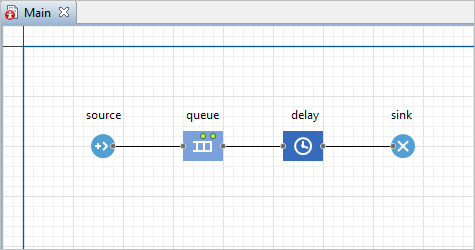
**Лабораторная работа №4**

**«Системы массового обслуживания»**

**Основная часть**

**Создание простой модели СМО**

Вначале мы создадим простейшую модель, в которой будем рассматривать обслуживание людей автоматом обслуживания. Зададим динамику процесса, создав диаграмму из блоков **Библиотеки моделирования процессов**. Добавьте блоки **Библиотеки моделирования процессов** на диаграмму и соедините их, как показано на рисунке.



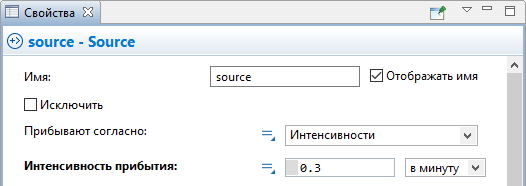
Данная схема моделирует простейшую систему очереди, состоящую из источника (генератора) агентов, задержки (и очереди перед задержкой) и финального уничтожения агентов.

Скажем пару слов об этих объектах диаграммы.

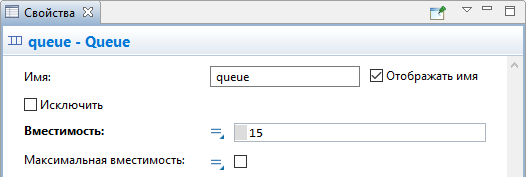
* http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ELT/images/source16.pngОбъект [**Source**](http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ProcessModeling/Source.html) генерирует агентов определенного типа. Обычно он используется в качестве начальной точки диаграммы процесса, формализующей поток агентов. В нашем примере агентами будут посетители банка, а объект **Source** будет моделировать их приход в банковское отделение.
* http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ELT/images/queue16.pngОбъект [**Queue**](http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ProcessModeling/Queue.html) моделирует очередь агентов, ожидающих приема объектами, следующими за данным в диаграмме процесса. В нашем случае он будет моделировать очередь клиентов, ждущих освобождения банкомата.
* http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ELT/images/delay16.pngОбъект [**Delay**](http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ProcessModeling/Delay.html) задерживает агентов на заданный период времени, представляя в нашей модели банкомат, у которого посетитель банковского отделения тратит свое время на проведение необходимой ему операции.
* http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ELT/images/sink16.pngОбъект [**Sink**](http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ProcessModeling/Sink.html) уничтожает поступивших агентов. Обычно он используется в качестве конечной точки потока агентов (и диаграммы процесса соответственно).

***Настройте блоки диаграммы***

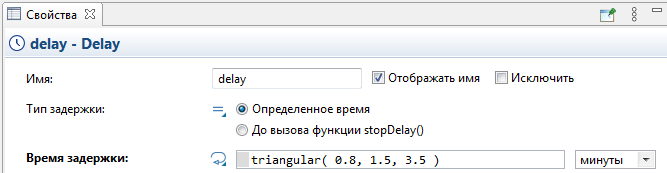
Выделите блок *source*. В панели **Свойства** укажите, как часто должны прибывать клиенты. Введите *0.3* и выберите *в минуту* в поле **Интенсивность прибытия**.



Измените свойства блока *queue*. Введите в поле **Вместимость** *15*. Что означает, что в очереди будут находиться не более 15 человек.



Измените свойства блока *delay*. Задайте время обслуживания в поле **Время задержки**, распределенное по треугольному закону со средним значением, равным *1.5*, минимальным - равным *0.8* и максимальным - *3.5* *минутам*.



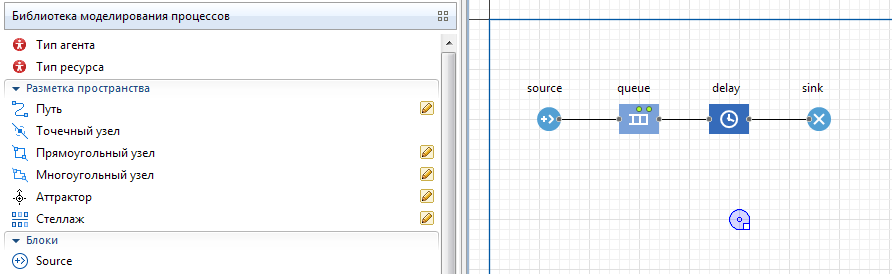
**2. Создание анимации модели**

Хотя и возможно анализировать работу модели с помощью диаграммы процесса, но куда удобнее было бы иметь более наглядную анимацию моделируемого процесса. В этом примере мы хотим создать визуализированный план банковского отделения.

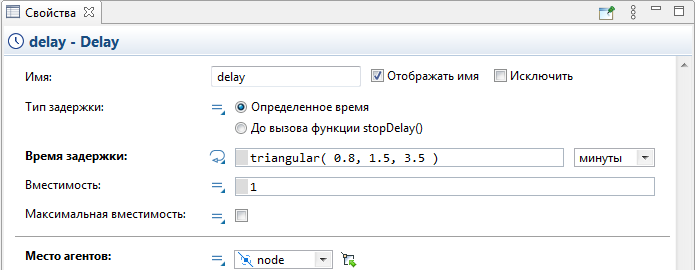
Поскольку в данном случае нас не интересует конкретное расположение объектов в пространстве, то можно просто добавить чисто схематическую анимацию интересующих объектов - в нашем случае, мы хотим видеть на анимации автомат обслуживания и ведущую к нему очередь клиентов. Анимация модели рисуется в той же диаграмме (в графическом редакторе), в которой задается и диаграмма моделируемого процесса.

***Добавление фигур разметки пространства***

1. Нарисуем точечный узел, обозначающий автомат обслуживания. Вначале откройте палитру **Разметка пространства** панели **Палитра**.
2. Перетащите элемент **Точечный узел** из палитры **Разметка пространства** в графический редактор и поместите его под блок-схемой процесса.

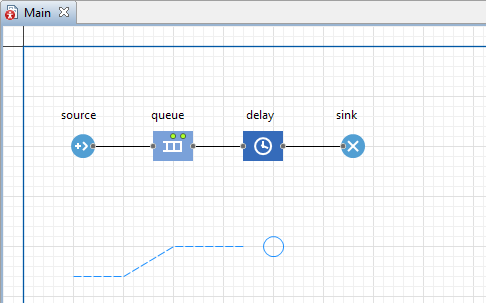


1. Выделите щелчком блок *delay* в диаграмме процесса, чтобы открыть его свойства.
2. Выберите точечный узел *node*, который мы нарисовали в параметре **Место агентов**. Вы можете выбрать его из выпадающего списка подходящих объектов, щелкнув стрелку "вниз", или выбрать фигуру из графического редактора, предварительно щелкнув кнопку справа от параметра (в таком случае все неподходящие объекты в графическом редакторе будут обесцвечены).

