

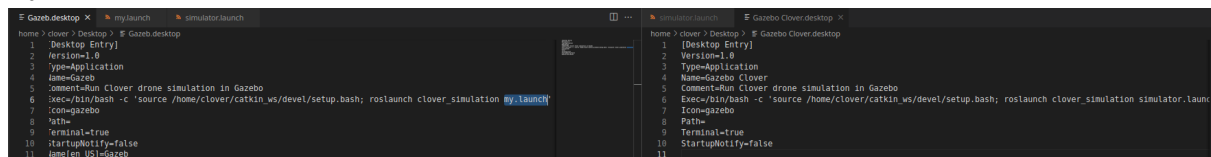
файлы, необходимые для установки мира Gazebo на другом компьютере без выполнения дополнительных настроек конфигурационных файлов квадрокоптера:

Нужно каждую папку в папке export вставить с дополнением по директориям, как они расположены в export 2, т.е заходим в /home/clover/catkin_ws/src/clover/ и по каждому пути в папке экспорта вставляем

Если делать самим - ниже:

Ярлык

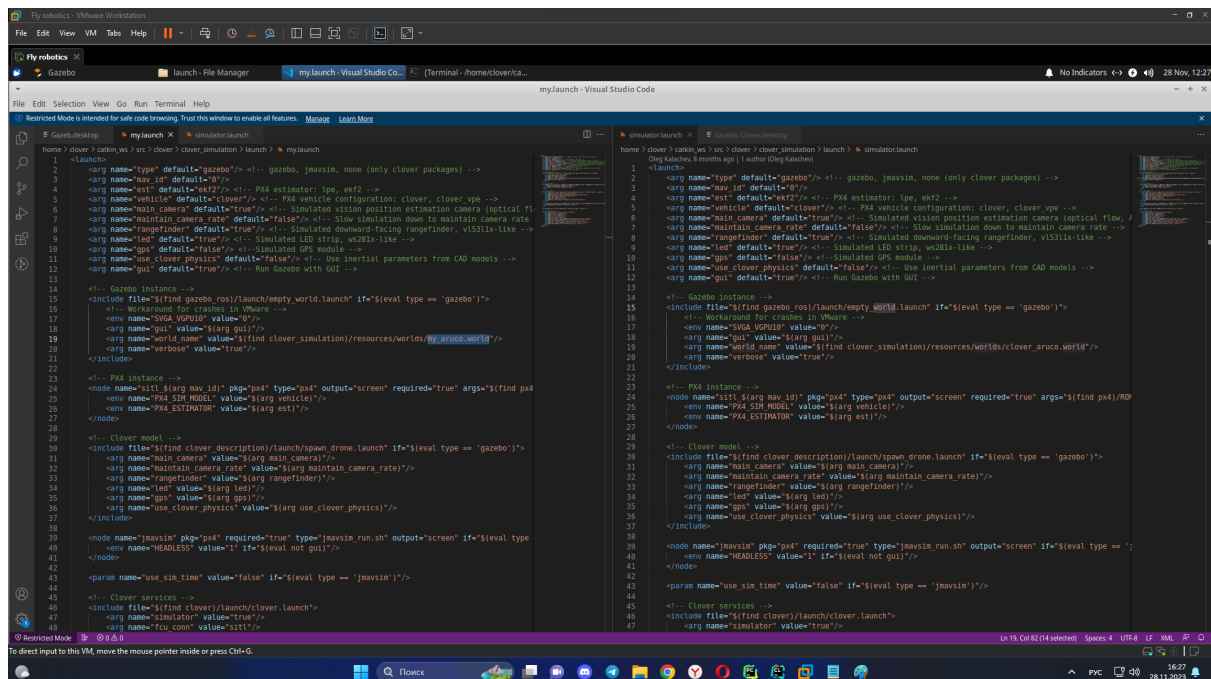
Первым делом нам нужно будет создать копию ярлыка для запуска ROS в паре с gazebo для нашего нового мира, копируем, вставляем ярлык Gazebo Clover на рабочем столе, переименовываем на MyGazebo. Открываем с помощью в VS code, там также меняем все вхождения "Gazebo Clover" на "MyGazebo", simulator.launch меняем на my.launch, сохраняем, выходим.



Идем по пути "/home/clover/catkin_ws/src/clover/clover_simulation/", дальше всегда все манипуляции будем проводить в этой директории (если не оговорено обратное).

Launch файл

Первым делом пройдем по ./launch, скопируем simulator.launch, вставим здесь же, и переименуем на my.launch, дальше откроем my.launch в VS code и через замену (ctrl + F и раскрыть выпадающее меню, написать в первую строку что заменить, во вторую - на что заменить), заменим clover_aruco.world на my.world, сохраним.



