Analysis of Process Models: Introduction, state space analysis and simulation in CPN Tools

prof.dr.ir. Wil van der Aalst

Technische Universiteit
Eindhoven
University of Technology

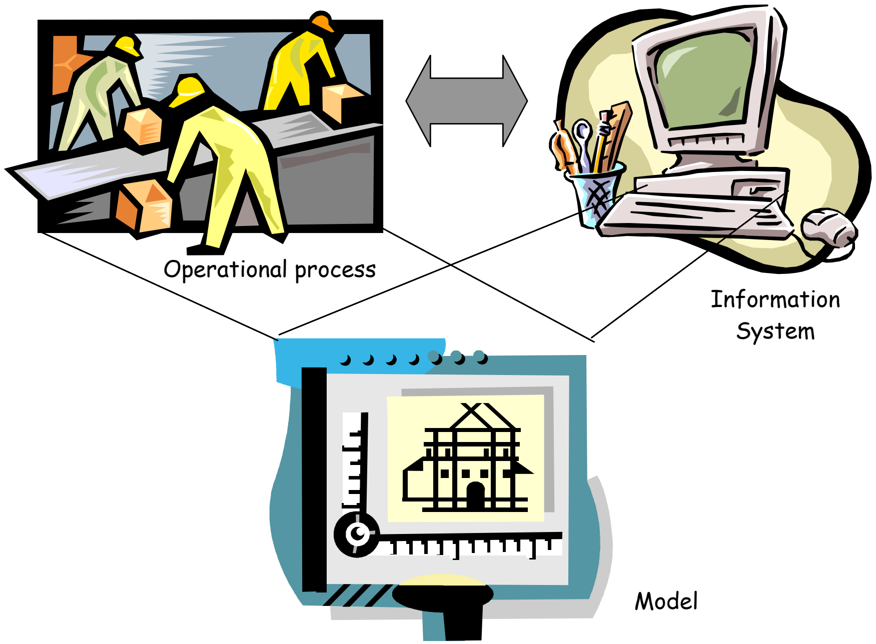
Where innovation starts

What is a Petri net?

- **A graphical notion** **(model = picture?)**
- **A mathematical notion** **(model = graph?)**
- **A programming notion** **(model = program?)**

- **A solver independent medium**
- **Starting point for a variety of analysis approaches**

Analysis



- Analysis is typically model-driven to allow e.g. what-if questions.
- Models of both operational processes and/or the information systems can be analyzed.
- Types of analysis:
  - *validation*
  - *verification*
  - *performance analysis*

Three types of analysis techniques

1. **Reachability/coverability graph**
2. **Structural techniques**
   - **Place and transition invariants**
   - **Marking equation**
   - **Traps, siphons, etc.**
3. **Simulation**

- **Each can be applied to both classical and high-level Petri nets.**
- **Nevertheless, for the second we restrict ourselves to classical Petri nets.**

Mapping technique/use:

- **reachability graph (validation, verification)**
- **invariants (validation, verification)**
- **simulation (validation, performance analysis)**

Informal introduction ...

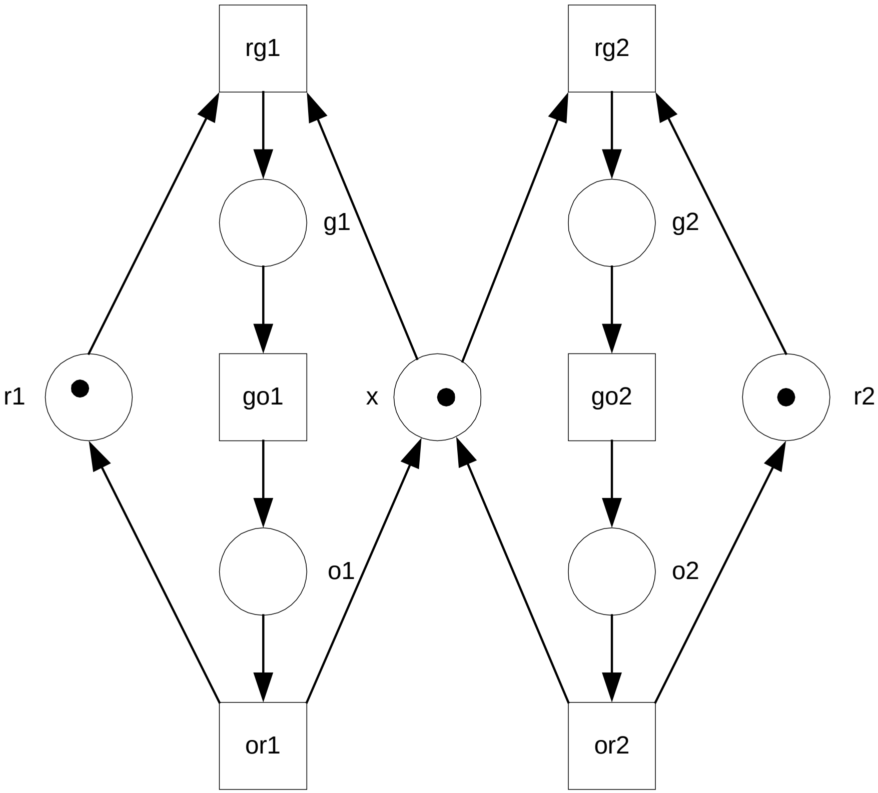
Technische Universiteit
Eindhoven
University of Technology