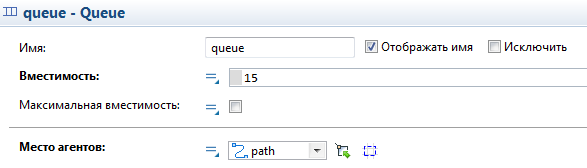


***Задание фигуры анимации очереди к банкомату***

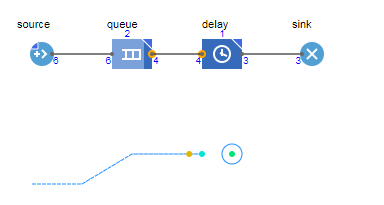
1. Нарисуем путь, обозначающий очередь к банкомату. Вначале откройте палитру **Разметка пространства** панели **Палитра**.
2. Двойным щелчком выделите элемент **Путь**  палитры **Разметка пространства**, чтобы перейти в *режим рисования*.
3. Теперь вы можете рисовать путь точка за точкой, последовательно щелкая мышью в тех точках диаграммы, куда вы хотите поместить вершины линии. Чтобы завершить рисование, добавьте последнюю точку пути двойным щелчком мыши.



1. Выделите щелчком блок *queue* в диаграмме процесса, чтобы открыть для него панель **Свойства**.
2. Выберите путь *path*, который мы только нарисовали в параметре **Место агентов**. Вы можете выбрать его из выпадающего списка подходящих объектов, щелкнув стрелку "вниз", или выбрать фигуру из графического редактора, предварительно щелкнув кнопку справа от параметра (в таком случае все неподходящие объекты в графическом редакторе будут обесцвечены).



Теперь вы можете запустить модель и изучить ее поведение. Вы увидите, что у нашей модели теперь есть простейшая анимация – автомат обслуживания и ведущая к нему очередь клиентов.

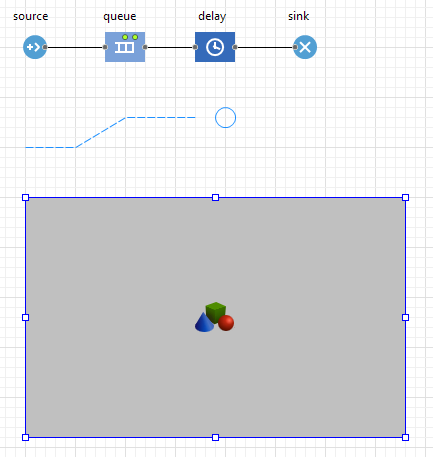


**Добавление 3D анимации**

Первым делом нам будет нужно добавить на диаграмму типа агента 3D Окно. 3D Окно используется для задания на диаграмме агента области, в которой во время запуска модели будет отображаться трехмерная анимация этой модели.

***Добавление 3D окна***

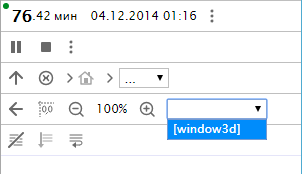
1. Перетащите элемент **3D Окно** из секции **3D** палитры **Презентация** в графический редактор.
2. Вы увидите в графическом редакторе закрашенную серым область. Поместите ее туда, где вы хотите видеть 3D анимацию во время запуска модели:



***Перемещение в 3D анимации***

Теперь вы можете запустить модель и наблюдать простую 3D анимацию.

1. При создании 3D окна, AnyLogic добавляет область просмотра, которая позволяет легко переключаться к сцене 3D анимации во время выполнения модели. Чтобы переключиться к 3D анимации в запущенной модели, откройте панель разработчика, щелкнув по кнопке http://127.0.0.1:49204/help/topic/com.anylogic.help/html/running/images/debug.png **Показать/скрыть панель разработчика** в правом углу панели управления. В открывшейся панели разработчика раскройте список и выберите опцию **[window3d]**.



1. Вы можете перемещаться по 3D сцене с помощью мыши и следующих клавиш:

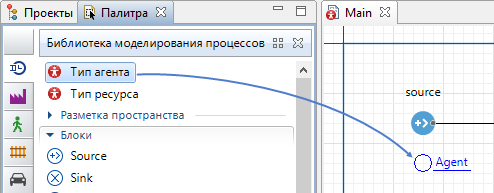
|  |  |
| --- | --- |
| **Чтобы** | **Выполните следующие действия** |
| **Переместить сцену** | 1. Нажмите левую кнопку мыши в области 3D окна и держите ее нажатой. 2. Передвиньте мышь в направлении перемещения. |
| **Повернуть сцену** | 1. Нажмите клавишу **Alt** и держите ее нажатой. 2. Нажмите левую кнопку мыши в области 3D окна и держите ее нажатой. 3.  Передвиньте мышь в направлении вращения. |
| **Приблизить/отдалить сцену** | 1. Покрутите колесо мыши от/на себя в области 3D окна. |

***Добавление 3D объектов***

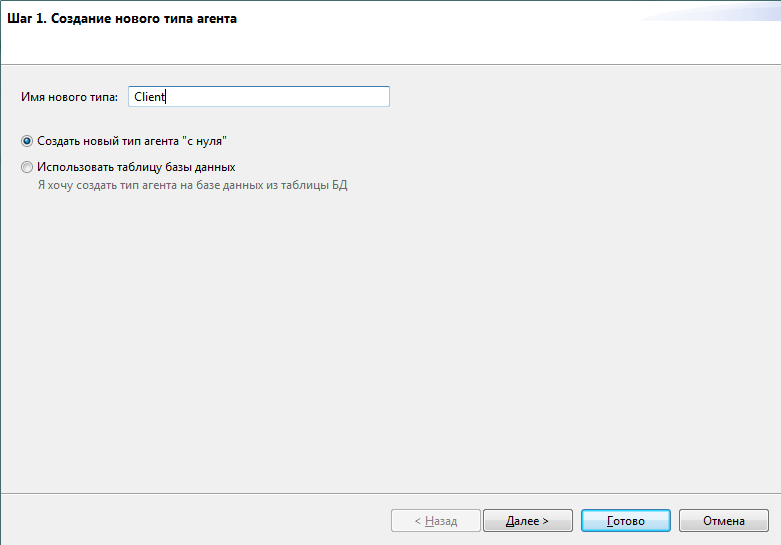
Теперь мы хотим задать фигуру клиента банка. По умолчанию клиенты в нашей модели обозначались цветными точками и отображались цветными цилиндрами в 3D анимации. Если мы хотим задать нестандартный тип клиента и выбрать для него красивую фигуру анимации, нам нужно создать новый тип агента.