Вставка кода для фигур

Далее из clover_simulation пройдем в resources/worlds, там скопируем clover_aruco.world, вставим, переименуем на my.world, откроем и вставим туда между последним тегом </include> и <scene> xml код для наших фигур:

```
<include>
```

```
  <uri>model://Blue_hexagon</uri>
```

```
  <pose>0.22 2.35 0.0 0 0 0</pose>
```

```
</include>
```

```
<include>
```

```
  <uri>model://Blue_rhombus</uri>
```

```
  <pose>2.8 2.48 0.0 0 0 0</pose>
```

```
</include>
```

```
<include>
```

```
  <uri>model://Green_pentagon</uri>
```

```
  <pose>0.1 3.16 0.0 0 0 0</pose>
```

```
</include>
```

```
<include>
```

```
  <uri>model://green_square</uri>
```

```
  <pose>1.93 1.67 0.0 0 0 0</pose>
```

```
</include>
```

```
<include>
```

```
  <uri>model://Red_pentagon</uri>
```

```
  <pose>2.74 3.56 0.0 0 0 0</pose>
```

```
</include>
```

```
<include>
```

```
  <uri>model://red_square</uri>
```

```
  <pose>3.11 0.84 0.0 0 0 0</pose>
```

```
</include>
```

```
<include>
```

```
  <uri>model://White_circle</uri>
```

```
  <pose>2.12 3.15 0.0 0 0 0</pose>
```

```
</include>
```

```
<include>
```

```
  <uri>model://Yellow_circle</uri>
```

```
  <pose>0.13 1.69 0.0 0 0 0</pose>
```

```
</include>
```

```
  <include>
```

```
  <uri>model://Yellow_rhombus</uri>
```

```
  <pose>2.07 0.7 0.0 0 0 0</pose>
```

```
</include>
```

Здесь первые два значения между тегами `<pose></pose>` - позиция X и Y на поле Gazebo, я вставлял рандомные значения, сгенерированные кодом на питоне в диапазоне x [0.1; 3.3] и y [0.1; 5.1], (чтобы фигуры не вылезали за поле аруко маркеров), также использовал ограничение на минимальное расстояние между точками в 0.5, чтобы фигуры не пересекались. Также сохраняем, выходим.

```

17 <include>
18   <uri-model>Blue_hexagon</uri>
19   <pose>0.22 2.35 0 0 0 0</pose>
20 </include>
21
22 <include>
23   <uri-model>Blue_rhombus</uri>
24   <pose>2.8 2.48 0 0 0 0</pose>
25 </include>
26
27 <include>
28   <uri-model>Green_pentagon</uri>
29   <pose>0.1 3.16 0 0 0 0</pose>
30 </include>
31
32 <include>
33   <uri-model>green_square</uri>
34   <pose>1.93 1.67 0 0 0 0</pose>
35 </include>
36
37 <include>
38   <uri-model>Red_pentagon</uri>
39   <pose>2.74 3.56 0 0 0 0</pose>
40 </include>
41
42 <include>
43   <uri-model>red_square</uri>
44   <pose>3.11 0.84 0 0 0 0</pose>
45 </include>
46
47 <include>
48   <uri-model>White_circle</uri>
49   <pose>2.12 3.15 0 0 0 0</pose>
50 </include>
51
52 <include>
53   <uri-model>Yellow_circle</uri>
54   <pose>0.13 1.69 0 0 0 0</pose>
55 </include>
56
57 <include>
58   <uri-model>Yellow_rhombus</uri>
59   <pose>2.07 0.7 0 0 0 0</pose>
60 </include>
61
62 <scene>
63   <name>0 0 0 0 0 0 1</name>
64 </scene>

```

```

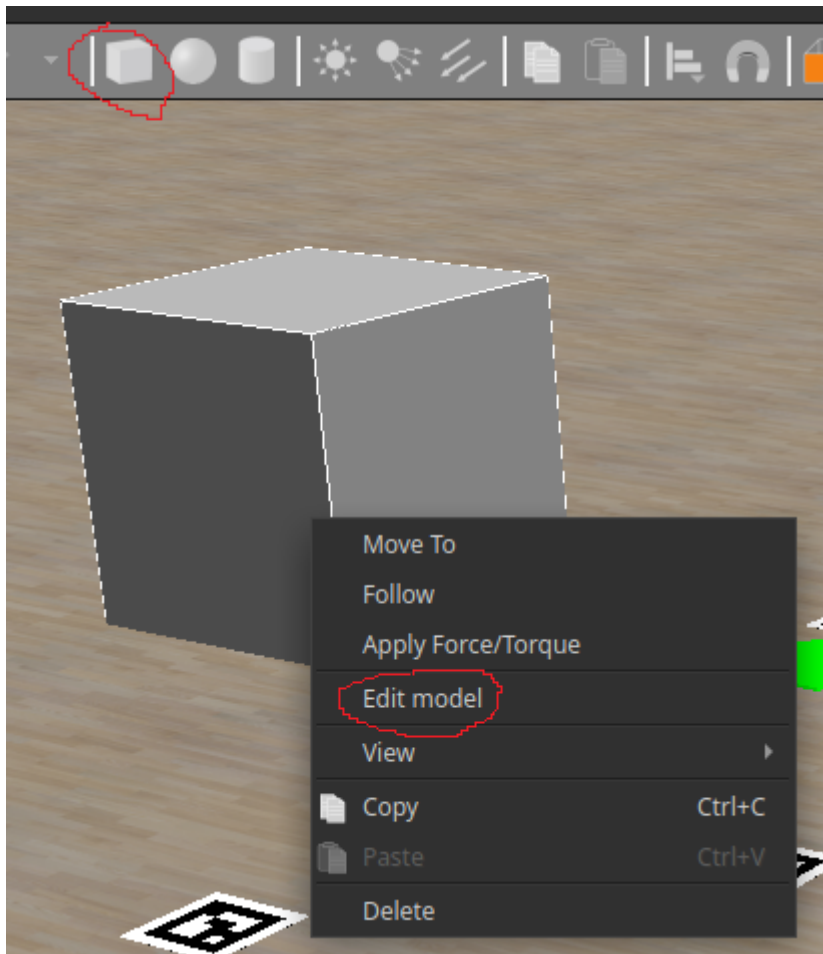
75 x = round(Random.uniform(0.1, 3.3), 2)
76 y = round(Random.uniform(0.1, 5.1), 2)
77 point = (x, y)
78
79 # Проверим, что расстояние между новой точкой и всеми существующими точками больше или равно min_d
80 if all((x - x1)**2 + (y - y1)**2 >= min_distance**2 for x1, y1 in coordinates):
81     coordinates.append(point)
82
83 return coordinates
84
85 # [генерация 9 пар координат]
86 num_points = 9
87 coordinates = generate_coordinates(num_points)
88

