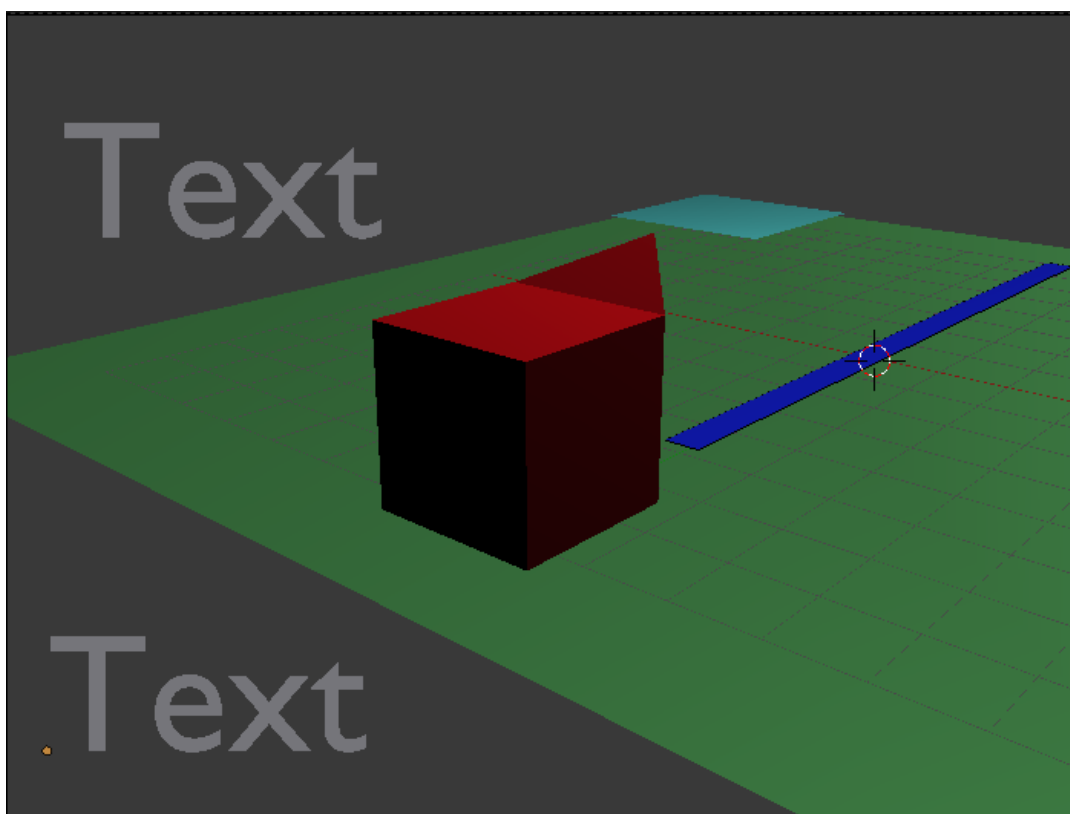
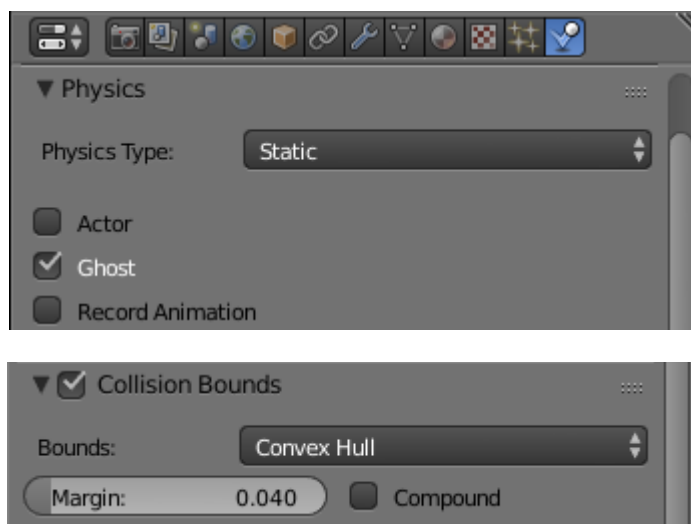


Перед следующим уроком, где мы будем разговаривать о цели игры, давайте разберёмся с тем, как обнаружить объект. И ещё, как передать данные от одного объекта другому. Здесь мы не будем использовать ни одной строчки кода на Python. Только логические блоки.