***Создание нового типа агента***

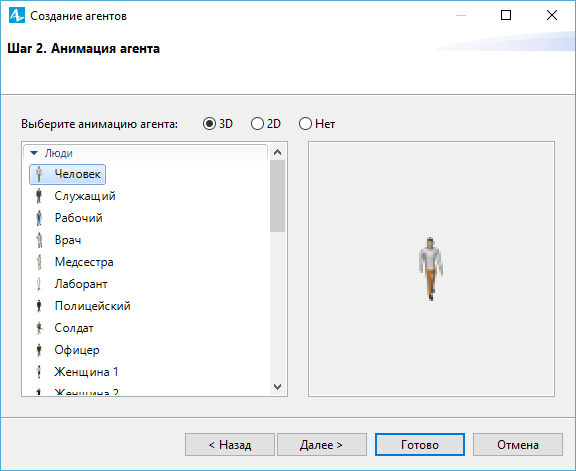
1. Откройте **Библиотеку моделирования процессов** в панели **Палитра** и перетащите элемент **Тип агента** в графический редактор.



1. Откроется диалоговое окно Мастера создания агентов на шаге **Создание нового типа агента**. Введите *Client* в поле **Имя нового типа**, оставьте опцию  **Создать новый тип агента "с нуля"** выбранной. Нажмите **Далее**.



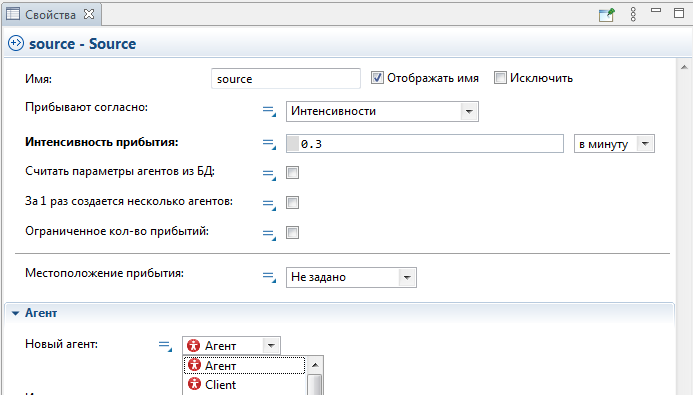
1. Выберите опцию **3D** для типа анимации и фигуру анимации *Человек* из списка 3D фигур.



1. Щелкните **Готово**. Диаграмма нового агента Clientоткроется автоматически. Вы можете найти3D фигуру *Человек* в начале координат.

***Настройка использования нового типа агентов в схеме***

1. На диаграмме Main, выделите блок *source* в графическом редакторе.
2. Выберите тип агента Customer  в выпадающем списке параметра Новый агент.

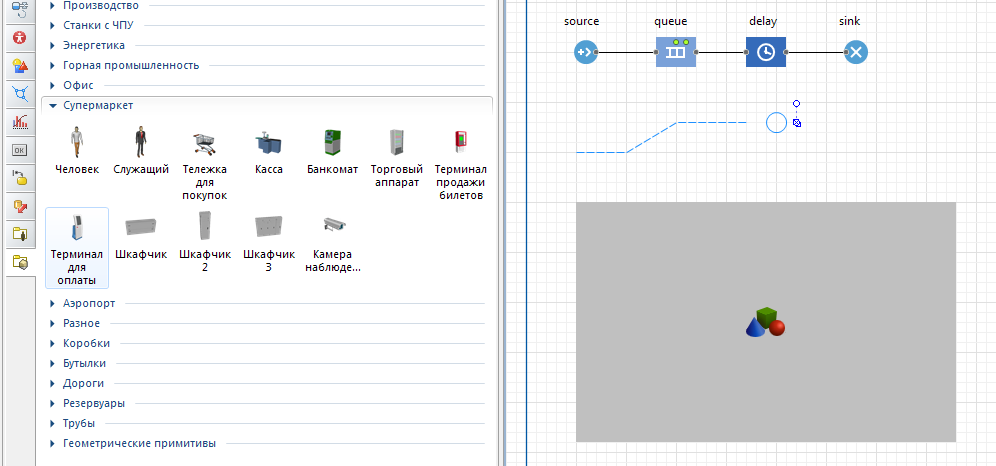


1. Запустите модель, чтобы увидеть анимацию клиентов в очереди.

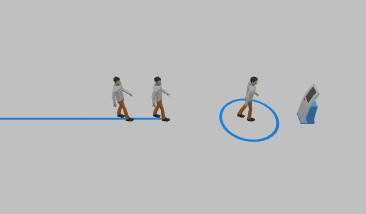


***Добавления объекта автомата обслуживания***

1. Откройте палитру **3D Объекты** в панели **Палитра**.
2. Перетащите 3D фигуру **Терминал для оплаты** из секции палитры **Супермаркет** в графический редактор и поместите ее на точечный узел.



1. Запустите модель, чтобы убедиться, что фигура банкомата стоит "лицом" к клиентам.



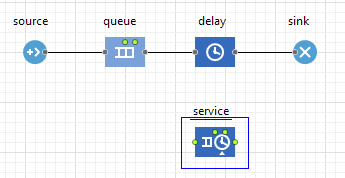
**3. Добавление кассиров**

Теперь мы усложним модель, добавив в нее служащих –кассиров. Мы могли бы промоделировать кассиров, как и банкомат, с помощью объекта **Delay**. Но куда более удобным представляется моделирование кассиров с помощью ресурсов. Ресурс – это специальный объект **Библиотеки моделирования процессов**, который может потребоваться агенту для выполнения какой-то задачи. В каждый момент времени ресурс может быть занят только одним агентом. В нашем примере посетителям супермаркета (агентам) необходимо оплатить свои покупки у кассиров (ресурсов).

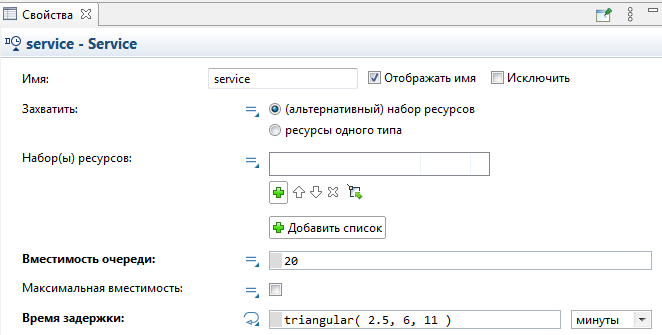
**Изменения в диаграмме процесса**

***Добавление обслуживания***

1. Откройте **Библиотеку моделирования процессов** в панели **Палитра** и перетащите на диаграмму класса Main блок [**Service**](http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ProcessModeling/Service.html). Объект **Service** захватывает для агента заданное количество ресурсов, задерживает агента, а затем освобождает захваченные им ресурсы.

