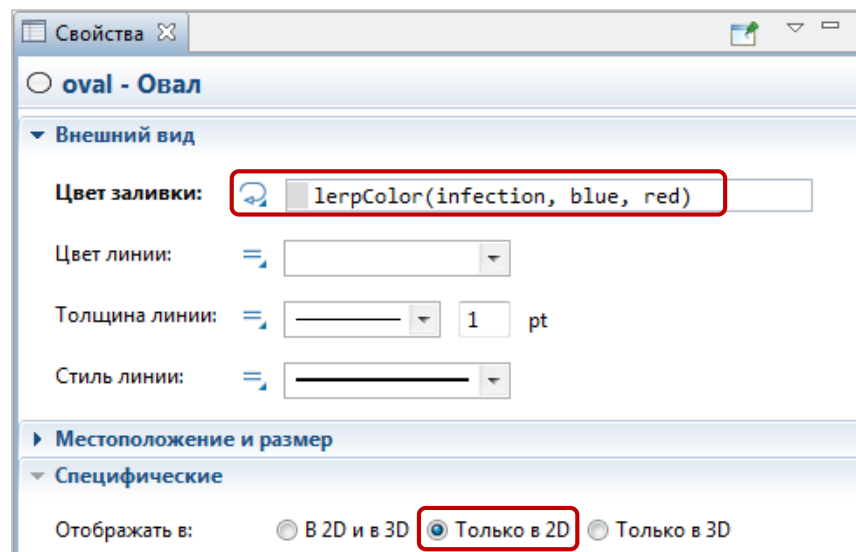
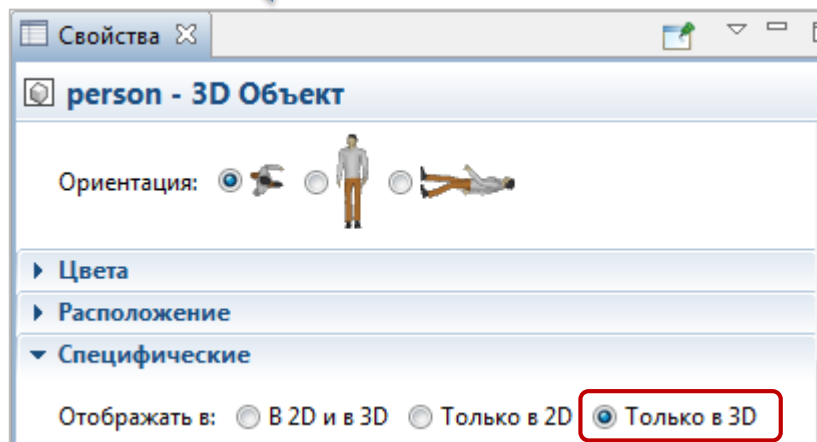
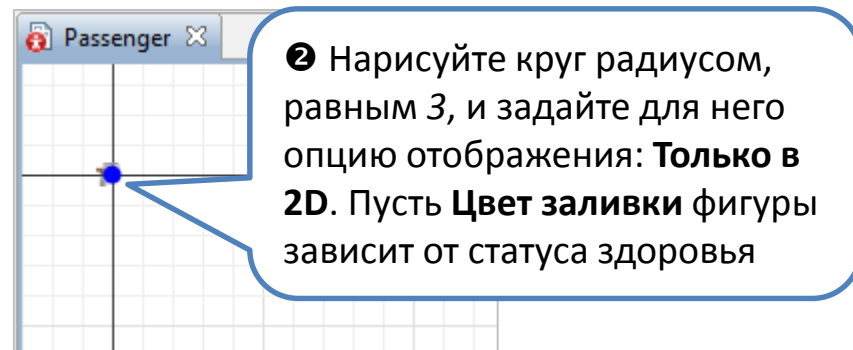
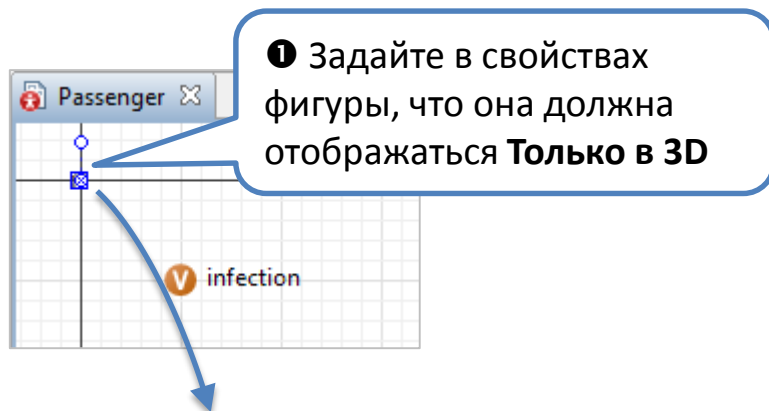
The image shows a screenshot of the AnyLogic software interface. At the top, there are two tabs: 'Main' and 'Passenger'. The 'Passenger' tab is active, showing a diagram of an agent type. A variable named 'infection' is highlighted in the diagram. A callout box points to this variable with the text: '1 Откройте диаграмму типа агента *Passenger*, дважды щелкнув по ней в дереве панели **Проекты**'. Below the diagram, the 'Свойства' (Properties) window for the 'infection' variable is open. It shows the following settings: 'Имя:' (Name) is 'infection', 'Отображать имя' (Show name) is checked, 'Исключить' (Exclude) is unchecked, 'Видимость:' (Visibility) is set to 'да' (yes), 'Тип:' (Type) is 'double', and 'Начальное значение:' (Initial value) is 'randomTrue(0.05) ? 1 : 0'. A callout box points to the 'Начальное значение:' field with the text: '2 Добавьте переменную, которая будет определять статус здоровья'. The 'Свойства' window also has expandable sections for 'Специфические' (Specific) and 'Описание' (Description).