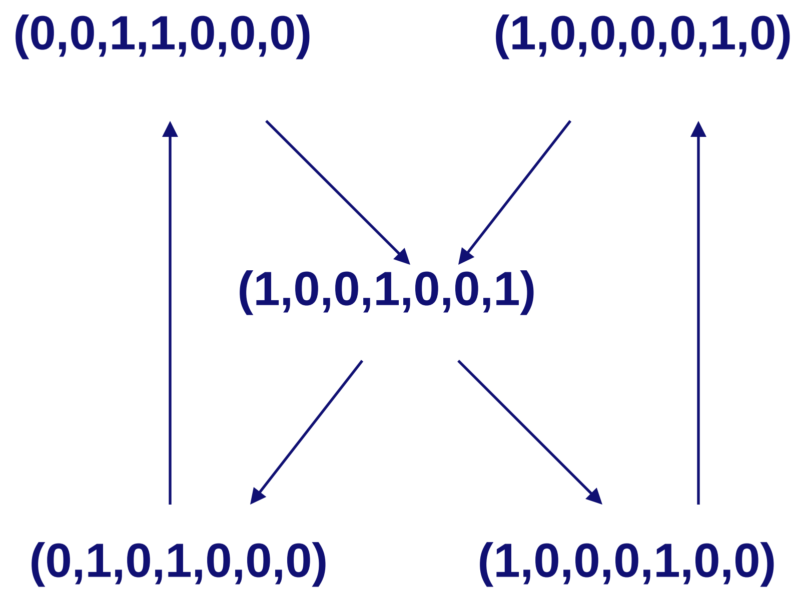
Where innovation starts

Examples of generic questions given a marked Petri net

- **terminating**
  it has only finite occurrence sequences
- **deadlock-free**
  each reachable marking enables a transition
- **live**
  each reachable marking enables an occurrence sequence containing all transitions
- **bounded**
  each place has an upper bound that holds for all reachable markings
- **1-safe**
  1 is a bound for each place p
- **reversible**
  m_0 is reachable from each reachable marking, i.e., the initial marking is a so-called home marking.

Reachability graph

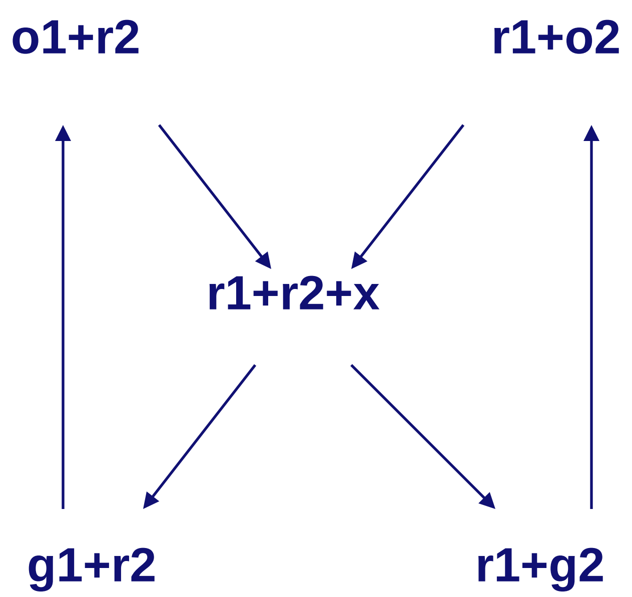




Five reachable states.
Traffic lights are safe!

Alternative notation

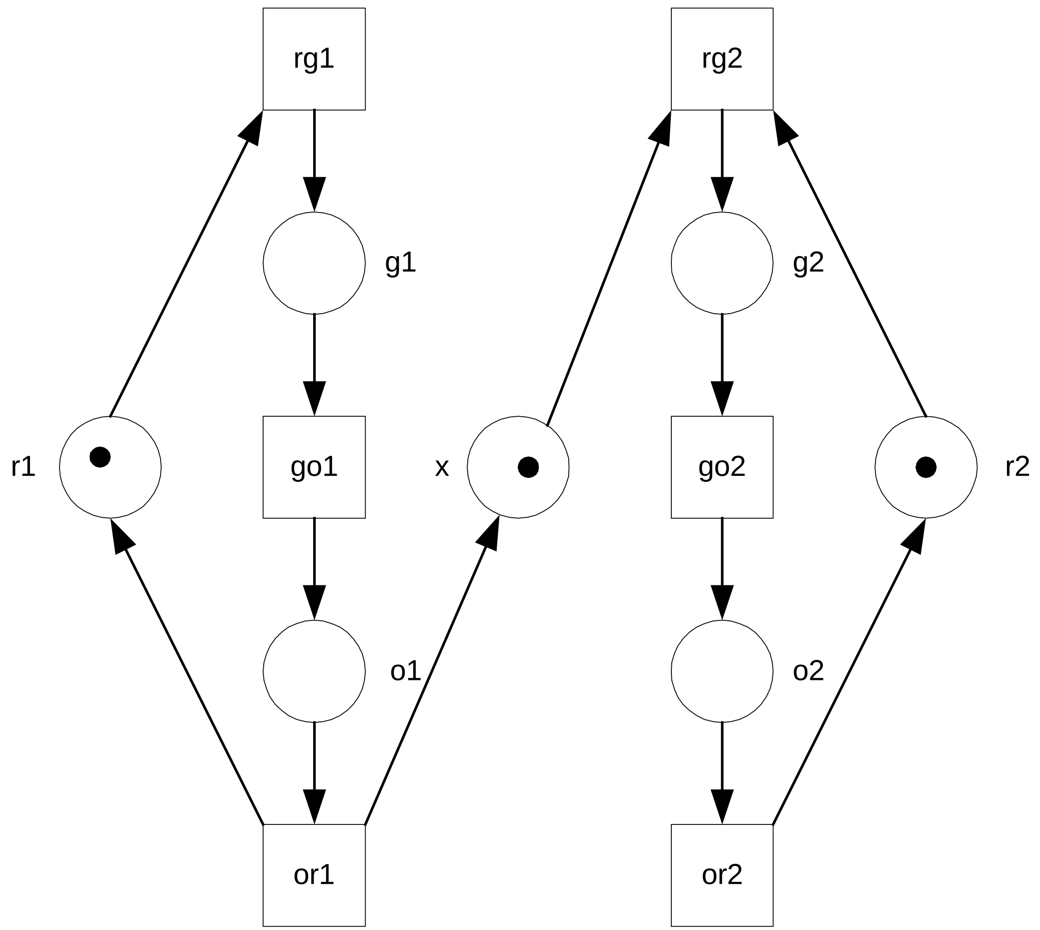
- **terminating**
  it has only finite occurre
  sequences
- **deadlock-free**
  each reachable marking es a
  transition
- **live**
  each reachable marking e es an
  occurrence sequence co g all
  transitions
- **bounded**
  each place has an upper that
  holds for all reachable ma s
- **1-safe**
  1 is a bound for each plac
- **reversible**
  m_0 is reachable from each hable
  marking, i.e., the initial m is a
  so-called home marking.



