

Визуальная Айвика 9

(англ. VisualAivika 9)

МОДЕЛИРОВАТЬ В ФУНКЦИОНАЛЬНОМ СТИЛЕ [HTTPS://VISUALAIVIKA.RU](https://visualaivika.ru)
СОРОКИН ДАВИД ЭРНЕСТОВИЧ <DAVSOR@MAIL.RU>

Визуальная Айвика 9

▶ Системная динамика

- ▶ Система ОДУ (конечный автомат, метод Рунге-Кутты)

▶ Пример

- ▶ `A = integ(R, InitialA);`
`InitialA = 1000;`
`R = ka * A;`
`ka = 0.07;`

▶ Дискретно-событийное моделирование

- ▶ Система массового обслуживания (заявки, очереди, ресурсы, вычисления блоков)

▶ Пример

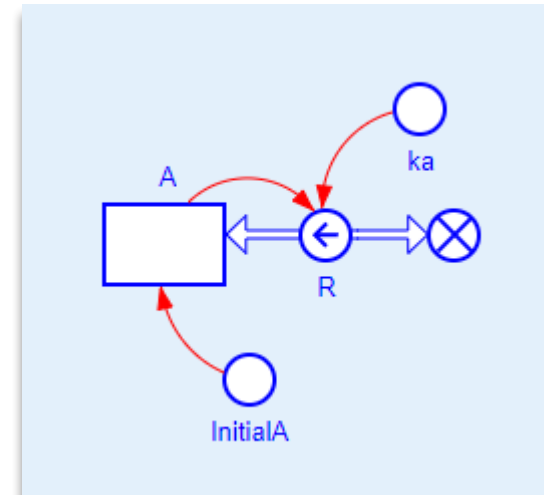
- ▶ `P3 = Block.preempt(Prof) >>> P4;`
`P4 = Block.advance(exponential(200)) >>> P5;`
`P5 = Block.return(Prof) >>> Block.terminate;`
`Prof = Facility.create();`

Визуальная Айвика 9 системная динамика

Пример

```
A = integ(R, InitialA);  
InitialA = 1000;  
R = ka * A;  
ka = 0.07;
```

Диаграмма потоков и накопителей



Уравнения и диаграмма – одно целое

«Визуальная Айвика» разрешает систему уравнений и на-лету строит вычислительный граф, по которому порождает код на C# для численного интегрирования с помощью конечного автомата

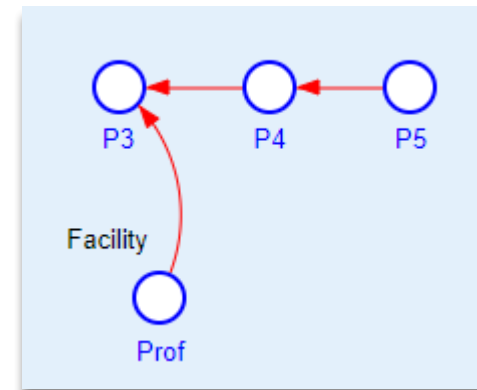
Визуальная Айвика 9

система массового обслуживания

Пример

```
P3 = Block.preempt(Prof) >>> P4;  
P4 = Block.advance(exponential(200)) >>> P5;  
P5 = Block.return(Prof) >>> Block.terminate;  
Prof = Facility.create();
```

Обработка заявок



Блоки P3, P4 и P5 – это вычисления, которые скомбинированы с помощью композиции

«Визуальная Айвика» строит вычислительный граф и порождает код на C# для запуска этих вычисления внутри конечного автомата

Визуальная Айвика 9

И система ОДУ, и система массового обслуживания преобразуются в один и тот же конечный автомат

Этот конечный автомат управляется очередью событий, что позволяет комбинировать обе парадигмы в рамках одной и той же имитационной модели

Визуальная Айвика 9

ЧИСЛЕННОЕ ИНТЕГРИРОВАНИЕ
СИСТЕМЫ УРАВНЕНИЙ
ЕСТЕСТВЕННЫМ ОБРАЗОМ
ВЫРАЖАЕТСЯ ЧЕРЕЗ КОНЕЧНЫЙ
АВТОМАТ

НО ЧЕМ ТОГДА ЯВЛЯЮТСЯ
ВЫЧИСЛЕНИЯ БЛОКОВ?

Визуальная Айвика 9

моделировать в функциональном стиле

Вычисление блока является «стрелкой» Клейсли

Стрелка построена на основе монады «продолжений»

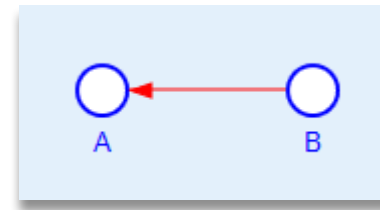
Продолжения привязаны к очереди событий

Но вы **можете использовать все это** без глубокого понимания функционального программирования!