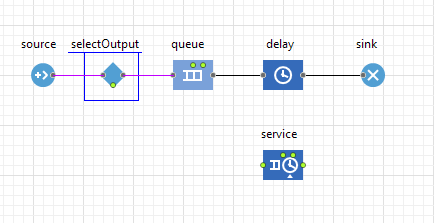


1. Перейдите в панель **Свойства** блока *service*.
2. Измените параметры объекта следующим образом:
   1. Ко всем кассирам будет вести одна общая очередь. Задайте максимальное количество человек в этой очереди в поле **Вместимость очереди**: *20*.
   2. Мы полагаем, что время обслуживания имеет треугольное распределение с минимальным значением равным *2.5*, средним - *6*, и максимальным - *11*минутам. Введите в поле **Время задержки**: triangular( 2.5, 6, 11 )

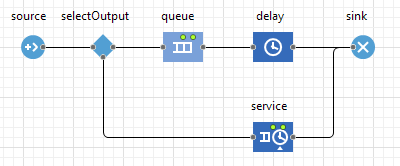


***Моделирование выбора клиентов***

1. Откройте **Библиотеку моделирования процессов** в панели **Палитра** и перетащите на диаграмму процесса Main блок [**SelectOutput**](http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ProcessModeling/SelectOutput.html) в свободное место между блоками *source* и *queue*. **SelectOutput** является блоком принятия решения. В зависимости от заданного вами условия, агент, поступивший в объект, будет поступать на один из двух выходных портов объекта.

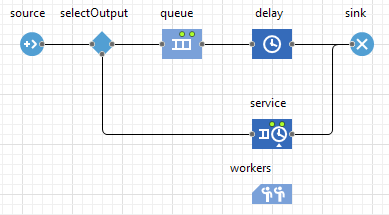


1. Выделите блок *selectOutput* в диаграмме процесса. В панели **Свойства** этого блока выберите опцию *При выполнении условия* в параметре **Выход true выбирается**. Убедитесь, что в поле **Условие** стоит выражение randomTrue( 0.5 ). В этом случае к кассирам и автомату будет приходить примерно равное количество клиентов.
2. Соедините блоки *selectOutput* и *service* с другими блоками так, как показано на рисунке ниже:

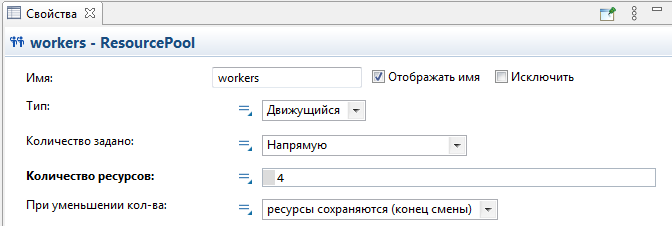


***Добавление ресурсов для сервиса***

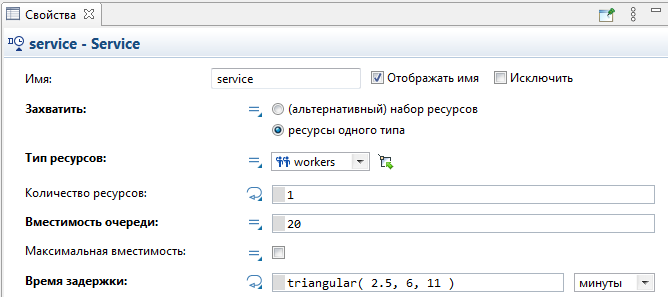
1. Откройте **Библиотеку моделирования процессов** в панели **Палитра** и перетащите блок [**ResourcePool**](http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ProcessModeling/ResourcePool.html) на диаграмму агента Main. Объект **ResourcePool** задает ресурсы определенного типа (в нашей модели это будут кассиры).
2. Поместите его, например, под блоком *service* и перейдите в панель **Свойства**.
3. Назовите объект *workers*.



1. Задайте число кассиров в поле **Кол-во ресурсов**: *4*.



1. Блок **ResourcePool** указывается в объектах, использующих ресурсы, в нашем случае это блок **Service**. Поэтому нам необходимо изменить свойства блока *service* диаграммы процесса.
2. Выделите блок *service* и перейдите в панель **Свойства**. Выберите опцию *Ресурсы одного типа* в параметре **Захватить ресурсы**. Затем укажите блок *workers*, который мы добавили на диаграмму, в параметре **Блок** **ResourcePool**. Вы можете выбрать его из выпадающего списка подходящих объектов, щелкнув стрелку "вниз", или выбрать фигуру из графического редактора, предварительно щелкнув кнопку справа от параметра (в таком случае все неподходящие объекты в графическом редакторе будут обесцвечены).



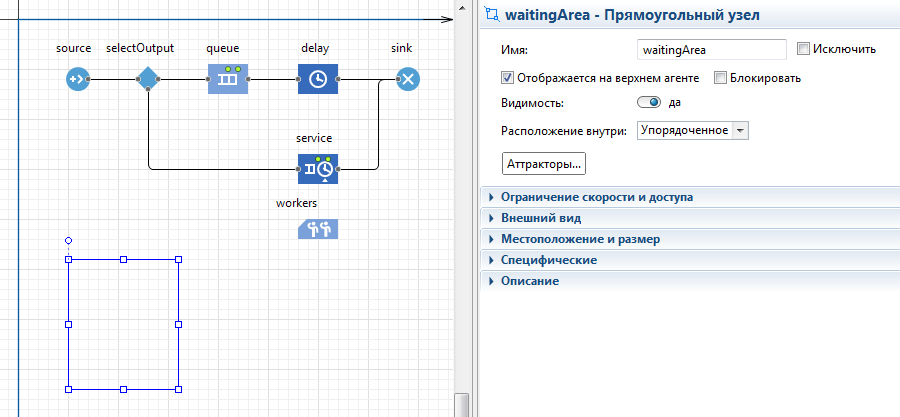
1. Поскольку наша модель изменилась, мы должны изменить и ее анимацию.

**Добавление фигур разметки пространства**

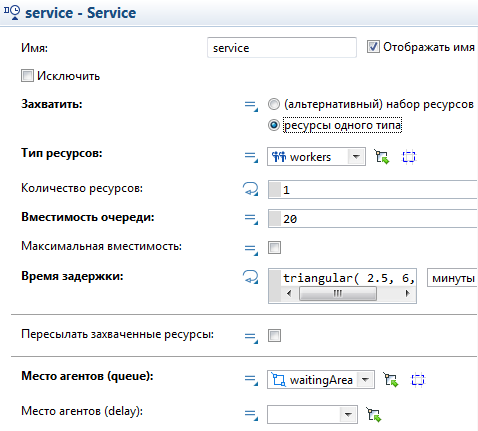
Теперь давайте нарисуем область для ожидания и место обслуживания клиентов кассирами.

***Задание фигуры разметки для электронной очереди***

1. В этот раз мы будем рисовать место ожидания клиентами, используя прямоугольный узел. Вначале откройте палитру **Разметка пространства** панели **Палитра**.
2. Двойным щелчком выделите элемент **Прямоугольный узел** палитры **Разметка пространства**, чтобы перейти в *режим рисования*.
3. Щелкните мышью в графическом редакторе, чтобы задать вершину верхнего левого угла, затем тащите прямоугольник, не отпуская кнопки мыши. Отпустите, когда прямоугольный узел достигнет нужной формы. Вы можете редактировать фигуру и после того, как ее рисование завершено
4. Назовите эту область *waitingArea*.
5. Переключите элемент управления **Видимость** в положение **нет**. Таким образом разметка пространства не будет видна на анимации во время выполнения модели.