```

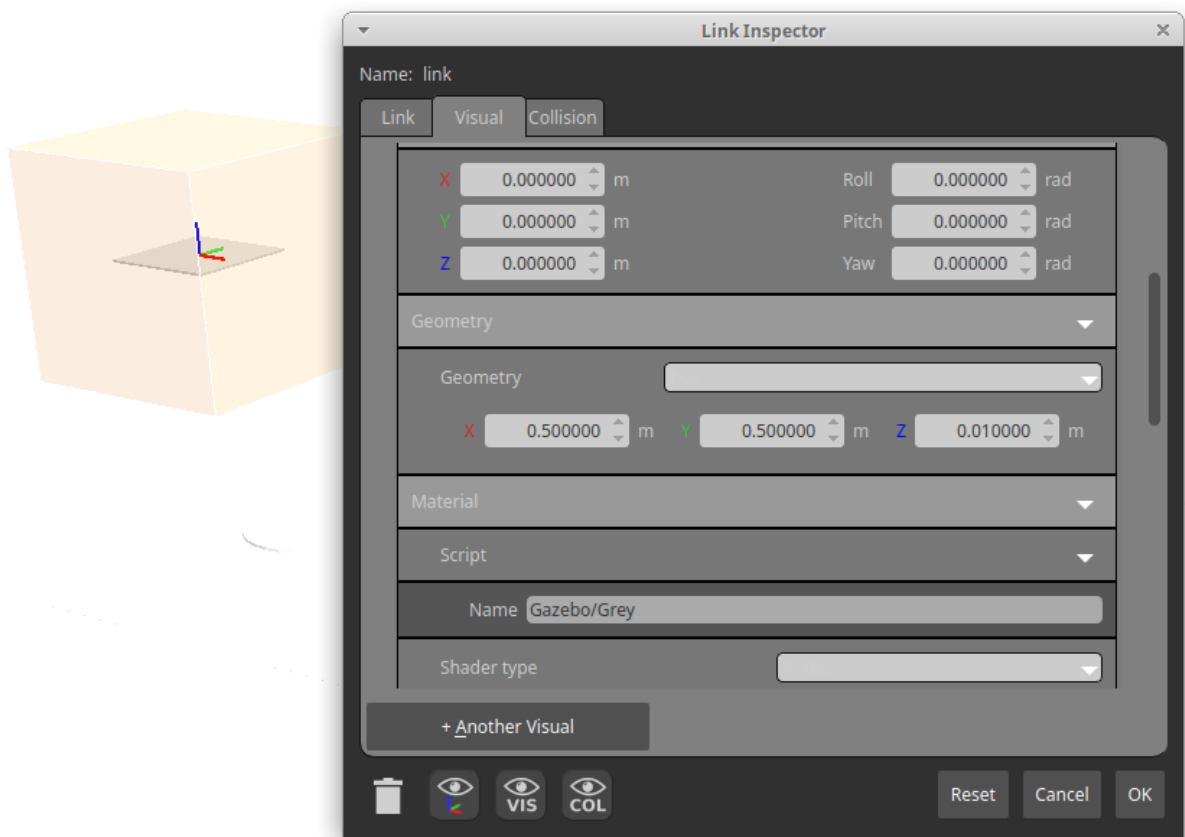
Создание фигур 1

Дальше самое интересное: создание фигур, для простоты квадраты и круги сделаем простыми средствами в gazebo: запустим любую карту, можно пустую. В меню сверху

возьмем куб, вставим, нажмем ПКМ на нем и 'edit model'