1 Откройте диаграмму типа агента *Passenger*, дважды щелкнув по ней в дереве панели **Проекты**

2 Добавьте переменную, которая будет определять статус здоровья

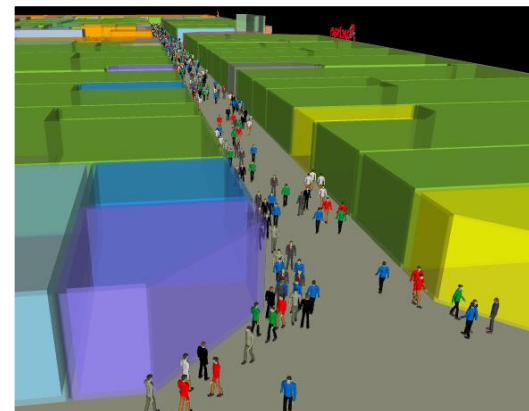
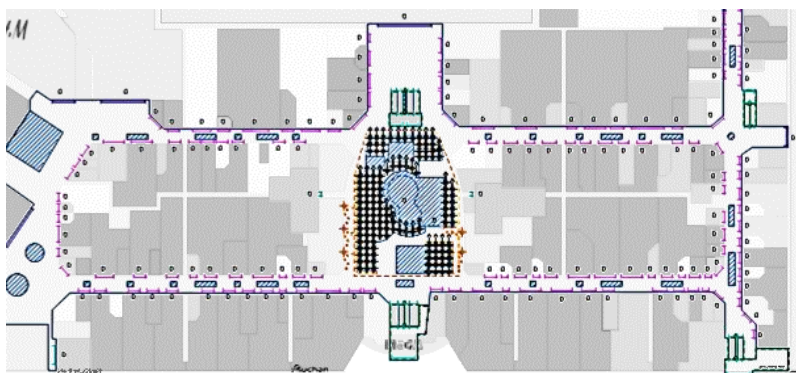