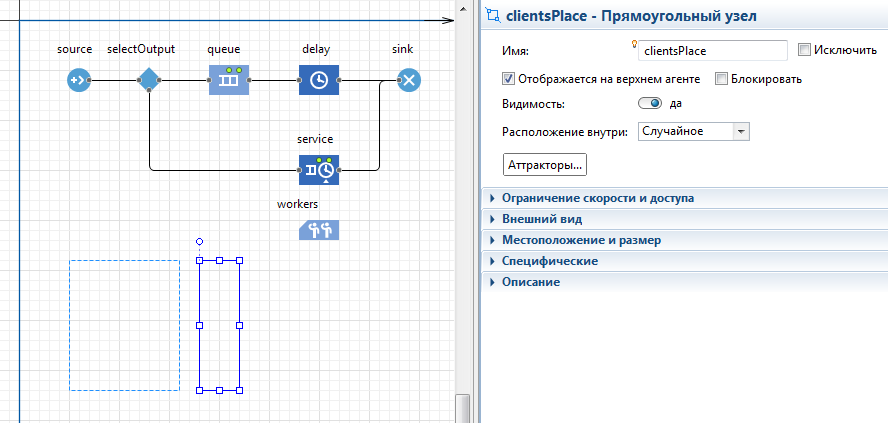


1. Выделите щелчком блок *service block* в диаграмме процесса и перейдите в его свойства.
2. Выберите только что нарисованный нами узел *waitingArea* в параметре **Место агентов (queue)**

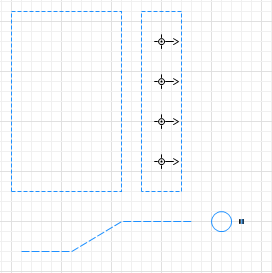


***Задание фигуры разметки места обслуживания клиентов***

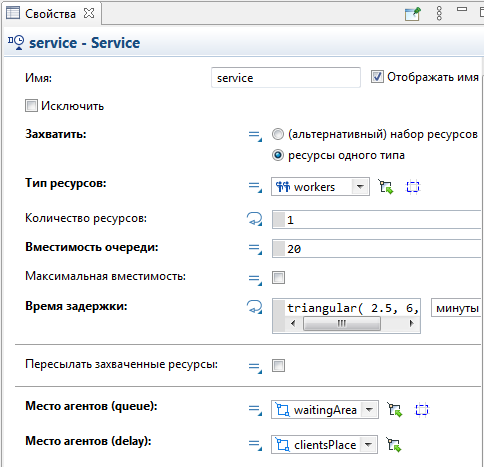
1. Клиентам магазина требуется место, на котором они могли бы находиться во время обслуживания у кассиров. Мы нарисуем такую область, используя прямоугольный узел.
2. Вначале откройте палитру **Разметка пространства** панели **Палитра**.
3. Двойным щелчком выделите элемент **Прямоугольный узел** http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ELT/images/RectangleNode_obj.gifпалитры **Разметка пространства**, чтобы перейти в *режим рисования*.
4. Щелкните мышью в графическом редакторе, чтобы задать вершину верхнего левого угла, затем тащите прямоугольник, не отпуская кнопки мыши. Отпустите, когда прямоугольный узел имеет нужную форму. Вы можете редактировать фигуру и после того, как ее рисование завершено.
5. Назовите эту область *clientsPlace*.
6. Переключите элемент управления **Видимость** в положение **нет**.



1. Мы будем использовать аттракторы, чтобы задать местоположение тех клиентов, которые будут обслуживаться у кассиров. Выделите узел *clientsPlace* в графическом редакторе и щелкните кнопку **Аттракторы** в свойствах узла. В открывшемся окне **Аттракторы** укажите число аттракторов *4* в режиме создания **Количество аттракторов**, затем щелкните **OK**. Вы увидите, что четыре аттрактора появились в узле *clientsPlace* на равном расстоянии друг от друга.

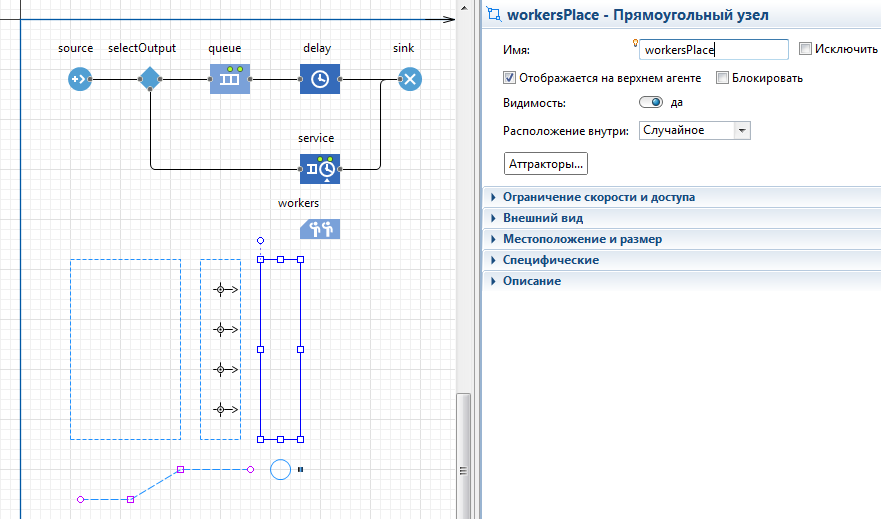


1. Теперь нам необходимо сослаться на эту фигуру в диаграмме процесса. Щелкните блок *service* и перейдите в панель **Свойства** этого блока.
2. Выберите нарисованный нами узел *clientsPlace* в параметре **Место агентов (delay)**

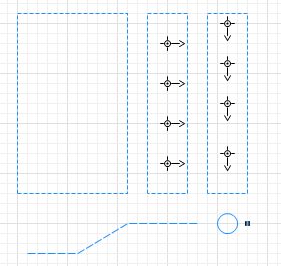


***Задание фигуры разметки для кассиров***

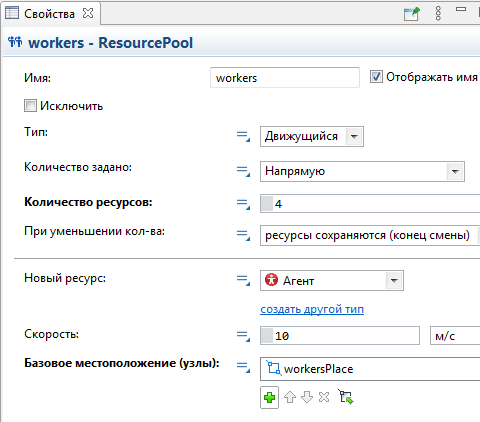
1. Кассирам также требуется место, где они могли бы находиться во время обслуживания клиентов. Мы нарисуем такую область, используя прямоугольный узел.
2. Вначале откройте палитру **Разметка пространства** панели **Палитра**.
3. Двойным щелчком выделите элемент **Прямоугольный узел** палитры **Разметка пространства**, чтобы перейти в *режим рисования*.
4. Щелкните мышью в графическом редакторе, чтобы задать вершину верхнего левого угла, затем тащите прямоугольник, не отпуская кнопки мыши. Отпустите, когда прямоугольный узел имеет нужную форму. Вы можете редактировать фигуру и после того, как ее рисование завершено.
5. Назовите эту область *workersPlace*.
6. Переключите элемент управления **Видимость** в положение **нет**.