Reachability graph (2)

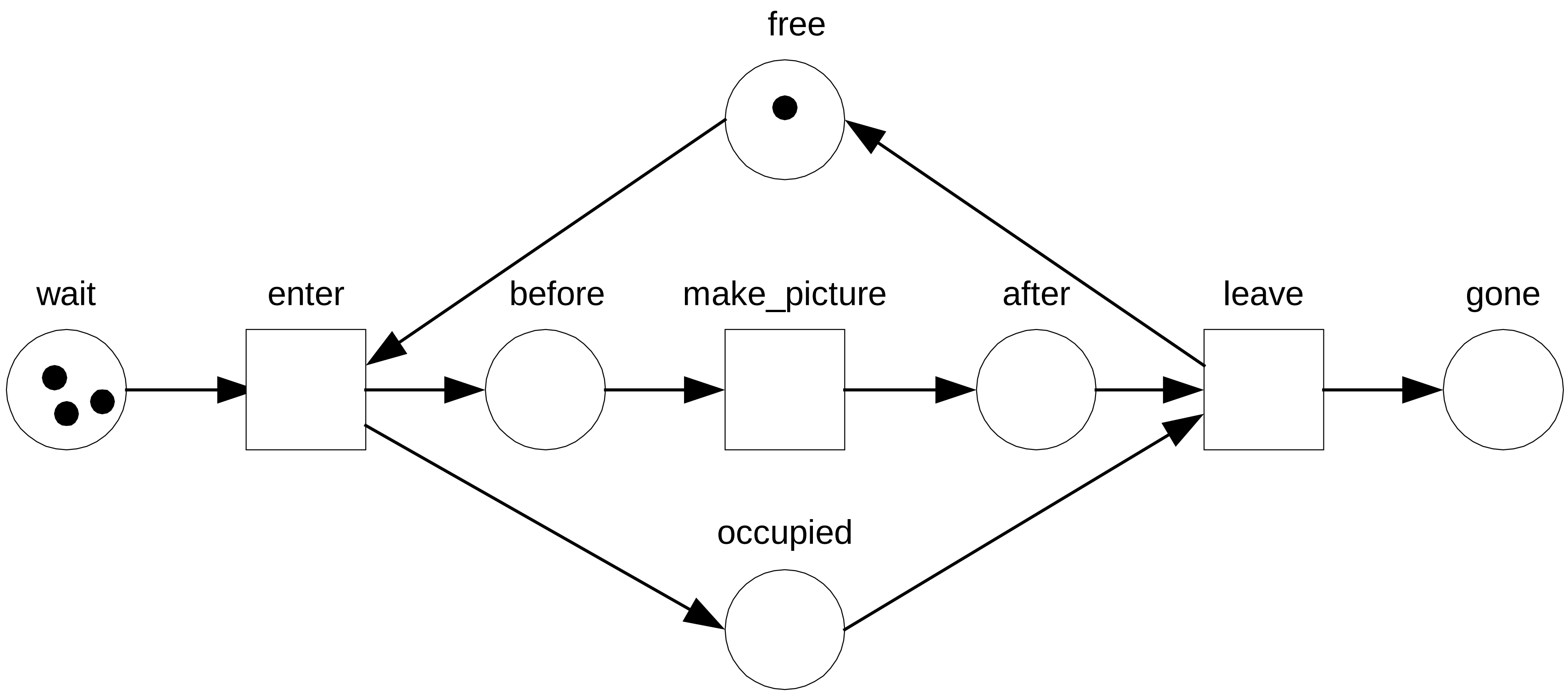
- Graph containing a node for each reachable state.
- Constructed by starting in the initial state, calculate all directly reachable states, etc.
- Expensive technique.
- Only feasible if finitely many states (otherwise use coverability graph).
- Difficult to generate diagnostic information.