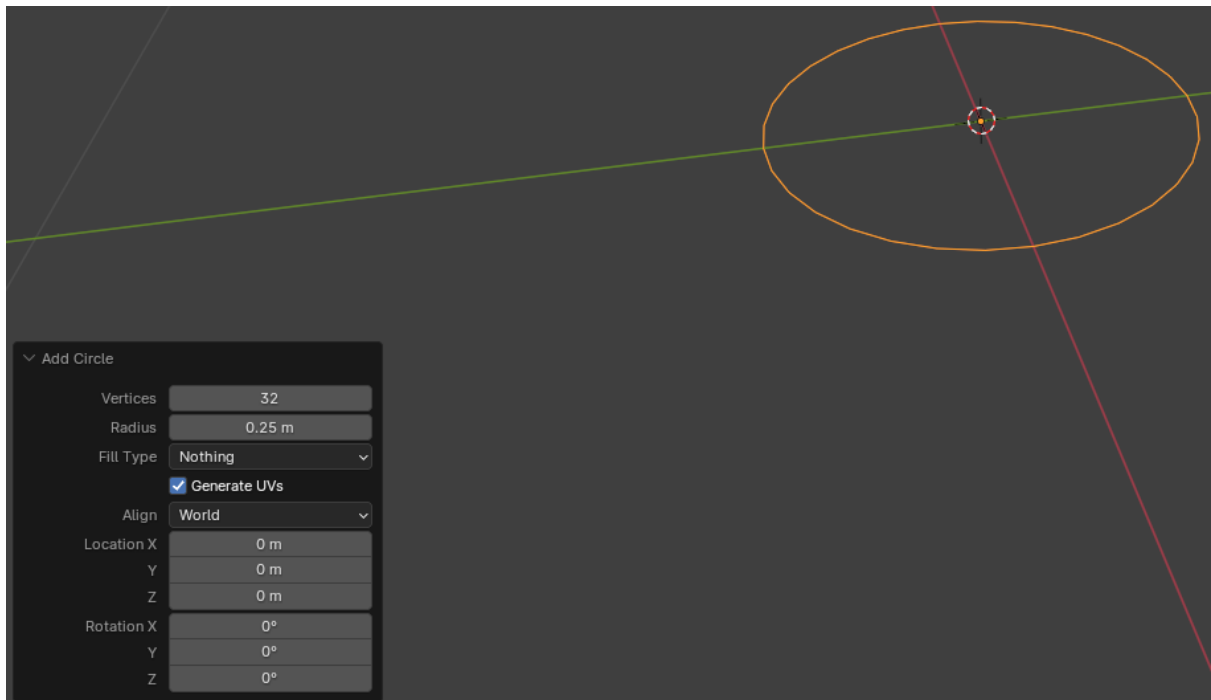


далее опять жмем ПКМ по кубу и открываем жмем на open link inspector, заходим во вкладку visual и меняем X, Y, Z на 0.5 0.5 0.01 соответственно.

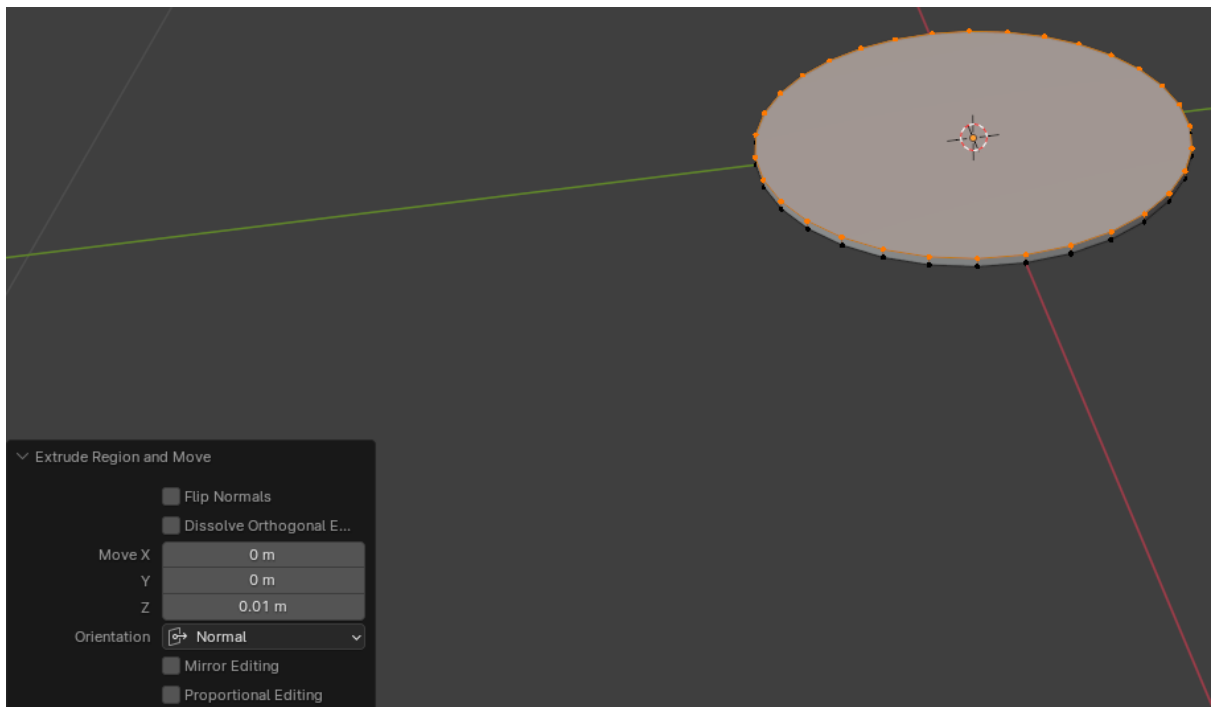


Жмем ок, и слева сверху file->exit model editor, сохраняем как green_square. Далее нам нужно будет единожды пройти по /home/clover/model_editor_models/, скопировать нашу папку, и перейти обратно в нашу постоянную рабочую директорию clover_simulator, дальше идем в ./models и вставляем туда папку, копируем ее, вставляем название и переименовываем копию в red_square, далее заходим в эту папку, открываем все файлы и все вхождения "green" меняем на "red".