Аэропорт. Фаза 4. Шаг 2

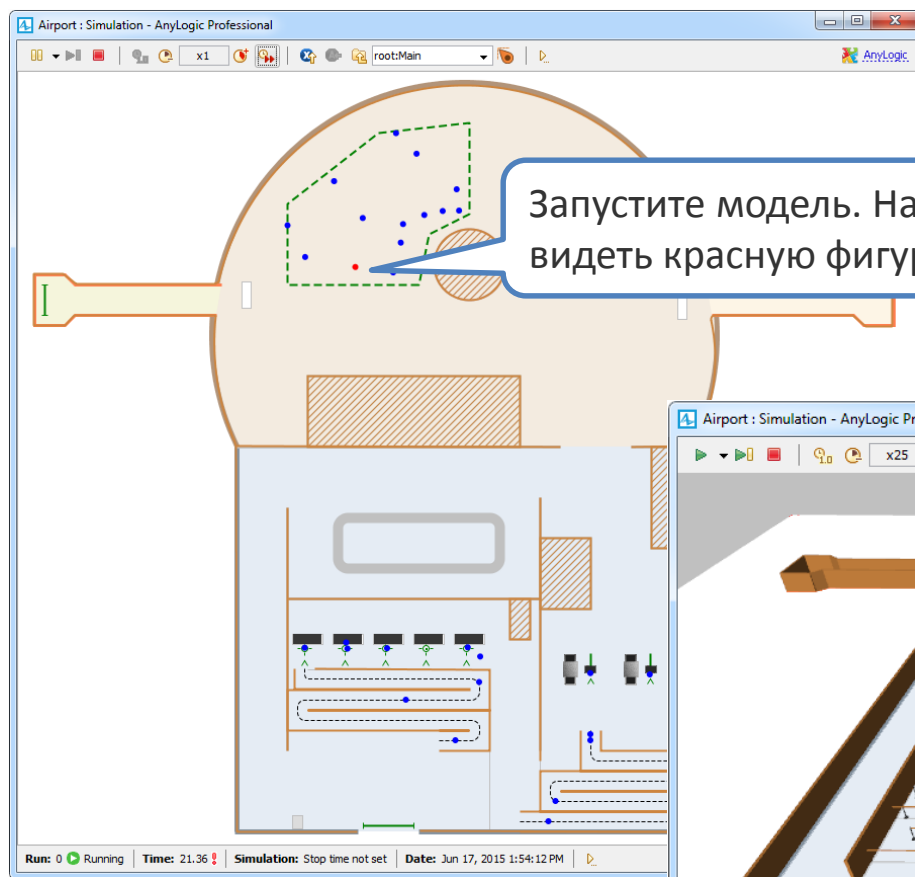


Различные 2D и 3D фигуры анимации

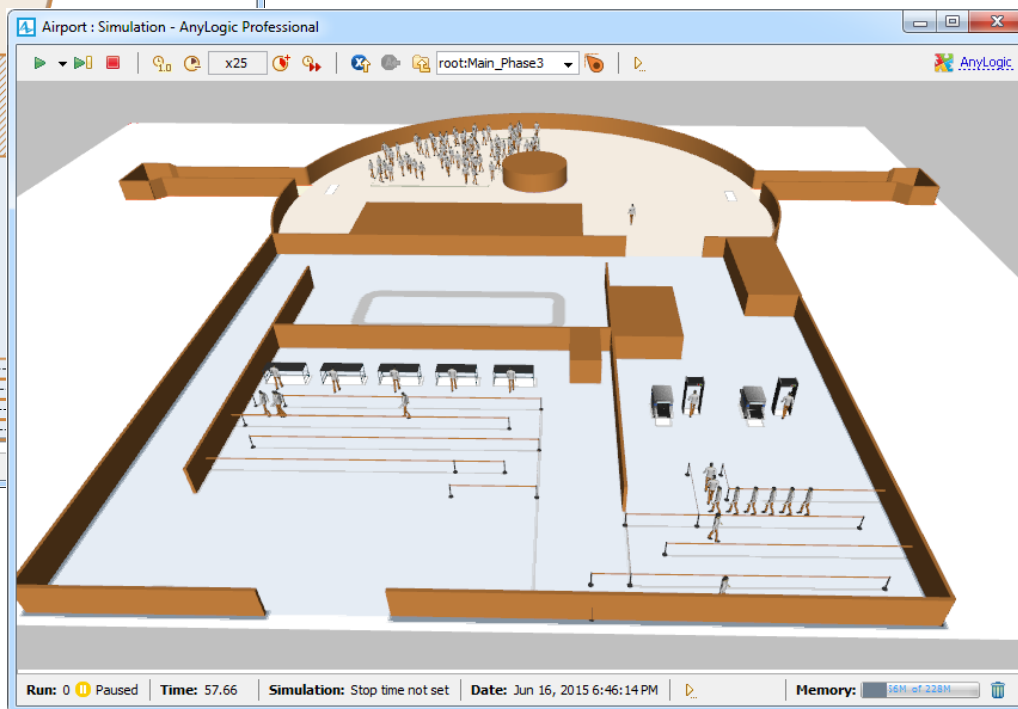
- По умолчанию все фигуры отображаются и на 2D, и на 3D анимации.
- Иногда Вам может понадобиться добавить для одного и того же объекта разные фигуры 2D и 3D анимации.
- Вы можете нарисовать две анимации и установить отображение одной фигуры только в 2D, а другой – только в 3D (в секции свойств **Специфические**, параметр **Отображать в**).