Infinite reachability graph



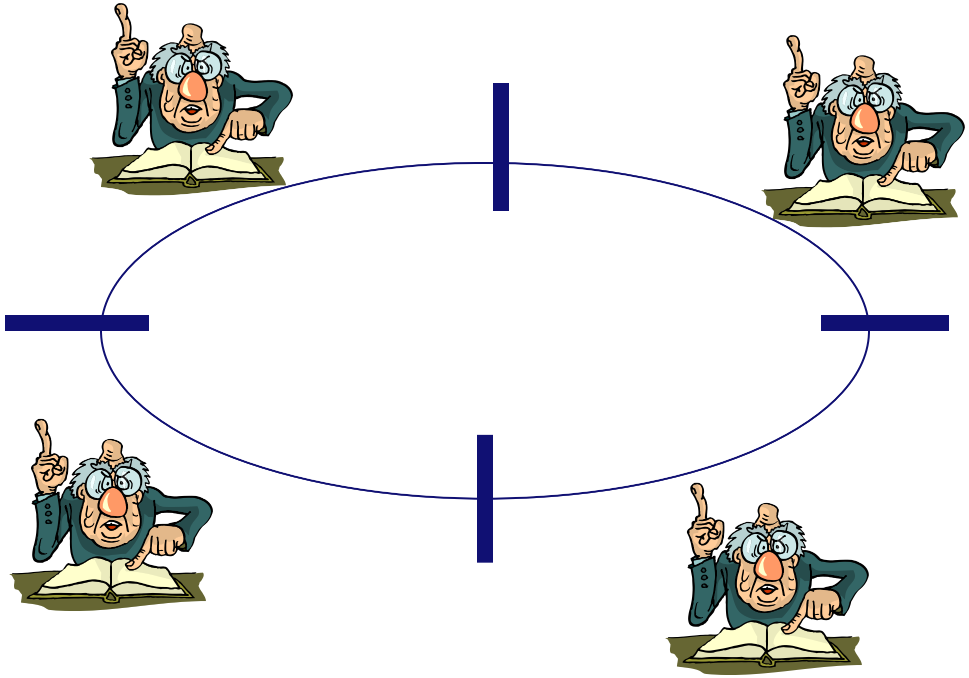
Therefore tools use a coverability graph which is always finite!

Exercise: Construct reachability graph



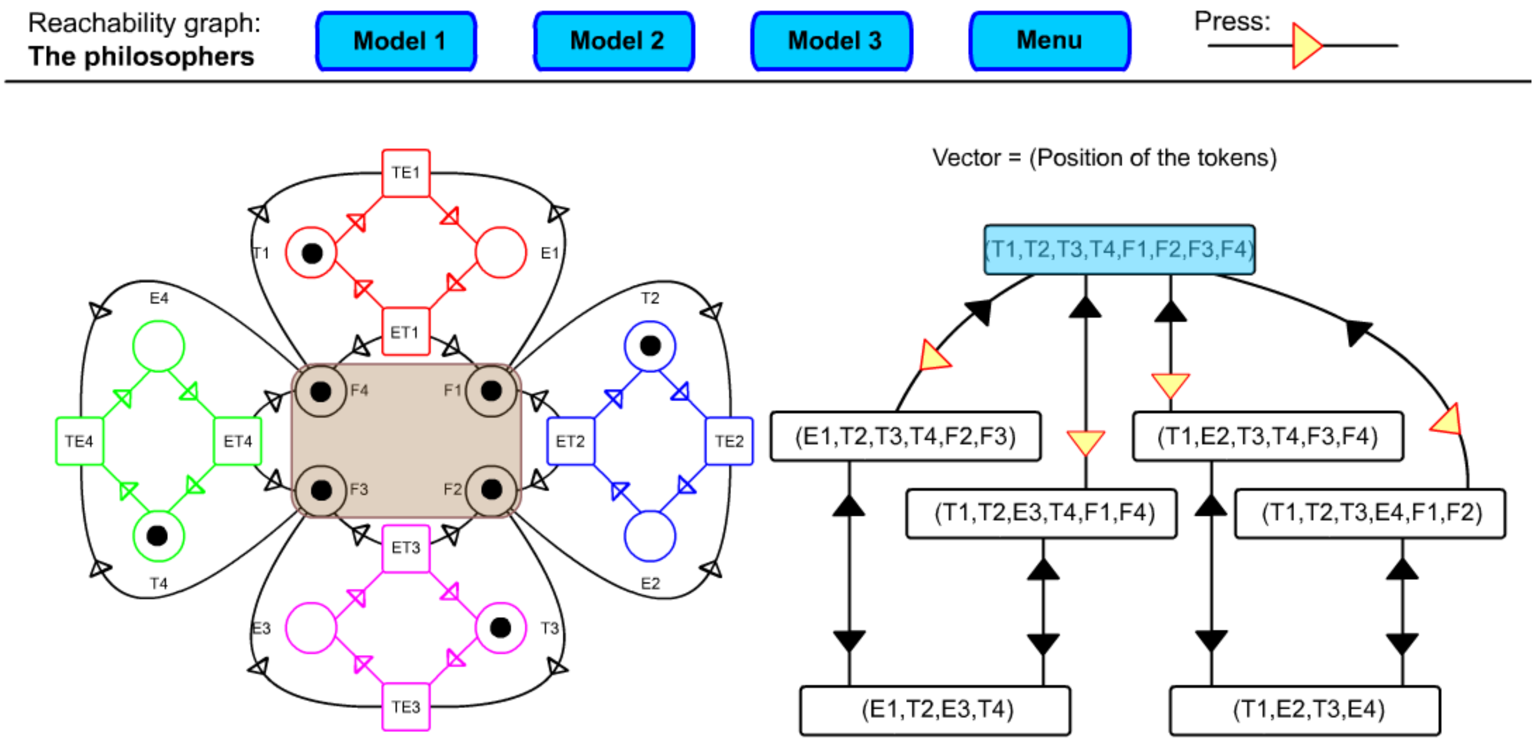
Exercise: Dining philosophers

- **4 philosophers sharing 4 chopsticks: A philosopher is either in state eating or thinking and needs two chopsticks to eat.**
- **Model as a Petri net and construct reachability graph.**



D:\www\wvdaalst\workflowcourse\examples\philosopher4.swf

See also: www.workflowcourse.com

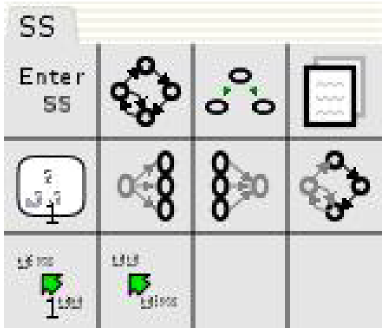


T: Thinking
TE: Finished thinking, Start eating
E: Eating
ET: Finished eating, Start thinking
F: Fork