Визуальная Айвика 9

Композиция блоков – это просто композиция «стрелок» (в терминах языка Haskell)

Композиция блоков



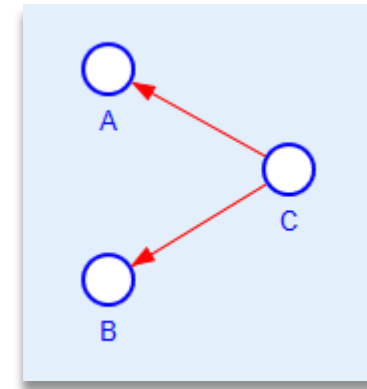
$A = block_1 \gg\gg block_2 \gg\gg B;$

Здесь мы создаем новый блок A, где каждая заявка сначала обрабатывается блоком $block_1$, затем – блоком $block_2$, а потом – блоком B

Визуальная Айвика 9

При слиянии блоков мы просто используем общий экземпляр вычисления одного и того же блока

Слияние блоков



```
A = Block.identity >>> C;  
B = Block.identity >>> C;
```

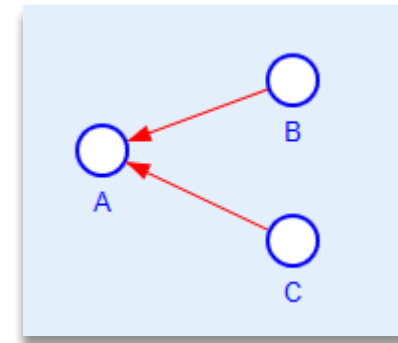
После начала обработки заявки блоком А она направляется в блок С

Снова, после начала обработки заявки блоком В она также направляется в блок С

Визуальная Айвика 9

Мы можем проверять условие для каждой заявки, чтобы выбрать следующий блок

Выбор следующих блоков



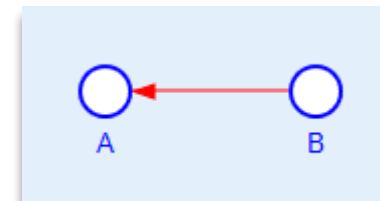
```
A = Block.select(if transact.Count > 7  
then B else C);
```

Здесь если значение некоторого атрибута Count больше, чем 7, то тогда заявка идет в блок B; иначе заявка обрабатывается блоком C

Визуальная Айвика 9

Задержка – это вычисление блока, которое задерживает каждую заявку на заданный интервал времени

Задержка заявок



```
A = Block.advance(random(1, 10)) >>> B;
```

Здесь мы задерживаем каждую заявку на случайный интервал времени от 1 до 10. Затем заявка обрабатывается блоком B

Визуальная Айвика 9

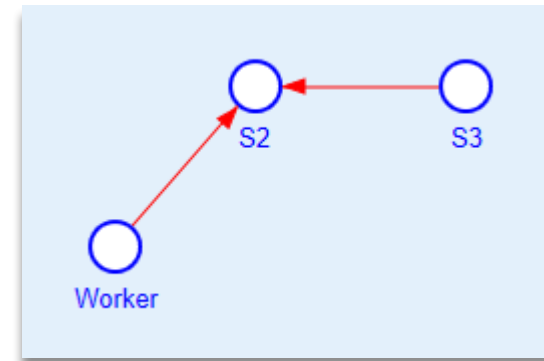
НО ЧТО ТОГДА НА СЧЕТ
РЕСУРСОВ И ОЧЕРЕДЕЙ?

Визуальная Айвика 9

Ресурсы и очереди – это сущности, которые могут быть использованы внутри вычислений блоков

Здесь роль таких сущностей вторична в отличие от вычислений блоков

Обработка ресурсов и очередей



```
Worker = Facility.create();
```

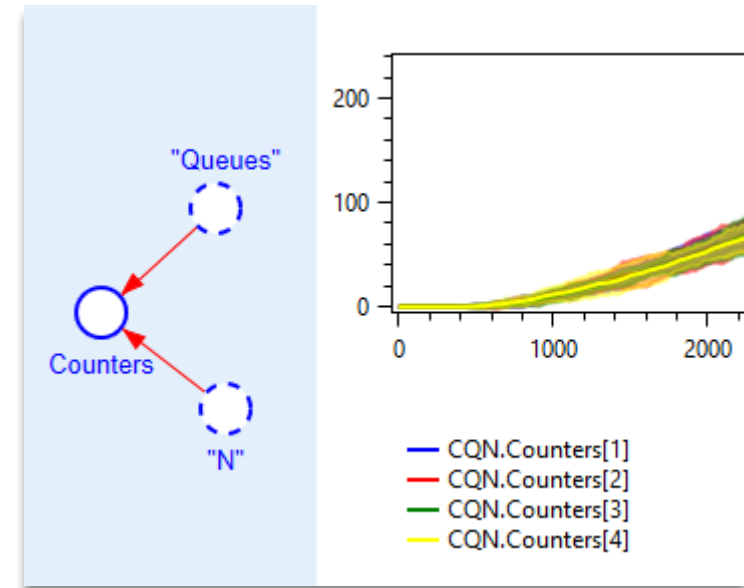
```
S2 = Block.seize(Worker) >>>  
Block.advance(random(5, 11)) >>>  
Block.release(Worker) >>> S3;
```