1. Мы будем использовать аттракторы, чтобы задать местоположение клерков. Выделите узел *workersPlace* в графическом редакторе и щелкните кнопку **Аттракторы** в свойствах узла. В открывшемся окне **Аттракторы** укажите число аттракторов *4* в режиме создания **Количество аттракторов**, затем щелкните **OK**.



1. Щелкните объект *workers* в диаграмме процесса и перейдите в его свойства.
2. Выберите нарисованный нами узел *tellerPlaces* в параметре **Базовое местоположение (узлы)**



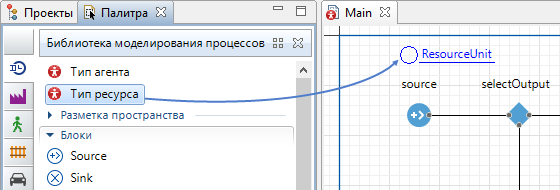
Вы можете запустить модель и наблюдать, как клиенты обслуживаются у автомата и проходят к кассирам.

**Добавление 3D объектов**

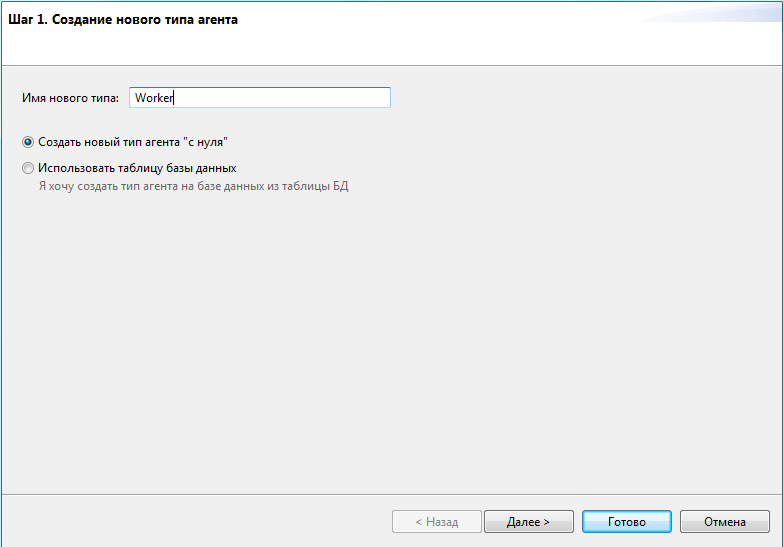
Давайте добавим 3D фигуры кассиров в нашу модель. Мы создадим новый тип ресурсов для анимации кассиров.

***Создание нового типа ресурсов***

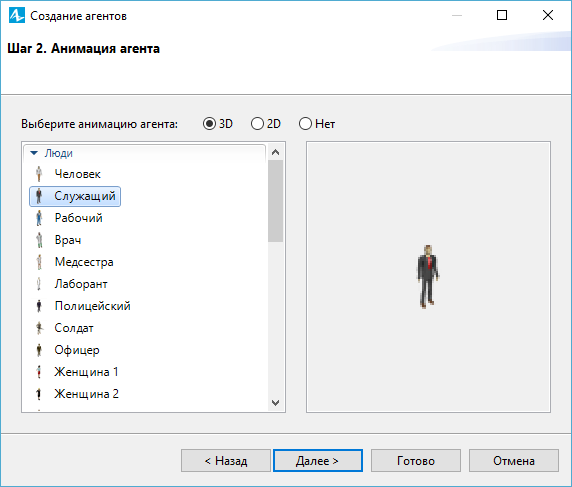
1. Откройте **Библиотеку моделирования процессов** в панели **Палитра**.
2. Перетащите элемент **Тип ресурса** http://127.0.0.1:49204/help/topic/com.anylogic.help/html/_ELT/images/Agent_icon.gif в графический редактор.



1. Откроется диалоговое окно Мастера создания агентов на шаге**Создание нового типа агента**. Введите *Worker* в поле **Имя нового типа**, оставьте опцию **Создать новый тип агента "с нуля"** выбранной. Нажмите **Далее**.



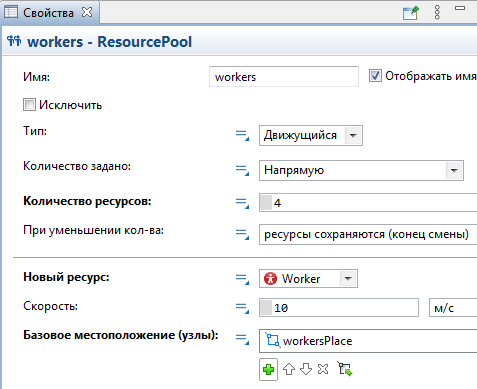
1. Выберите опцию **3D** для типа анимации и фигуру анимации *Служащий* из списка 3D фигур.



1. Щелкните **Готово**. Новая диаграмма Worker автоматически откроется. Вы можете найти 3D фигуру *Служащий* в начале координат. Переключитесь обратно на диаграмму класса Main.

***Настройка использования нового типа ресурсов в схеме***

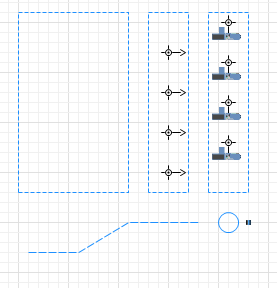
1. На диаграмме Main, выделите блок *workers* в графическом редакторе.
2. Выберите тип ресурсов Worker в выпадающем списке параметра **Новый ресурс.**



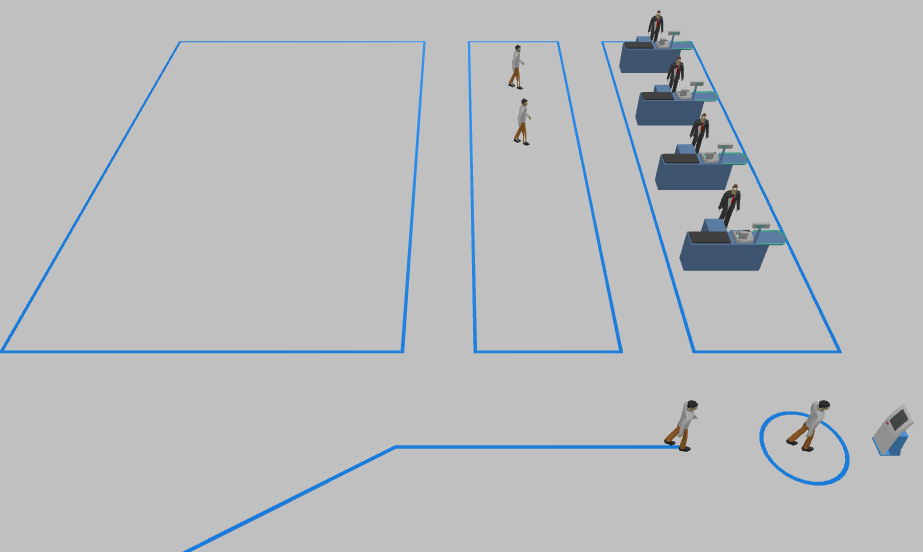
1. Запустите модель, чтобы увидеть получившуюся анимацию кассиров.

