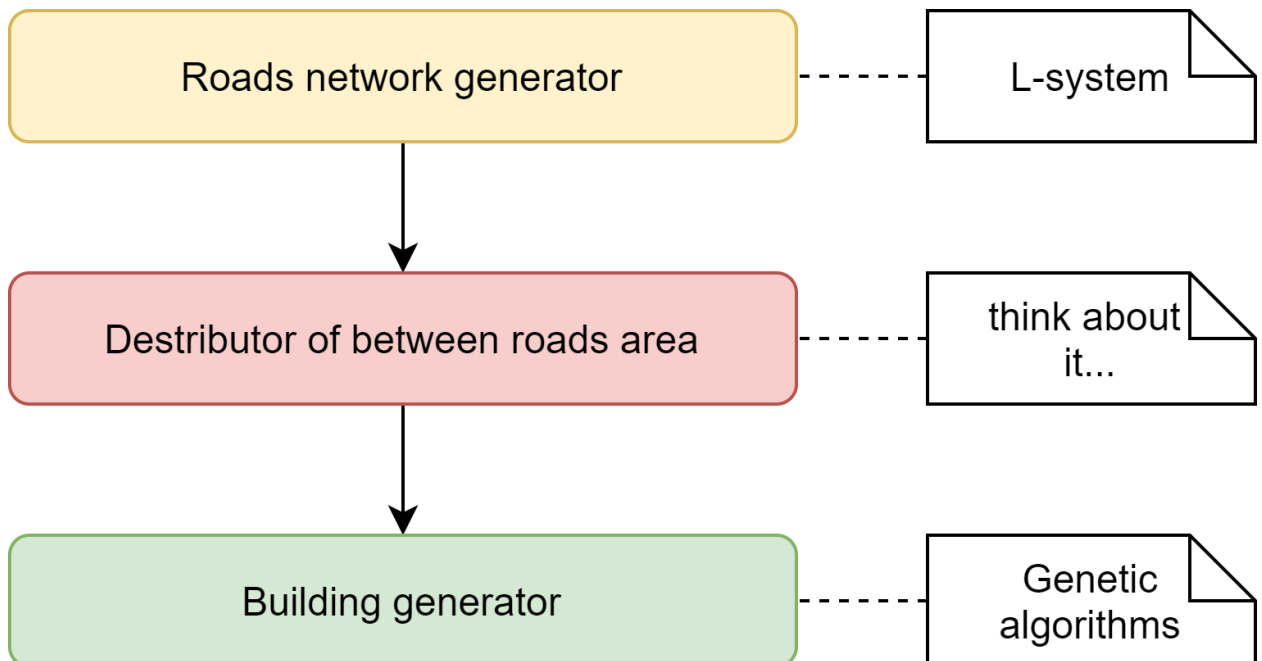


## 1. План системы

При реализации демо было принято решение следовать следующему укрупненному плану:



Это классическое разбиение генерации города на этапы. Сначала генерируется дорожная сеть, после полученные участки территорий, образованных дорогами, разделяются между домами. Далее дома генерируются на основе ограничений, полученных на этапе разделения территории.

## 2. Выбор формализма

Существует подход, предлагающий произвести ряд манипуляций над грамматикой для L-системы из оригинальной статьи (слияние некоторых правил, избавление от контекстной зависимости и т.д.), позволяющих избавиться от формализма L-систем в пользу приоритетной очереди.

### L-система:

```
w: R(0, initialRuleAttr) ?I(initRoadAttr, UNASSIGNED)
p1: R(del, ruleAttr) : del<0 -> e
p2: R(del, ruleAttr) > ?I(roadAttr, state) : state==SUCCEED
    { globalGoals(ruleAttr, roadAttr) creates the parameters for:
      pDel[0-2], pRuleAttr[0-2], pRoadAttr[0-2] }
    -> +(roadAttr.angle)F(roadAttr.length)
      B(pDel[1], pRuleAttr[1], pRoadAttr[1]),
      B(pDel[2], pRuleAttr[2], pRoadAttr[2]),
      R(pDel[0], pRuleAttr[0]) ?I(pRoadAttr[0], UNASSIGNED) [i]
p3: R(del, ruleAttr) > ?I(roadAttr, state) : state==FAILED -> e
p4: B(del, ruleAttr, roadAttr) : del>0 -> B(del-1, ruleAttr, roadAttr)
p5: B(del, ruleAttr, roadAttr) : del==0 -> [R(del, ruleAttr)?I(roadAttr, UNASSIGNED)]
p6: B(del, ruleAttr, roadAttr) : del<0 -> e
p7: R(del, ruleAttr) < ?I(roadAttr, state) : del<0 -> e
p8: ?I(roadAttr, state) : state==UNASSIGNED
    { localConstraints(roadAttr) adjusts the parameters for:
      state, roadAttr }
    -> ?I(roadAttr, state)
p9: ?I(roadAttr, state) : state!=UNASSIGNED -> e
```

### Приоритетная очередь:

```
initialize priority queue Q with a single entry:  
  r(0,ra,qa)  
  
initialize segment list S to empty  
  
until Q is empty  
  pop smallest r(t,ra,qa) from Q  
  compute (nqa, state) = localConstraints(qa)  
  if (state == SUCCEED) {  
    add segment(ra) to S  
    compute (pdel[3], nqa[3], nra[3]) = globalGoals(qa,ra)  
    if (pdel[0] >= 0) add r(t+1+pdel[0], nra[0], nqa[0]) to Q  
    if (pdel[1] >= 0) add r(t+1+pdel[1], nra[1], nqa[1]) to Q  
    if (pdel[2] >= 0) add r(t+1+pdel[2], nra[2], nqa[2]) to Q
```

Основная логика генерации при этом концентрируется в методах `localConstraints` и `globalGoals`.

### 3. Инструменты для реализации демо проекта

Чтобы упростить задачу и быстрее проверить гипотезу было решено отказаться от тяжеловесных инструментов вроде UE или Houdini в пользу чего-то более простого и легковесного. В результате проб и обсуждений в качестве такого инструмента был выбран OpenGL.