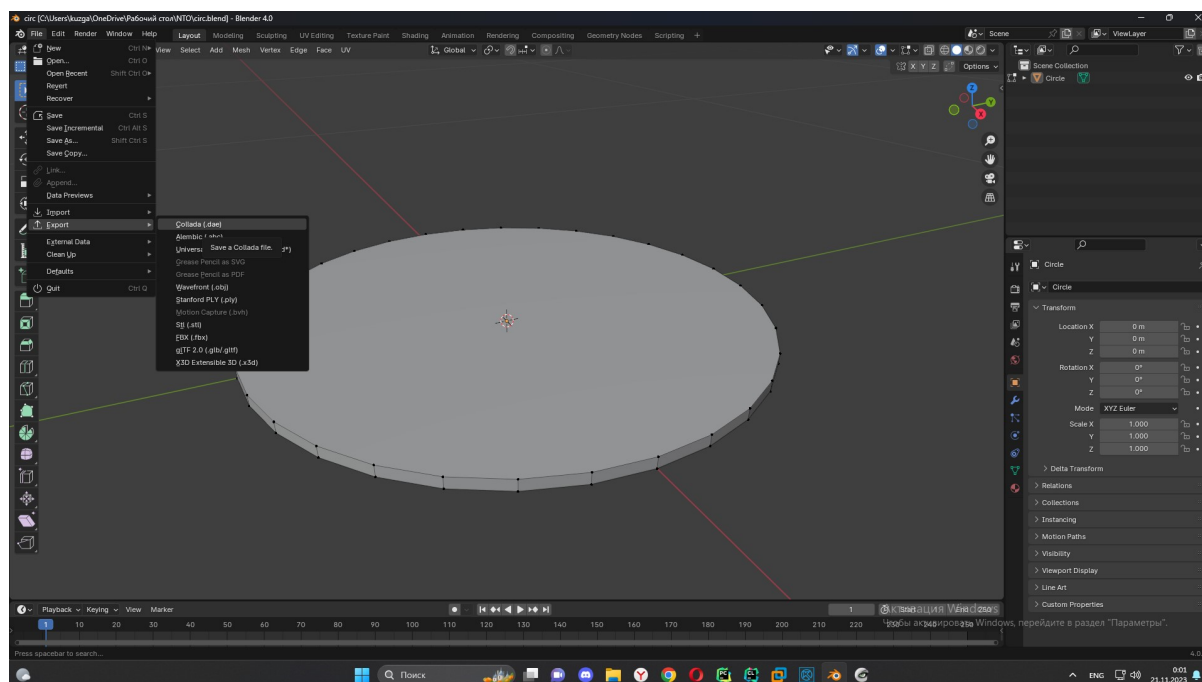
Дальше нам понадобится Blender, в нем мы создадим все оставшиеся фигуры, начнем с круга. Итак сначала удалим кубик - пример, сверху справа выделим все объекты мышкой и нажмем delete, далее на рабочем поле нажмем shift + A -> mesh -> circle, раскроем окно "Add circle" снизу сверху, поменяем "radius" на 0.25. Закрываем окно gazebo, в термине нажимаем Ctrl + c несколько раз, через секунду закрываем терминал.



Далее нажмем A, Tab, F, E, чуть поднимем курсор вверх, ЛКМ, поменяем значение в поле Z на 0.01m



Далее импортируем как файл Collada:

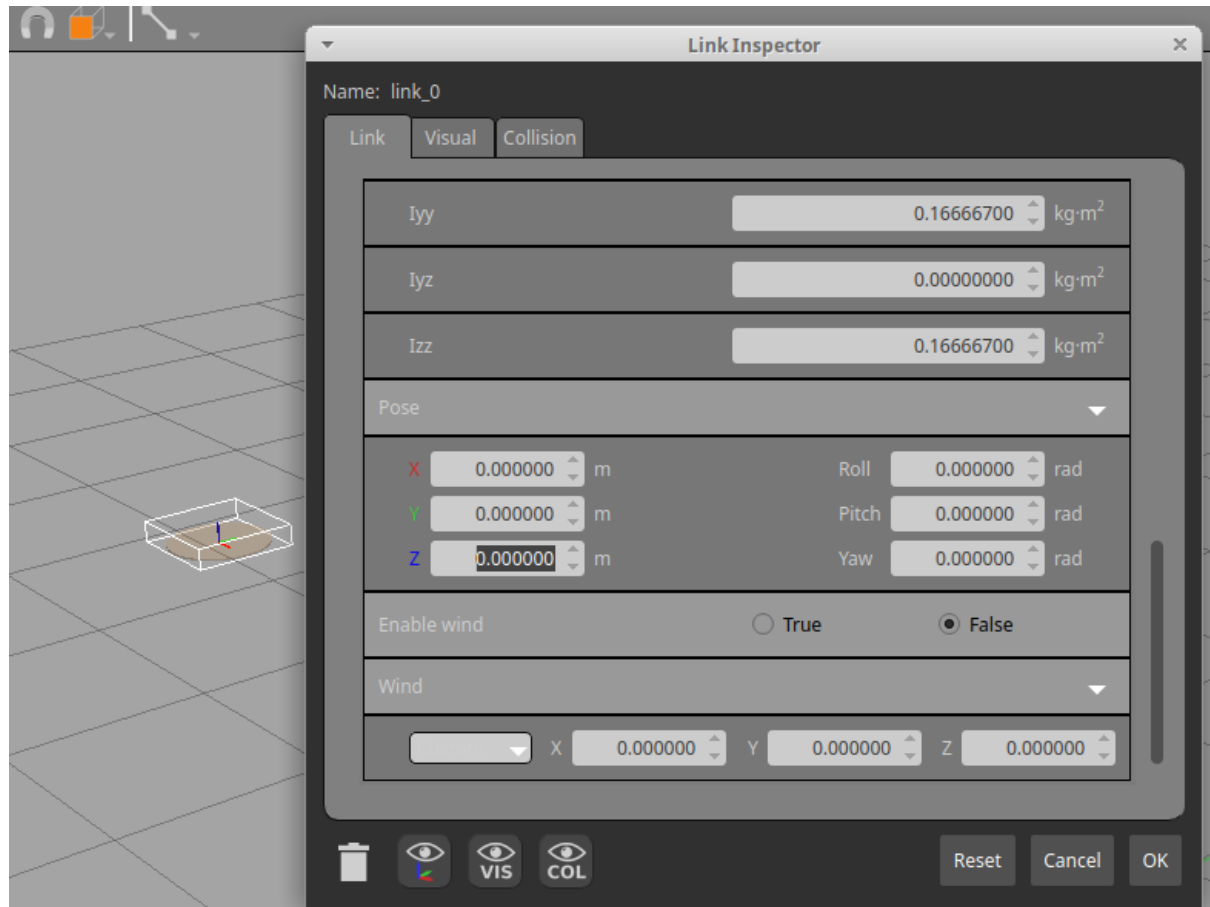


Таким же образом сделаем пяти/шестиугольник, и ромб, но в первой картинке количество вершин (32) поменяем на нужное нам (5,6,4) соответственно, также импортируем. По итогу получим 7 фигур, с размерами (по описанной окружности 0.5м), то есть все фигуры вписываются в квадрат со стороной 0.5м.

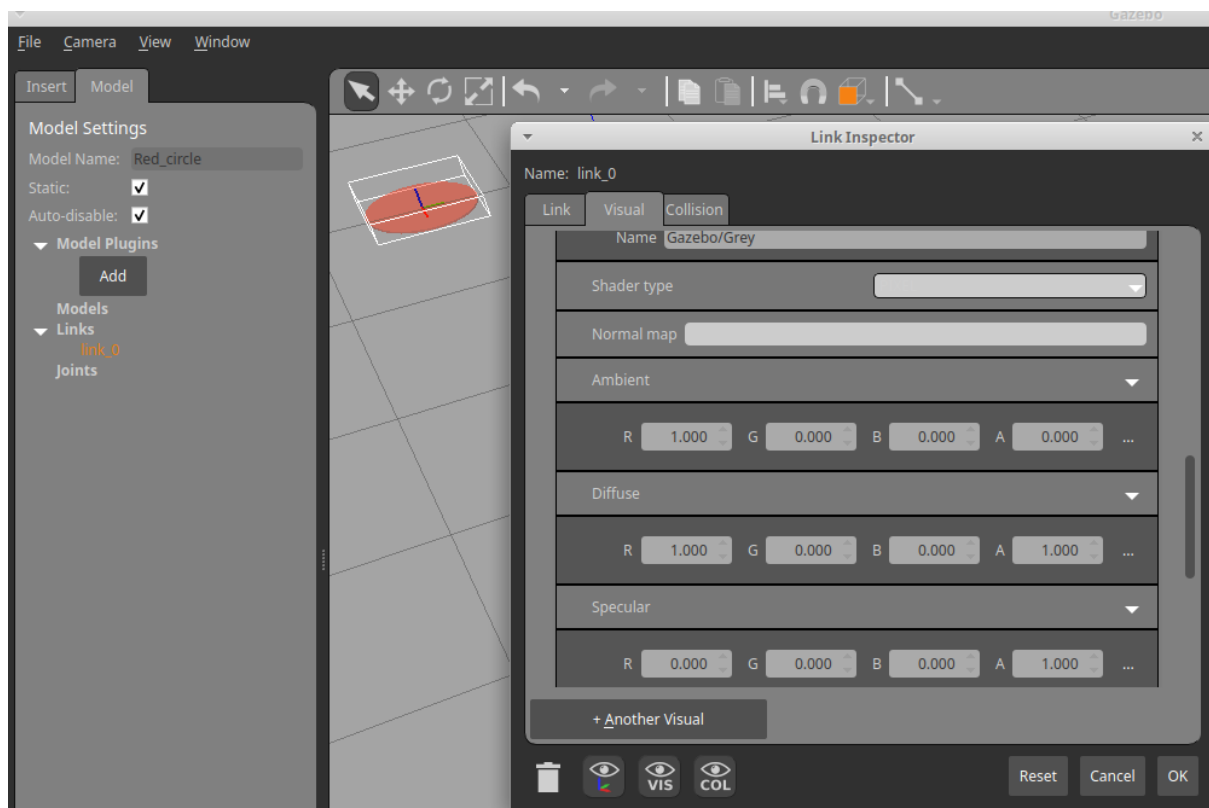
Создание фигур 2

Далее все эти файлы копируем на в нашу в `clover_simulation/models/DAE_data` стол виртуалки, запускаем терминал (`ctrl + alt + t`) пишем `"gazebo -paused"`, вставляем любую фигуру, далее также пкм редактировать модель, дальше нажимаем add shape и выбираем наш .dae файл с рабочего стола, добавляем, жмем пкм на куб, удаляем, далее жмем на нашу новую фигуру ПКМ, нажимаем `"open link inspector"` и на странице