***Добавление кассовых аппаратов для кассиров***

1. Откройте палитру **3D Объекты** в панели **Палитра**.
2. Перетащите четыре 3D фигуры **Касса** из секции палитры **Супермаркет** в графический редактор и поместите их в узел *workersPlace.*
3. Расположите столы на аттракторах, так как аттракторы обозначают место, где стоят кассиры
4. При необходимости, выровняйте расположение всех восьми аттракторов и столов.



Теперь вы можете запустить модель и увидеть в 3D анимации, как некоторые клиенты идут к автомату, а другие обслуживаются у кассиров.

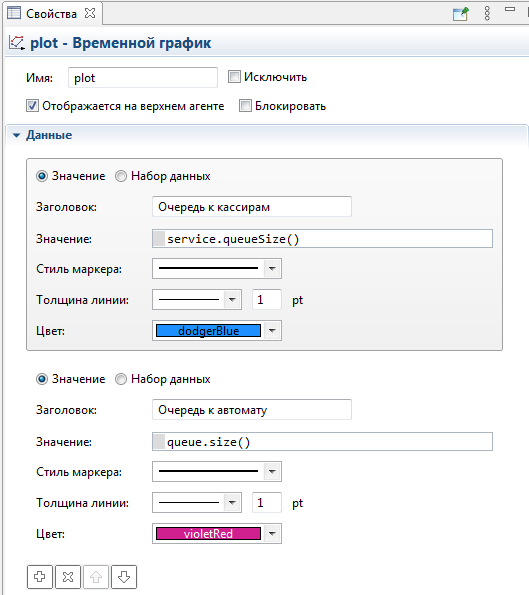


**4. Добавление графиков для отображения основных характеристик**

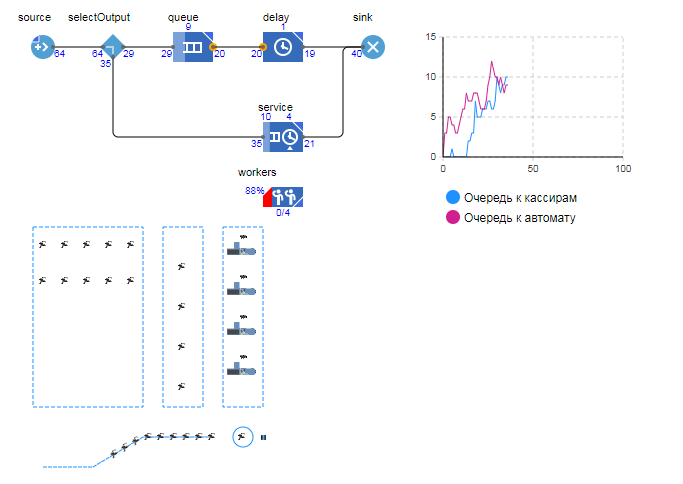
Теперь мы добавим на схему класса Main несколько графиков, которые позволят нам отследить основные характеристики системы массового обслуживания.

**Определение размера очереди**

1. Добавьте на схему временной график
2. В качестве значений примем размеры очереди для элементов *queue* и *service,* как показано на рисунке ниже.

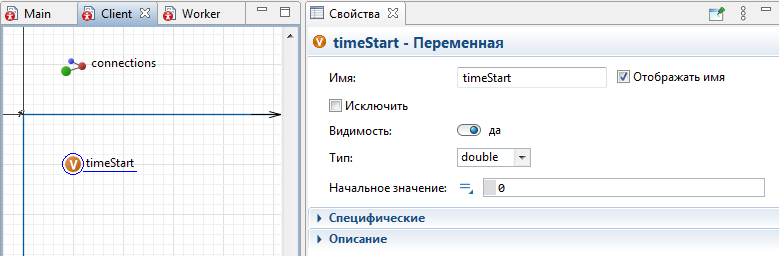


1. Приблизительный вид графиков после запуска процесса моделирования представлен на изображении ниже.

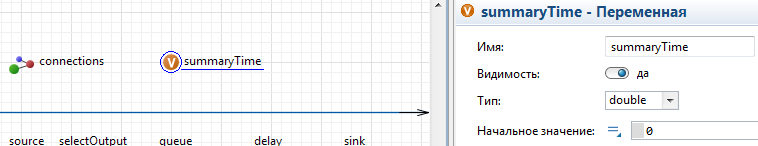


**Определение среднего времени нахождения агента в очереди**

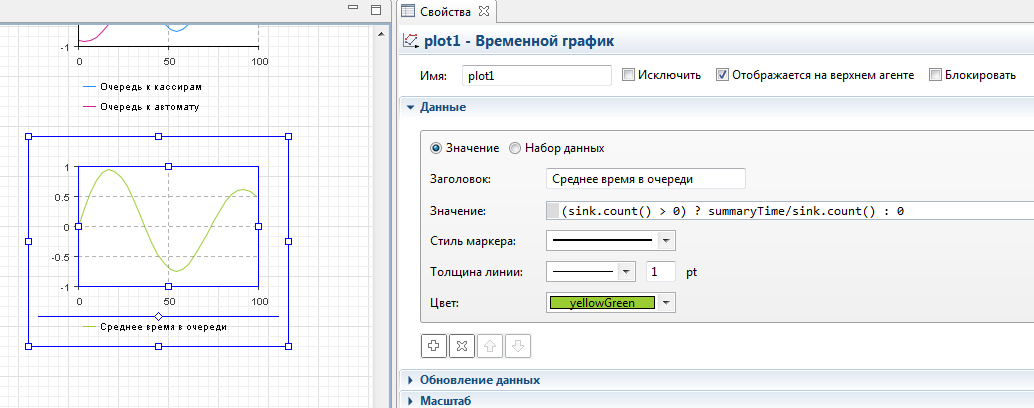
1. Для вычисления данного значения необходимо добавить переменную *timeStart* внутрь класса *Client*. Начальное значение – 0.