Здесь мы захватываем ресурс, случайно задерживаем заявку, а затем освобождаем ресурс до перехода к блоку S3

Визуальная Айвика 9

Ресурсы и очереди собирают статистику, которая может быть отображена на графиках

Вывод статистики для ресурсов и очередей



Counters =

```
[ Queue.enqueueCount(Queues[i]) | i <- 1..N ];
```

Здесь мы отображаем массив счетчиков (количество помещений в очередь) для массива очередей

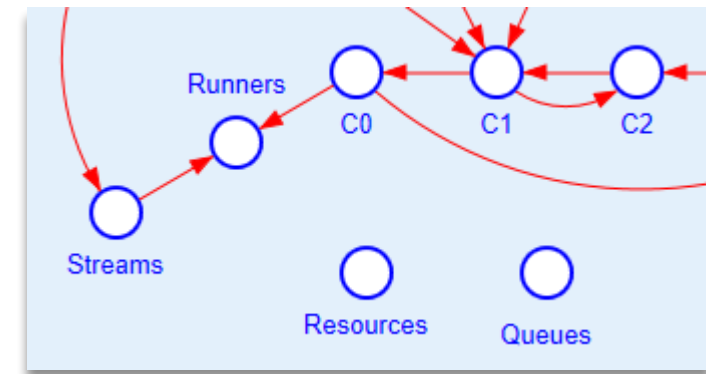
Визуальная Айвика 9

НО КАК ТОГДА НАЧАТЬ
ОБРАБОТКУ ЗАЯВОК
ВЫЧИСЛЕНИЯМИ БЛОКОВ?

Визуальная Айвика 9

Заявки рождаются из потоков внешних событий

Создание вычислений потоков



Streams =

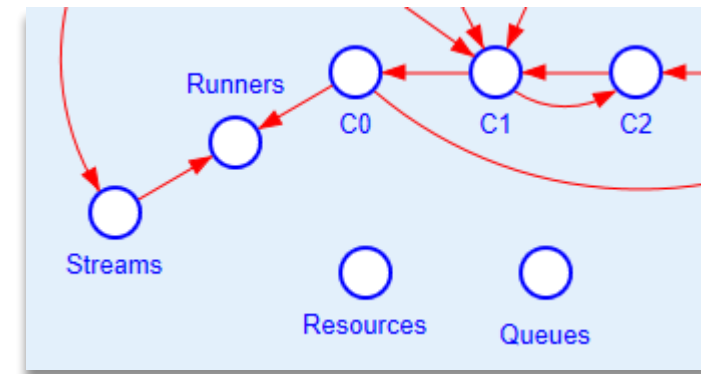
```
[ Stream.take(Stream.randomInt(0, 0),  
  Initial_Job_Count_Per_Tandem) | i <- 1..N ]
```

В начале мы должны создать поток внешних событий. Здесь мы создаем массив потоков по заданному числу событий

Визуальная Айвика 9

Мы берем входной поток внешних событий и начинаем создавать заявки для того, чтобы они были обработаны заданным вычислением блока

Порождение заявок



```
Runners =  
  [ do! Block.runByStream(Streams[i], C0[i]) |  
    i <- 1..N ];
```

Эта конструкция создает заявки по заданному массиву потоков и затем начинает обрабатывать такие заявки по заданным вычислениям блоков (массив C0)

Визуальная Айвика 9

А ЧТО ДЕЛАТЬ, ЕСЛИ НЕТ
ГОТОВЫХ ВЫЧИСЛЕНИЙ
БЛОКОВ?

Визуальная Айвика 9

Вы можете **расширить свою имитационную модель** с помощью произвольного кода .NET (F# и C#) на основе IronAivika [<https://gitflic.ru/project/dsorokin/iron-aivika>]

Пример

```
F =  
  [<assembly("lib/DelayBlockFunction.dll")>]  
  [<class(VisualAivika.Demo.DelayBlockFunction)>]  
  extern fun (A: Block): Block;
```

```
Y1 = F(Block.advance(10) >>> Block.terminate);
```

Визуальная Айвика 9

Более того, вы **можете экспортировать свою имитационную модель** как приложение .NET

Такие экспортированные приложения не будут зависеть от визуальной среды «Визуальная Айвика». Вы сможете контролировать как и когда собирать и запускать такие приложения

Визуальная Айвика 9

моделировать в функциональном стиле

Сорокин Давид Эрнестович (Россия, Йошкар-Ола)

davsor@mail.ru

[HTTPS://visualaivika.ru](https://visualaivika.ru)