link мотаем вниз, меняем все значения pose на нули. После



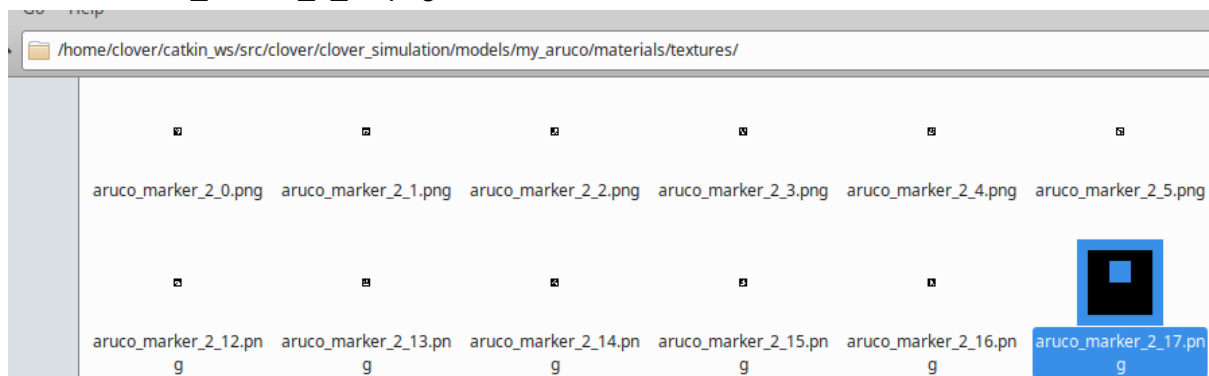
Далее переходим в этом же окне на вкладку Visual, будем менять цвет, в поле Deffuse, Ambient будем ставить единицу, на позицию нашего цвета (соответственно красный 1 0 0, зеленый 0 1 0, синий 0 0 1, желтый 1 1 0). После этого жмем О. Далее слева сверху нажимаем Model, ставим галочку на static, там же меняем имя в формате Color_figure (в данном случае это будет Red_circle). Затем слева сверху закрываем file editor, сохраняем по пути clover_simulator/models с таким же названием какое мы только что написали в Model.



Такую процедуру делаем для каждого нашего .dae файла, для каждого цвета.

Аруко маркеры

Для создания поля Аруко маркеров 5 по X и 7 по Y мы в models сделаем копию aruco_smit_txt и назовем ее my_aruco (выше мы уже подключили эту модель в мир), зайдём в папку и зайдём в файл. Там мы должны будем удалить все теги и их содержимое <visual>, у которых значение name="visual_marker_2_34" больше 34 (т. е. все от 35 до 99 включительно), далее мы должны будем поменять им все координаты. Итак 0-ой аруко маркер мы поставим в 0 0 0, следовательно 1ый мы поставим в 0.9 (расстояния между маркерами 0.6, между их центрами $0.6 + 0.15 + 0.15 = 0.9$) и тд до 4-го включительно, 5-ый поставим в 0 0.9, 6-ой в 0.9 0.9 и тд). Стоит заметить, что в стандартном наборе отсутствует аруко маркер 17, мы должны сделать его тут <https://chev.me/arucogen/> с параметром 4x4 и ID 17. Далее в фотошопе мы должны добавить ему белые границы, шириной и высотой в одну-шестую оригинальной картинки. Далее добавим эту картинку в models/my_aruco/textures с именем aruco_marker_2_17.png.



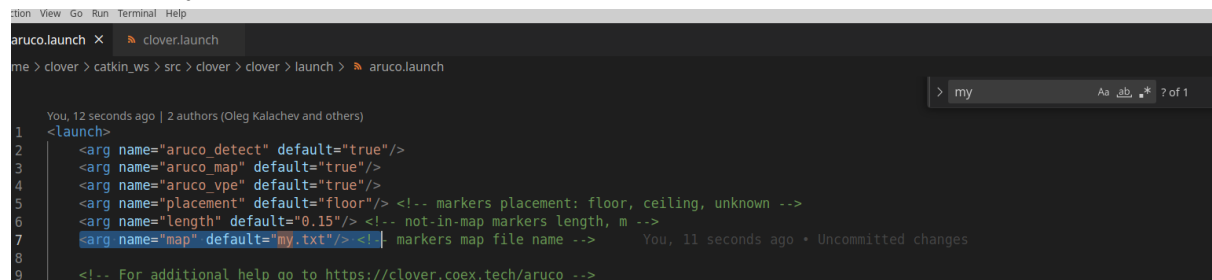
Также нужно в my_aruco/materials открыть файл в vs code и вставить [material aruco/marker_2_17](#)

```

{
  technique
  {
    pass
    {
      texture_unit
      {
        texture aruco_marker_2_17.png
        filtering none
        scale 1.0 1.0
      }
    }
  }
}

```

Также важная деталь: нужна написать свой txt карты с позицией каждого аруко маркера для построения видения карты: переходим по пути на скриншоте и сmit.txt меняем на my.txt



```

1 <launch>
2   <arg name="aruco_detect" default="true"/>
3   <arg name="aruco_map" default="true"/>
4   <arg name="aruco_vpe" default="true"/>
5   <arg name="placement" default="floor"/> <!-- markers placement: floor, ceiling, unknown -->
6   <arg name="length" default="0.15"/> <!-- not-in-map markers length, m -->
7   <arg name="map" default="my.txt"/> <!-- markers map file name -->
8   <!-- For additional help go to https://clover.coex.tech/aruco -->
9