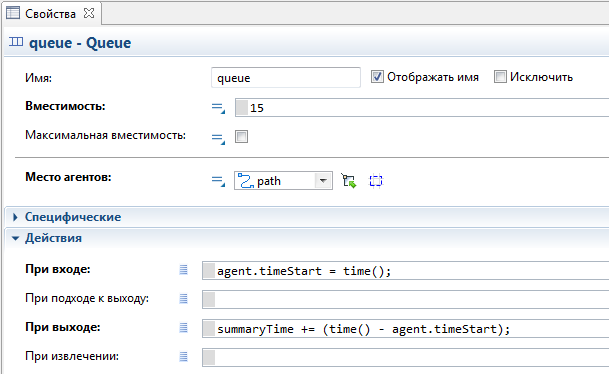
1. Также необходимо добавить переменную для накопления общего времени ожидания всех клиентов в класс *Main*. Начальное значение также 0.



1. В качестве среднего времени ожидания примем отношение общей суммы времени, проведенного в очереди, к количеству обработанных заявок. В момент времени, когда еще ни одна заявка не обработана, попытка расчета значения будет вызывать ошибку. Для ее устранения воспользуемся тернарным оператором, как показано ниже.

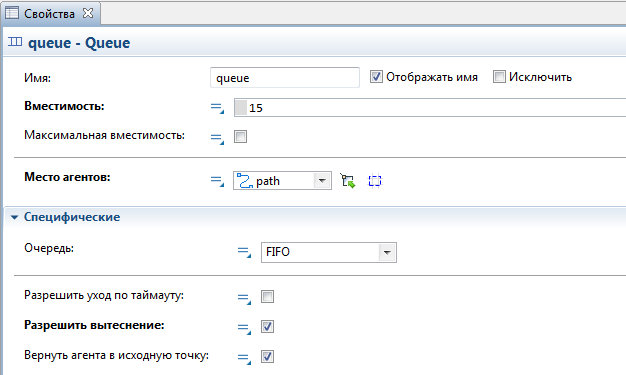


1. Также в элементах *queue* и *service* необходимо задать действия для агентов при входе и выходе из блока. Таким образом мы получим от каждого агента то время, которое он провел в ожидании обслуживания.

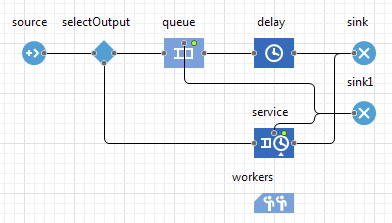


**Определение процента отказов от обслуживания**

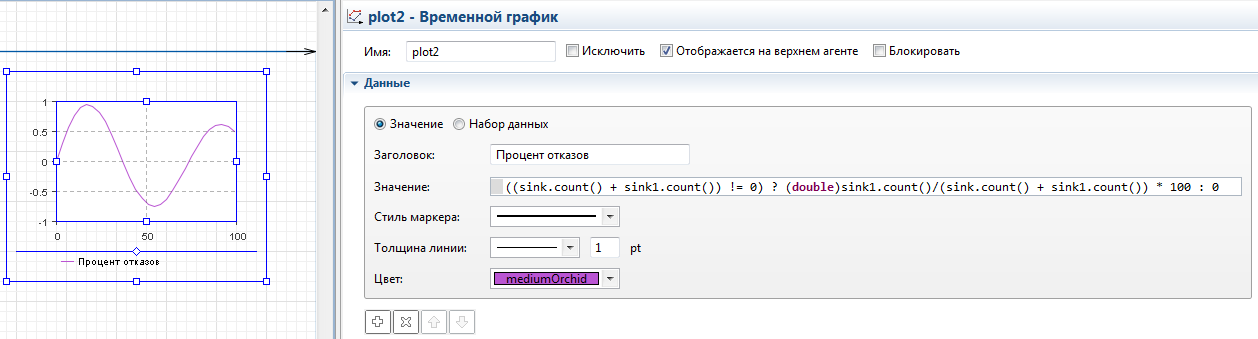
1. Для реализации самой механики отказа от обслуживания необходимо разрешить объектам *queue* и *service* вытеснения путем проставления соответствующей галочки в свойствах элементов.



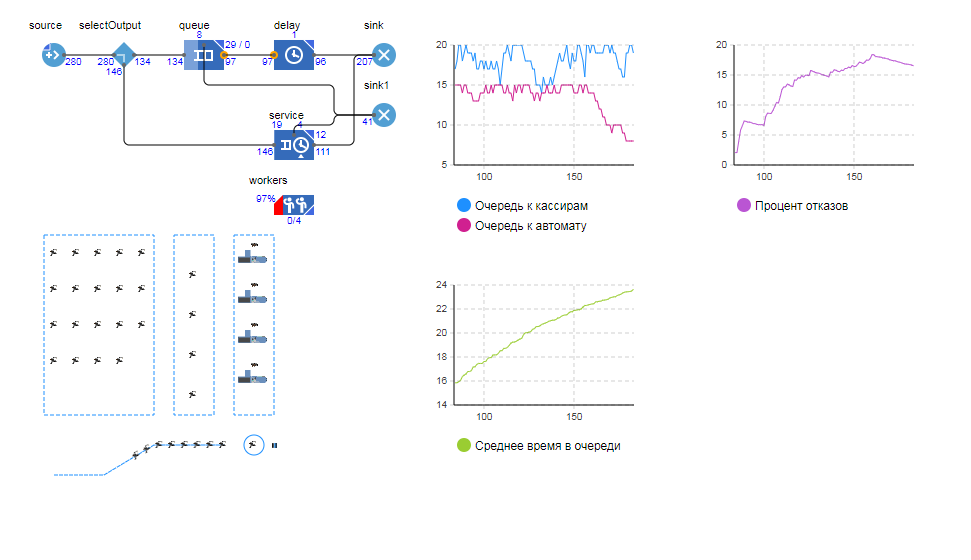
1. Также необходимо провести стрелки от портов элементов, отвечающих за данную механику, к отдельному блоку уничтожения агентов. Модель реализации представлена на рисунке ниже.



1. Значение рассчитывается, как отношение заявок, по которым был получен отказ, к общему количеству обработанных заявок, умноженному на 100. Здесь же следует учесть вероятность деления на 0.



1. Диаграмма нашей модели со всеми основными характеристиками СМО приведена на рисунке ниже.



**Дополнительная часть**

1. Порядок поступления заявок реализовать в соответствии с действительным положением вещей, с двумя пиками: в районе 13:00 и 18:00. (2 балла)
2. Реализовать три типа заявок(обрабатываемые банкоматом, требующие участия оператора, допускающие оба сценария) и распределение их по исполнителям в соответствии с типом. (2 балла)
3. Изменить порядок распределения заявок по исполнителям следующим образом: четыре автомата, очередь к которым представляет общую область, четыре кассира с индивидуальными очередями. Распределение по индивидуальным очередям происходит на основе общего времени обработки заявок, уже находящихся в очереди (т.е., как в магазине в реальной жизни, выбирается очередь, в которой у людей минимальное количество покупок). Для реализации данного функционала необходимо добавить время обработки заявки в качестве параметра внутрь класса агента. (6 баллов)