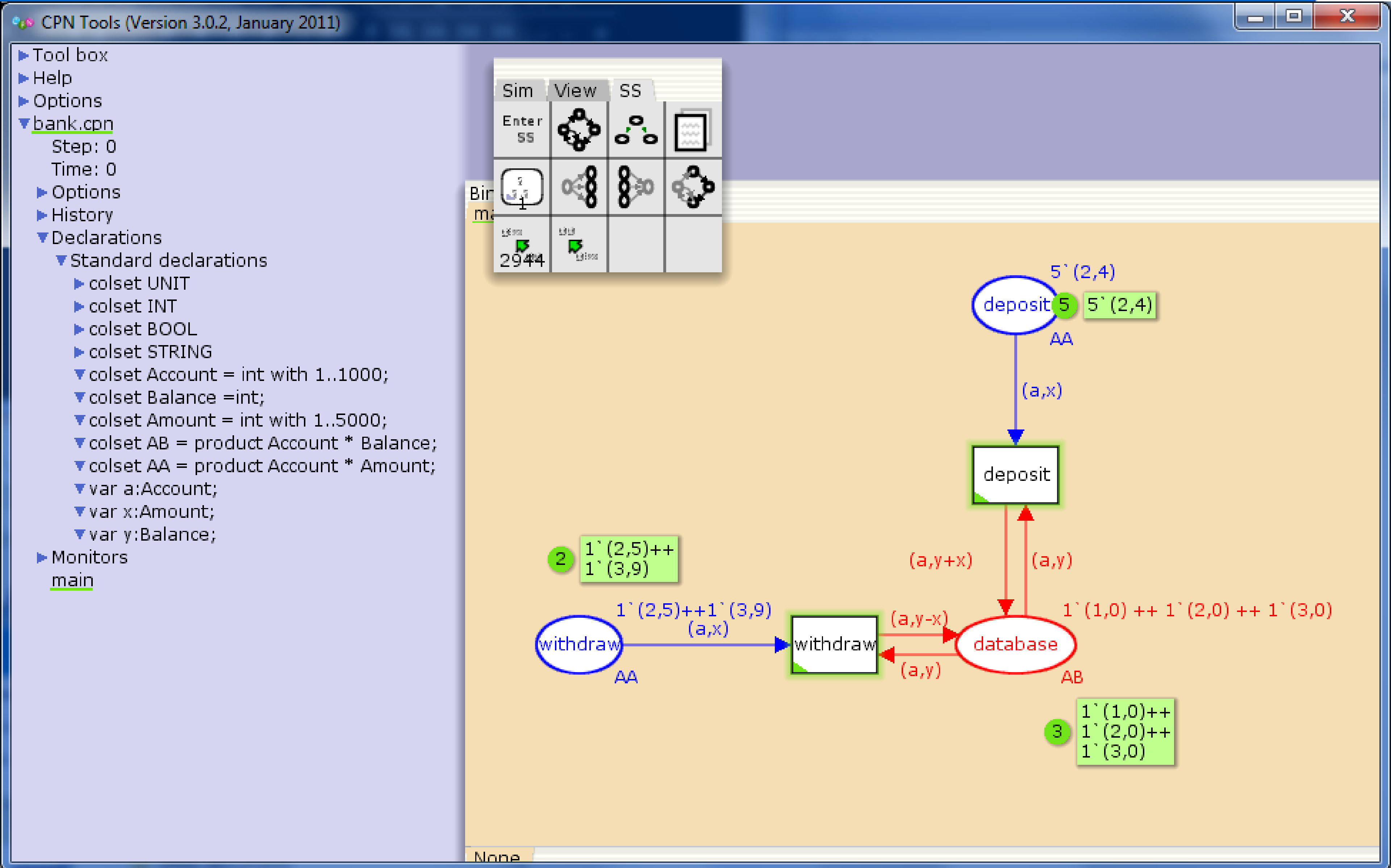
D:\www\wvdaalst\workflowcourse\examples\philosopher4_RG.swf

Analysis in CPN Tools

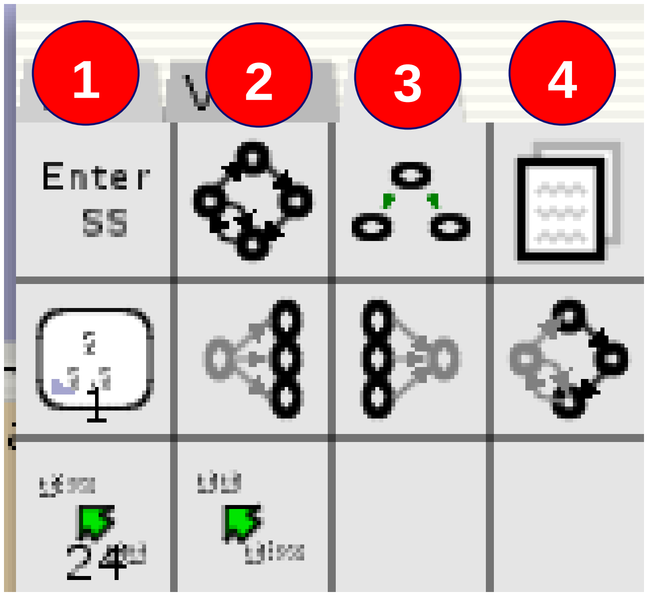
- Only state-space analysis, i.e., no invariants.
- Generate report in text file.
- State-space visualization from version 2.2.
- Steps:
  1. Enter the State Space Tool (to generate ML code)
  2. Calculating the state space
  3. Calculating the SCC graph
     (to calculate home states and fairness)
  4. Save/view state space report