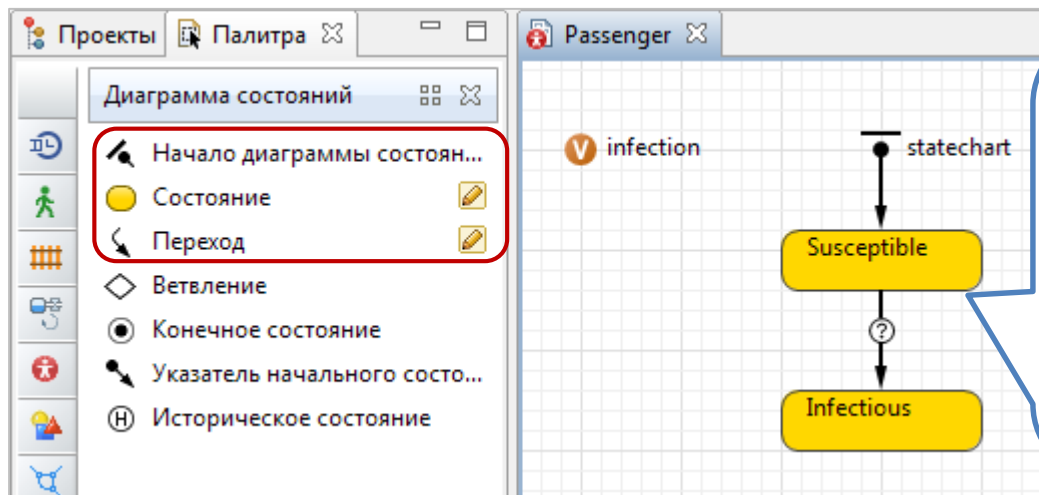
Аэропорт. Фаза 4. Шаг 3



Запустите модель. На 2D анимации Вы можете видеть красную фигуру зараженного пассажира.



Аэропорт. Фаза 4. Шаг 4



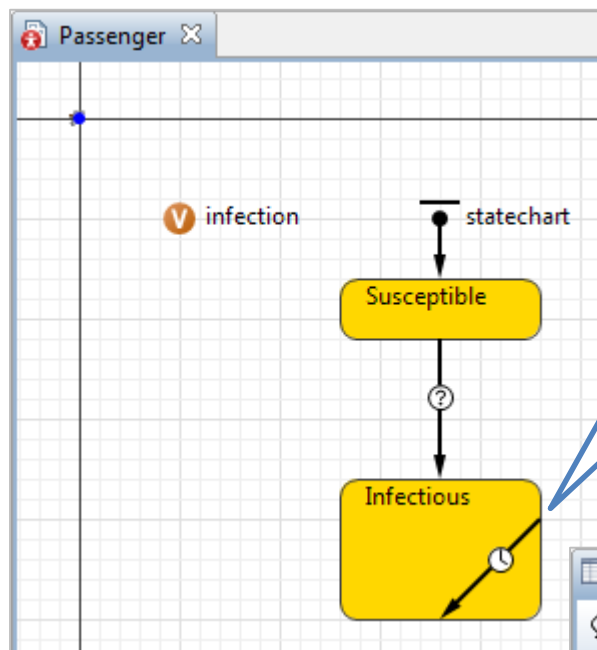
❶ Нарисуйте диаграмму состояний, состоящую из двух состояний: *Susceptible* («восприимчивый к заражению», помеченное как первичное элементом **Начало диаграммы состояний**), и *Infectious* («зараженный»).