Создадим простую сцену с динамическим кубиком и статической плоскостью. Прикрепим к кубику камеру слежения и настроим ему логику управления движением. На плоскости «земли» разместим ещё две (чуть выше над ней): одну длинную, и ещё одну в углу. Длинную мы назовём «Liny». Её наш кубик будет пересекать и на экране мы увидим количество пересечений. Угловую плоскость мы назовём «Finish». Если мы на неё переместим наш кубик, то появится текстовое сообщение. Две этих плоскости статичны, но галочку Ghost в их физике поставить необходимо. И наконец, разместим два текстовых объекта. Один (нижний) для подсчёта пересечений, а второй – для финишного сообщения. В разделе «Готовые проекты» вы можете скачать blend файл проекта к уроку. В общем сцена выглядит так:

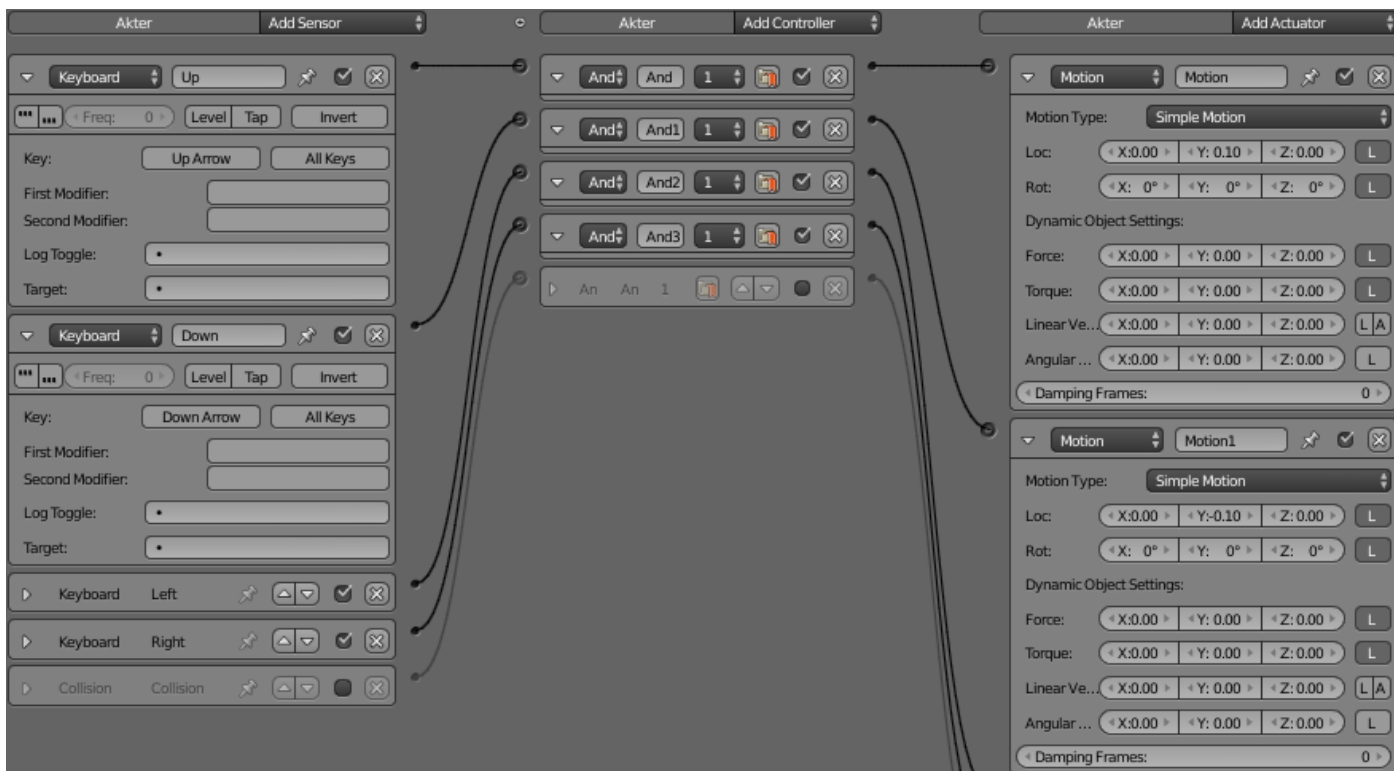


Физика «Liny» и «Finish»:

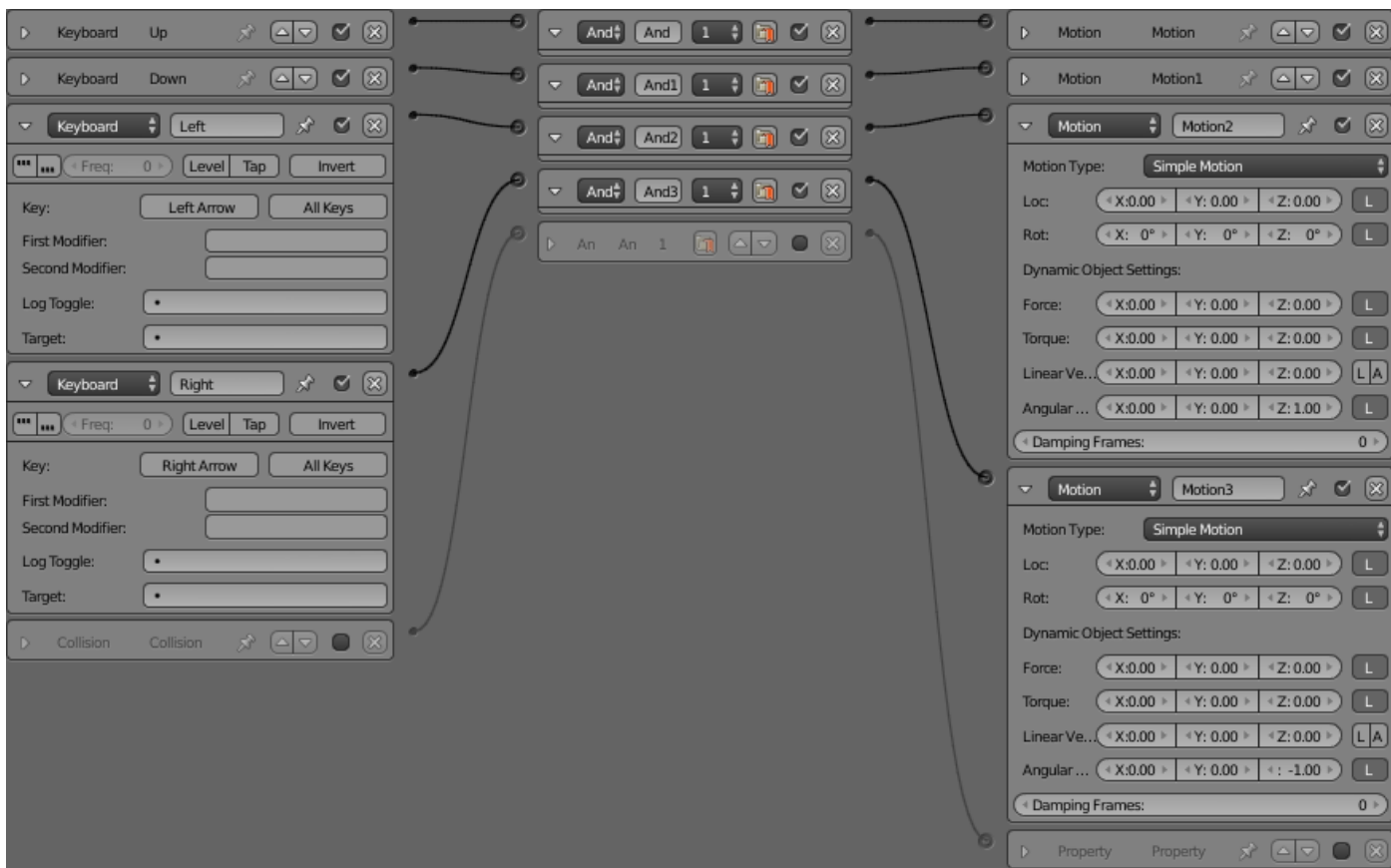


Всем объектам (кроме текстовых) назначаем свои материалы. Поскольку пересечение будет отслеживаться именно по ним. Не забудьте, что кубик (Akter) должен быть Rigid Body 😊. Логика управления кубиком выглядит так:

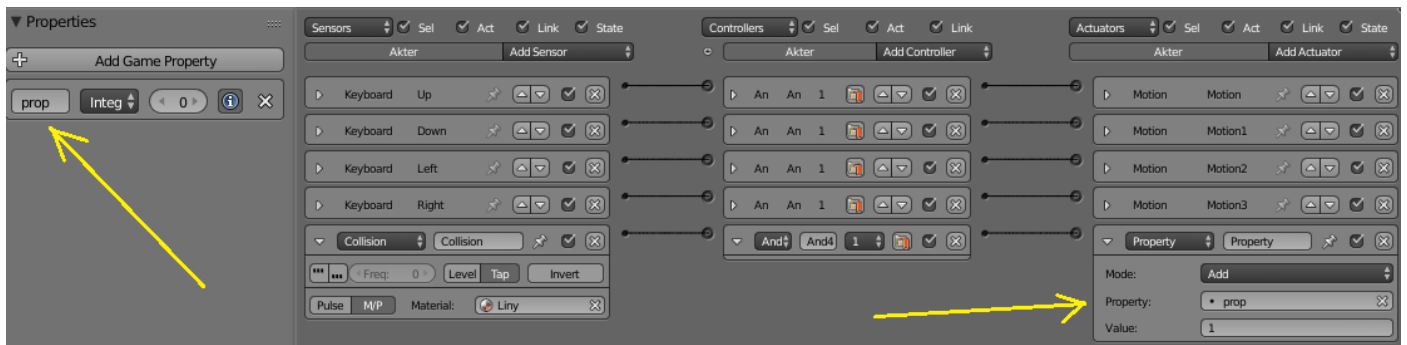
### Вперёд – Назад



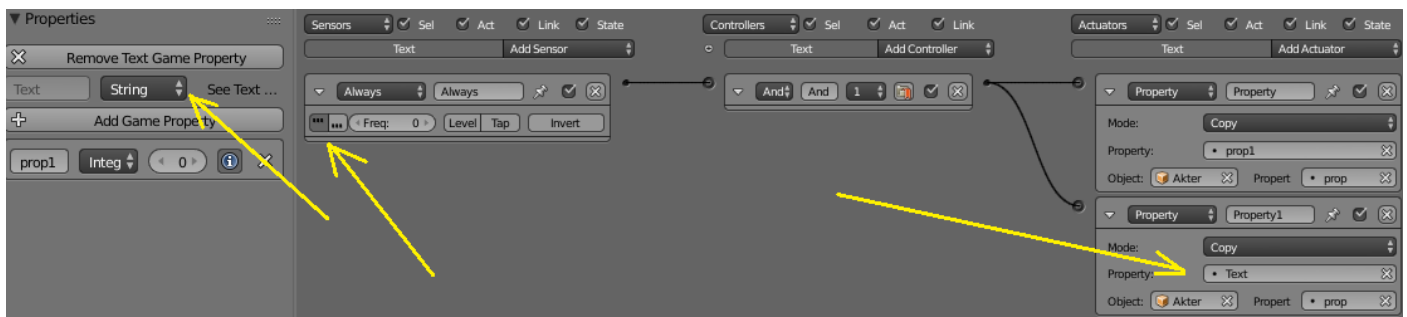
### Влево – вправо



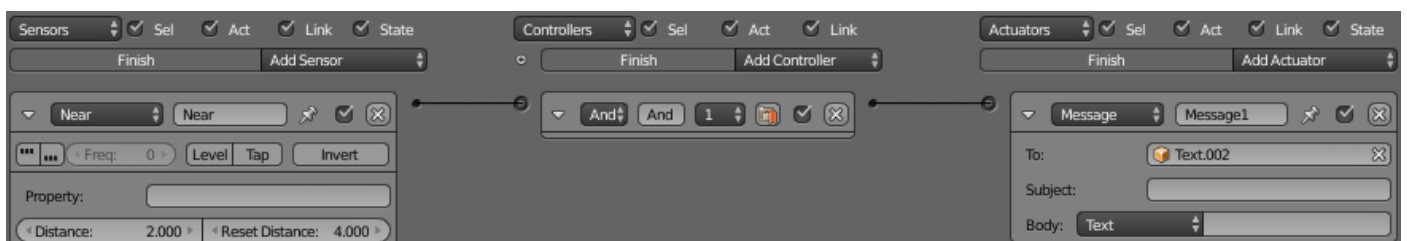
А теперь настраиваем подсчёт пересечений плоскости «Liny»:



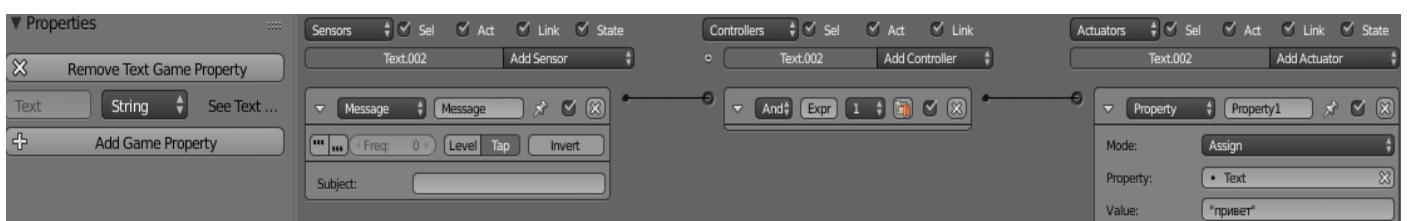
Мы в объекте «Akter» создали свойство «prop». В столкновениях назначили материал объекта «Liny» и связали с актуатором Property. В нём мы установили Add (добавлять), указали свойство (чему добавлять) и Value (сколько добавлять). Теперь выделяем нижний текстовый объект и настраиваем подсчёт пересечений:



Создаём текстовое свойство (показано стрелкой). В сенсоре Always нажимаем кнопку (отслеживать постоянно). В актуаторе Property выбираем Copy (копировать свойство), задаём наше текстовое свойство и указываем из какого объекта и какое свойство мы хотим скопировать. Готово. К слову, свойство prop1 я создал сугубо в образовательных целях. Роли оно в данном случае не играет, но зато показывает, что и в обычное свойство (не текстовое) тоже можно скопировать данные. Теперь если вы запустите игру и пересечёте линию, то нижний текстовый объект будет отображать цифры. Переходим к плоскости «Finish». Вот её логика:



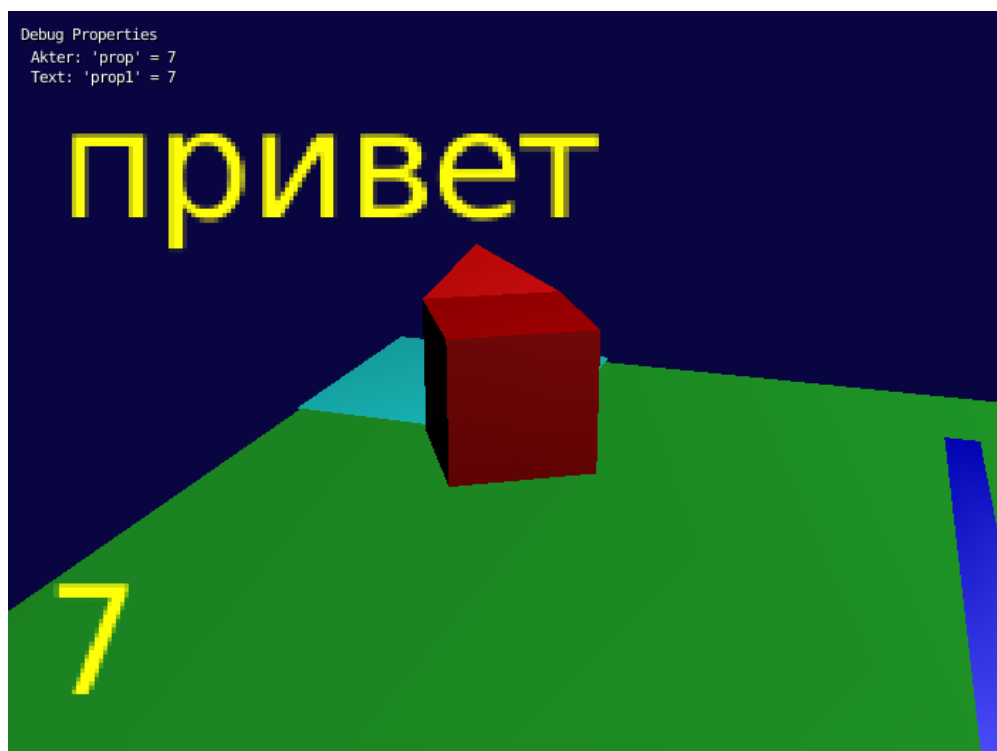
Сенсор Near срабатывает при приближении любого динамического объекта на расстоянии Distance, и посылает один положительный импульс. Который я адресовал актуатору сообщений Message. Что он делает? Просто посылает сообщение, что он сработал. Сообщение может быть принято в любом объекте, имеющем сенсор Message. В данном случае мы отправили сообщение верхнему текстовому объекту. Открываем логику верхнего текстового объекта:



Снова вставляем текстовое свойство. Сенсор **Message** только принимает сообщение, что где-то сработал актуатор **Message**. Сенсор передаёт один положительный импульс на актуатор **Property**. В нём мы выставляем **Assigning** (присвоить), указываем текстовое свойство и пишем текст, который нужно показать на экране. Готово. Как только кубик приблизится к центру плоскости **Finish** на расстояние 2 blend единиц, то сработает сенсор **Near**, который отправит об этом сообщение текстовому объекту. Где сенсор сообщений примет его и присвоит текстовому объекту текстовое свойство «Привет». Оно и отобразится на экране.

Мы рассмотрели два важных момента игры. Первое – отслеживание объекта. Второе – передача данных между объектами.

Для отслеживания динамических объектов есть два сенсора: **Near** и **Radar**. **Near** видит сквозь стены и обнаруживает любой динамический объект в своём радиусе. Хотя, можно задать определённое свойство для объекта отслеживания. **Radar** видит сквозь стены и отслеживает любой динамический объект (либо по свойству) в своём радиусе, но в определённом направлении. Другие направления игнорируются. **Ray**, в противоположность первым двум, может быть прикреплен к динамическому объекту и отслеживает статические объекты (стены, препятствия и т.д.). Не видит сквозь стены. Отслеживает объекты по материалу. Как вы понимаете, без этих сенсоров нельзя сделать искусственный интеллект для шутера или автосимулятора. В дальнейшем мы будем их рассматривать более подробно.



20 мая 2015 года

Составил **Niburiec** для сайта <http://blender-game.ucoz.ru>