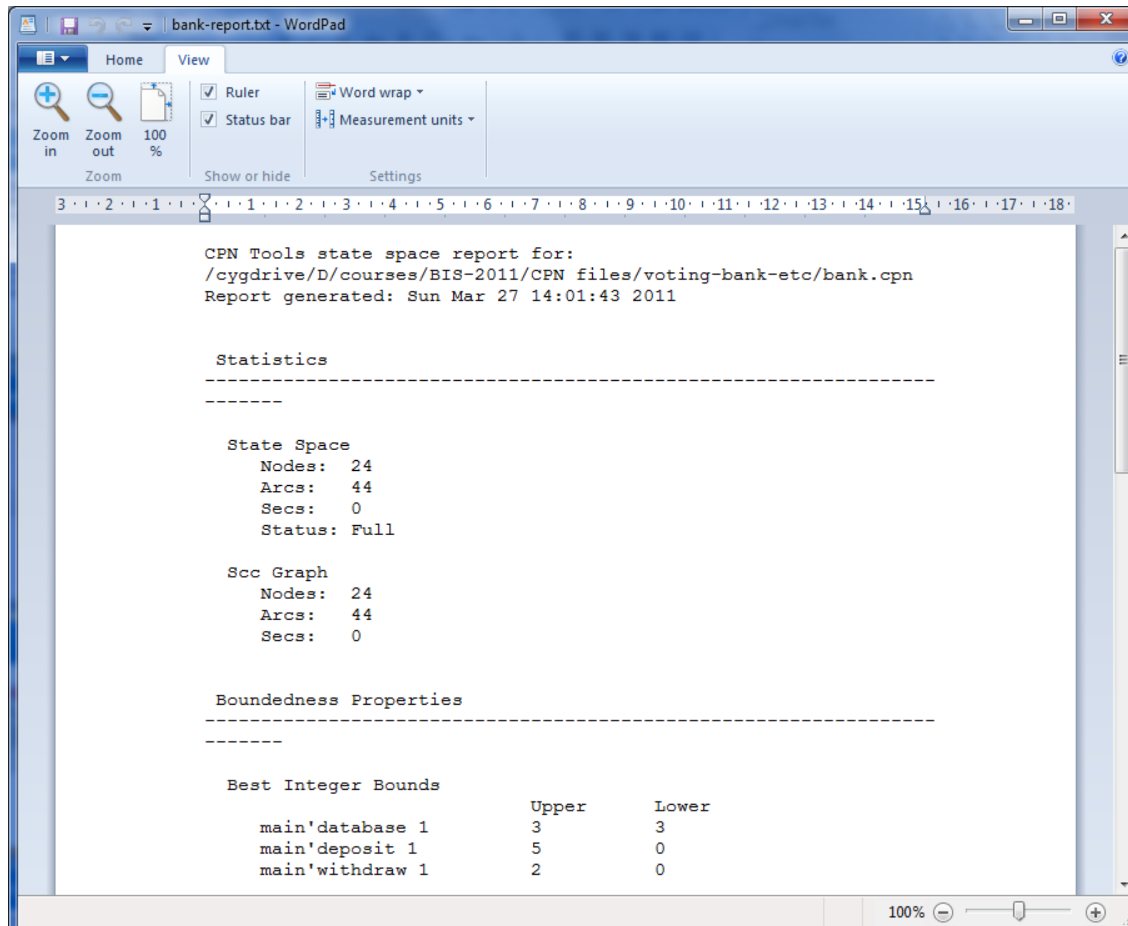


Example


CPN Tools (Version 3.0.2, January 2011)
Tool box
Help
Options
bank.cpn
Step: 0
Time: 0
Options
History
Declarations
Standard declarations
colset UNIT
colset INT
colset BOOL
colset STRING
colset Account = int with 1..1000;
colset Balance =int;
colset Amount = int with 1..5000;
colset AB = product Account * Balance;
colset AA = product Account * Amount;
var a:Account;
var x:Amount;
var y:Balance;
Monitors
main
Sim
View
SS
Enter SS
2944
5`(2,4)
deposit
5
5`(2,4)
AA
(a,x)
deposit
(a,y+x)
(a,y)
2
1`(2,5)++
1`(3,9)
1`(2,5)++1`(3,9)
(a,x)
withdraw
withdraw
(a,y-x)
database
1`(1,0) ++ 1`(2,0) ++ 1`(3,0)
(a,y)
AA
AB
3
1`(1,0)++
1`(2,0)++
1`(3,0)
None

Create report






```
CPN Tools state space report for:
/cygdrive/D/courses/BIS-2011/CPN files/voting-bank-etc/bank.cpn
Report generated: Sun Mar 27 14:01:43 2011


 Statistics
------------------------------------------------------------------
-------

  State Space
     Nodes:  24
     Arcs:   44
     Secs:   0
     Status: Full

  Scc Graph
     Nodes:  24
     Arcs:   44
     Secs:   0


 Boundedness Properties
------------------------------------------------------------------
-------

  Best Integer Bounds
                             Upper      Lower
     main'database 1         3          3
     main'deposit 1          5          0
     main'withdraw 1         2          0
```

Report (1)

CPN Tools state space report for:
/cygdrive/D/courses/BIS-2011/CPN files/voting-bank-etc/bank.cpn
Report generated: Sun Mar 27 14:01:43 2011

Statistics

State Space
Nodes: 24
Arcs: 44
Secs: 0
Status: Full

Scc Graph
Nodes: 24
Arcs: 44
Secs: 0

Report (2)

Boundedness Properties

Best Integer Bounds

	Upper	Lower
main'database 1	3	3
main'deposit 1	5	0
main'withdraw 1	2	0

Best Upper Multi-set Bounds

main'database 1 1`(1,0)++ 1`(2,(~5))++ 1`(2,(~1))++ 1`(2,0)++ 1`(2,3)++ 1`(2,4)++1`(2,7)++1`(2,8)++1`(2,11)++1`(2,12)++1`(2,15)++1`(2,16)+1`(2,20)++1 `(3,(~9))++1`(3,0)

main'deposit 1 5`(2,4)