❷ Переход приводит состояние человека к состоянию *Infectious* («зараженный»), когда выполняется заданное условие

The screenshot shows the 'Свойства' (Properties) window for a transition, titled 'transition - Переход'. It contains several fields and checkboxes. The 'Имя:' (Name) field is set to 'transition'. There are two checkboxes: 'Отображать имя' (Show name) and 'Исключить' (Exclude). The 'Происходит:' (Occurs) dropdown menu is set to 'При выполнении условия' (When condition is met). The 'Условие:' (Condition) field is set to 'infection > 0.5'. The 'Действие:' (Action) and 'Доп. условие:' (Additional condition) fields are empty.



Аэропорт. Фаза 4. Шаг 5



❶ Добавьте внутренний переход *infectOthers* внутри состояния *Infectious*, который будет имитировать то, как инфицированный человек постепенно заражает окружающих его людей.

Свойства

infectOthers - Переход

Имя: ☐ Отображать имя

☐ Исключить

Происходит:

Таймаут:

Действие:

```
for( Passenger p : agentsInRange(30) )
    send( "Infection", p );
```

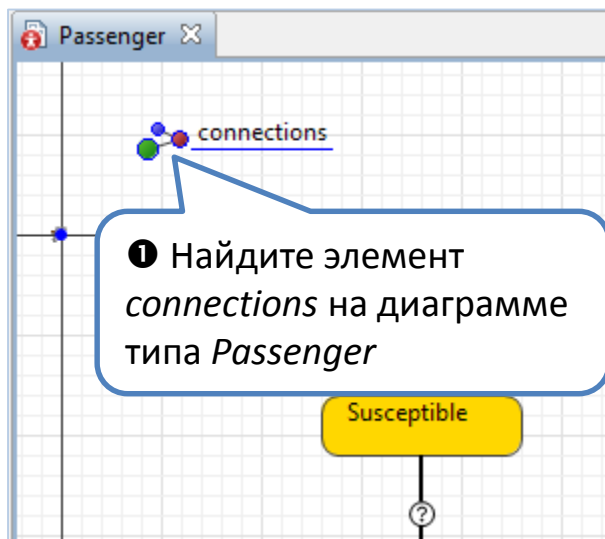


Цикл `for`. Форма “Итератор по коллекции”

- Здесь мы используем термин “итератор по коллекции” для обозначения одной из форм цикла `for`. Эта форма используется для итерации по элементам коллекции или агентам популяции. Если вам нужно произвести какие-то действия с каждым агентом популяции, мы рекомендуем использовать эту форму цикла, поскольку она более компактна и имеет более простой синтаксис, а также поддерживается всеми типами коллекций (в отличие от итераций с использованием счетчика).



Аэропорт. Фаза 4. Шаг 6



❶ Найдите элемент *connections* на диаграмме типа *Passenger*

❷ Задайте, как связанные агенты будут реагировать при получении сообщения

Свойства

connections - Связь с агентами

Имя: ☒ Отображать имя ☐ Исключить

Видимость: ☒ да

Это стандартная связь соединения агентов

Тип агента:

☒ Набор ссылок
☐ Одиночная ссылка

Эта стандартная связь является двунаправленной

Взаимодействие

Эти действия выполняются для сообщений, поступающих во все связи агента

Тип сообщения:

Действие при получении сообщения:

Перенаправлять сообщение в:

Диаграммы состояний	
<input checked="" type="checkbox"/> statechart	



Аэропорт. Фаза 4. Шаг 7