```

Далее этот файл нужно будет солздать по пути как на скрине

```
aruco.launch my.txt x clover.launch
home > clover > catkin_ws > src > clover > aruco_pose > map > my.txt
1 # id length x y z rot_z rot_y rot_x
2 0 0.3 0.0 0.0 0 0 0
3 1 0.3 0.9 0.0 0 0 0
4 2 0.3 1.8 0.0 0 0 0
5 3 0.3 2.7 0.0 0 0 0
6 4 0.3 3.6 0.0 0 0 0
7 5 0.3 0.0 0.9 0 0 0
8 6 0.3 0.9 0.9 0 0 0
9 7 0.3 1.8 0.9 0 0 0
10 8 0.3 2.7 0.9 0 0 0
11 9 0.3 3.6 0.9 0 0 0
12 10 0.3 0.0 1.8 0 0 0
13 11 0.3 0.9 1.8 0 0 0
14 12 0.3 1.8 1.8 0 0 0
15 13 0.3 2.7 1.8 0 0 0
16 14 0.3 3.6 1.8 0 0 0
17 15 0.3 0.0 2.7 0 0 0
18 16 0.3 0.9 2.7 0 0 0
19 17 0.3 1.8 2.7 0 0 0
20 18 0.3 2.7 2.7 0 0 0
21 19 0.3 3.6 2.7 0 0 0
22 20 0.3 0.0 3.6 0 0 0
23 21 0.3 0.9 3.6 0 0 0
24 22 0.3 1.8 3.6 0 0 0
25 23 0.3 2.7 3.6 0 0 0
26 24 0.3 3.6 3.6 0 0 0
27 25 0.3 0.0 4.5 0 0 0
28 26 0.3 0.9 4.5 0 0 0
29 27 0.3 1.8 4.5 0 0 0
30 28 0.3 2.7 4.5 0 0 0
31 29 0.3 3.6 4.5 0 0 0
32 30 0.3 0.0 5.4 0 0 0
33 31 0.3 0.9 5.4 0 0 0
34 32 0.3 1.8 5.4 0 0 0
35 33 0.3 2.7 5.4 0 0 0
36 34 0.3 3.6 5.4 0 0 0
```

| # id | length | x | y | z | rot_z | rot_y | rot_x |
|------|--------|---|-----|-----|-------|-------|-------|
| 0 | 0.3 | | 0.0 | 0.0 | 0 | 0 | 0 |
| 1 | 0.3 | | 0.9 | 0.0 | 0 | 0 | 0 |
| 2 | 0.3 | | 1.8 | 0.0 | 0 | 0 | 0 |
| 3 | 0.3 | | 2.7 | 0.0 | 0 | 0 | 0 |
| 4 | 0.3 | | 3.6 | 0.0 | 0 | 0 | 0 |
| 5 | 0.3 | | 0.0 | 0.9 | 0 | 0 | 0 |
| 6 | 0.3 | | 0.9 | 0.9 | 0 | 0 | 0 |
| 7 | 0.3 | | 1.8 | 0.9 | 0 | 0 | 0 |

| | | | | | | | |
|----|-----|-----|-----|---|---|---|---|
| 8 | 0.3 | 2.7 | 0.9 | 0 | 0 | 0 | 0 |
| 9 | 0.3 | 3.6 | 0.9 | 0 | 0 | 0 | 0 |
| 10 | 0.3 | 0.0 | 1.8 | 0 | 0 | 0 | 0 |
| 11 | 0.3 | 0.9 | 1.8 | 0 | 0 | 0 | 0 |
| 12 | 0.3 | 1.8 | 1.8 | 0 | 0 | 0 | 0 |
| 13 | 0.3 | 2.7 | 1.8 | 0 | 0 | 0 | 0 |
| 14 | 0.3 | 3.6 | 1.8 | 0 | 0 | 0 | 0 |
| 15 | 0.3 | 0.0 | 2.7 | 0 | 0 | 0 | 0 |
| 16 | 0.3 | 0.9 | 2.7 | 0 | 0 | 0 | 0 |
| 17 | 0.3 | 1.8 | 2.7 | 0 | 0 | 0 | 0 |
| 18 | 0.3 | 2.7 | 2.7 | 0 | 0 | 0 | 0 |
| 19 | 0.3 | 3.6 | 2.7 | 0 | 0 | 0 | 0 |
| 20 | 0.3 | 0.0 | 3.6 | 0 | 0 | 0 | 0 |
| 21 | 0.3 | 0.9 | 3.6 | 0 | 0 | 0 | 0 |
| 22 | 0.3 | 1.8 | 3.6 | 0 | 0 | 0 | 0 |
| 23 | 0.3 | 2.7 | 3.6 | 0 | 0 | 0 | 0 |
| 24 | 0.3 | 3.6 | 3.6 | 0 | 0 | 0 | 0 |
| 25 | 0.3 | 0.0 | 4.5 | 0 | 0 | 0 | 0 |
| 26 | 0.3 | 0.9 | 4.5 | 0 | 0 | 0 | 0 |
| 27 | 0.3 | 1.8 | 4.5 | 0 | 0 | 0 | 0 |
| 28 | 0.3 | 2.7 | 4.5 | 0 | 0 | 0 | 0 |
| 29 | 0.3 | 3.6 | 4.5 | 0 | 0 | 0 | 0 |
| 30 | 0.3 | 0.0 | 5.4 | 0 | 0 | 0 | 0 |
| 31 | 0.3 | 0.9 | 5.4 | 0 | 0 | 0 | 0 |
| 32 | 0.3 | 1.8 | 5.4 | 0 | 0 | 0 | 0 |
| 33 | 0.3 | 2.7 | 5.4 | 0 | 0 | 0 | 0 |
| 34 | 0.3 | 3.6 | 5.4 | 0 | 0 | 0 | 0 |

На этом все, закрываем все лишнее, нажимаем на ранее созданный ярлык и у нас открывается наш новый gazebo мир с подключенным ROS'ом