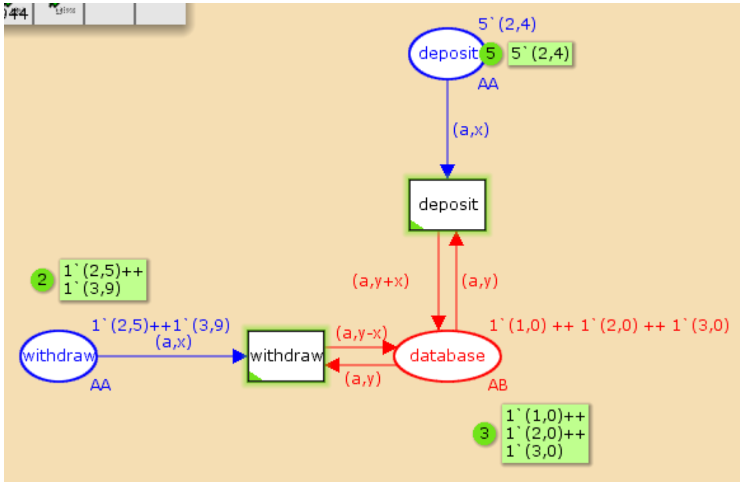
main'withdraw 1 1`(2,5)++1`(3,9)

Best Lower Multi-set Bounds

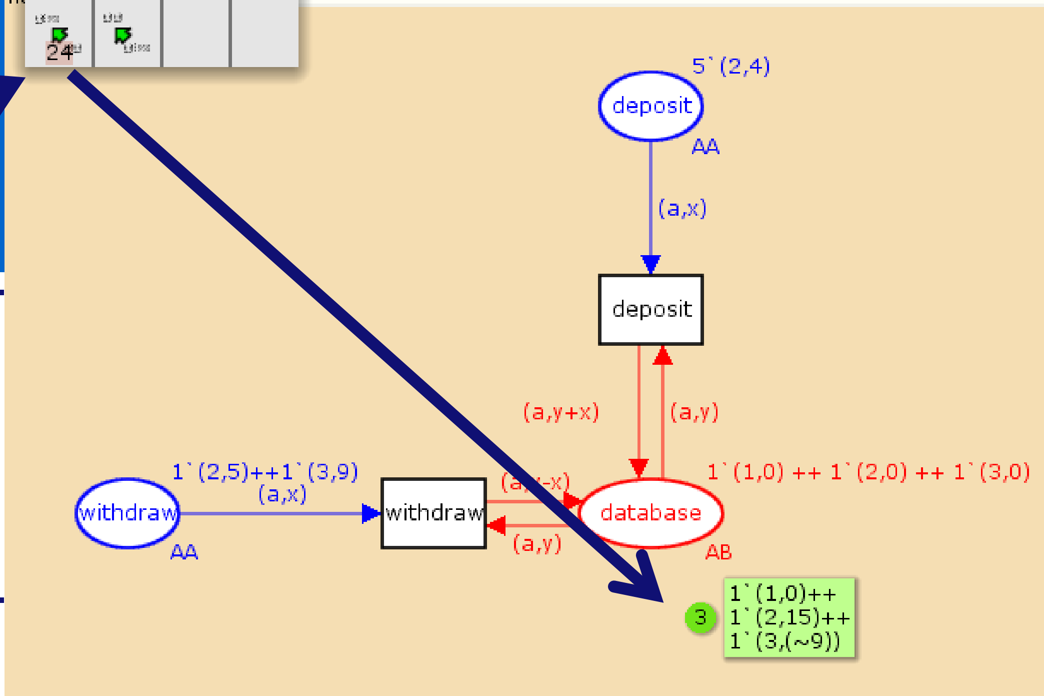
main'database 1 1`(1,0)

main'deposit 1 empty

main'withdraw 1 empty



Report (3)



Home Markings

[24]

Liveness Properties

Dead Markings

[24]

Dead Transition Instances

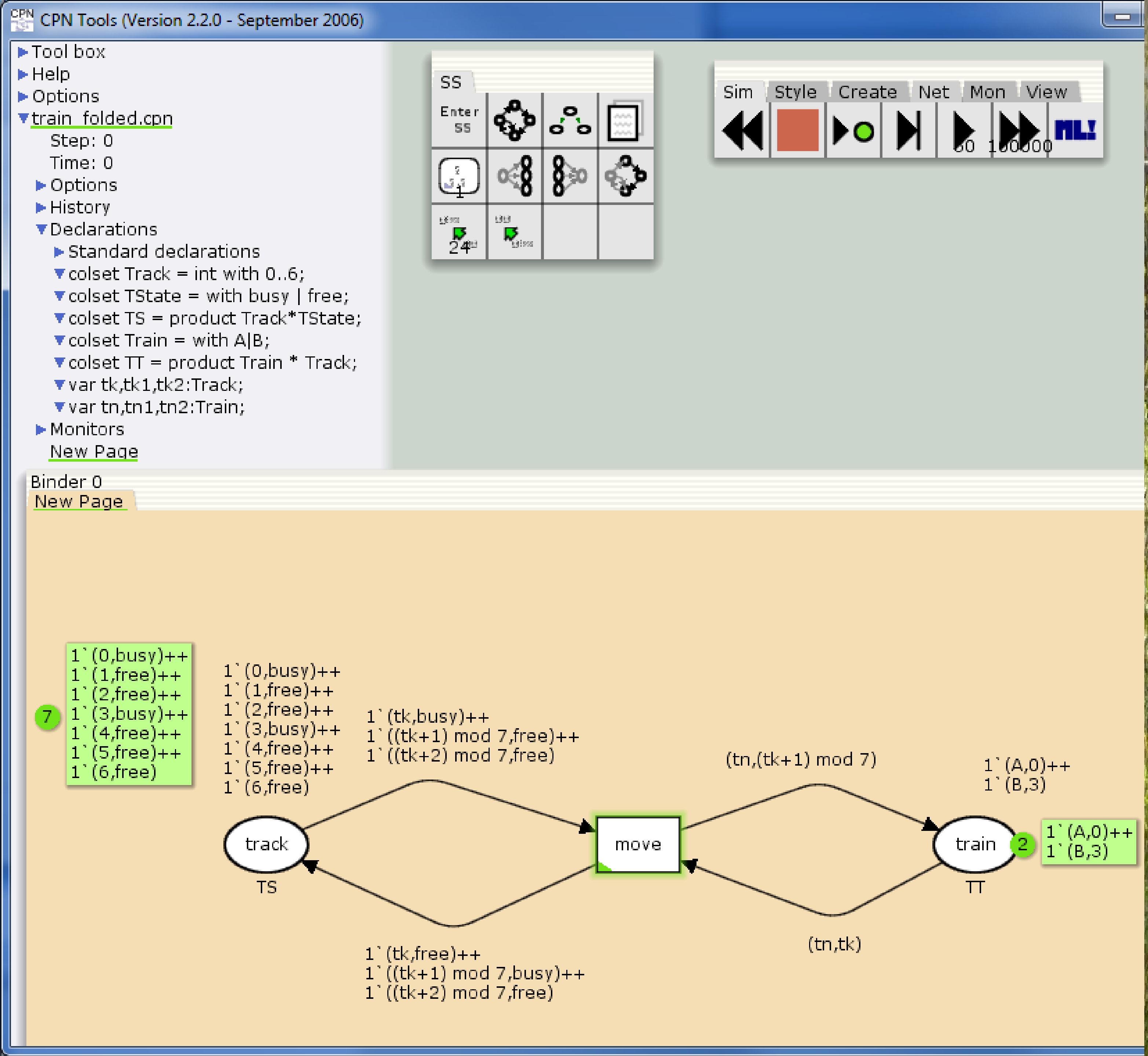
None

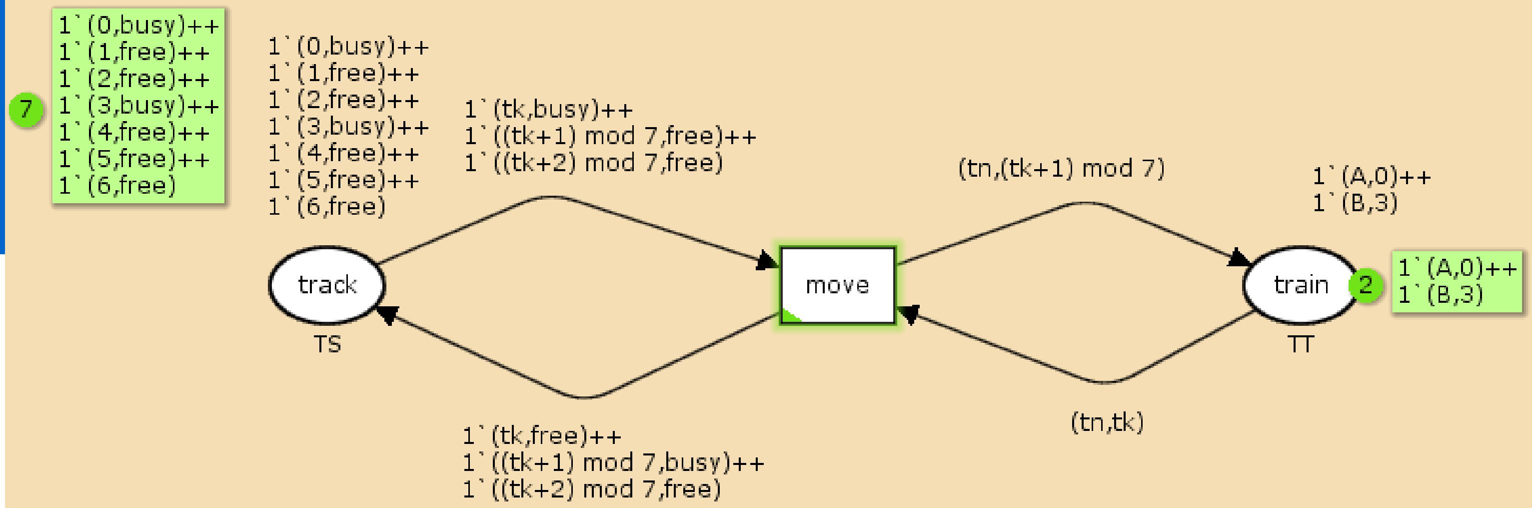
Live Transition Instances

None

Fairness Properties

No infinite occurrence sequences.


CPN Tools (Version 2.2.0 - September 2006)
Tool box
Help
Options
train_folded.cpn
Step: 0
Time: 0
Options
History
Declarations
Standard declarations
colset Track = int with 0..6;
colset TState = with busy | free;
colset TS = product Track*TState;
colset Train = with A|B;
colset TT = product Train * Track;
var tk,tk1,tk2:Track;
var tn,tn1,tn2:Train;
Monitors
New Page
SS
Enter SS
Sim
Style
Create
Net
Mon
View
Binder 0
New Page
1`(0,busy)++
1`(1,free)++
1`(2,free)++
1`(3,busy)++
1`(4,free)++
1`(5,free)++
1`(6,free)
7
1`(0,busy)++
1`(1,free)++
1`(2,free)++
1`(3,busy)++
1`(4,free)++
1`(5,free)++
1`(6,free)
1`(tk,busy)++
1`((tk+1) mod 7,free)++
1`((tk+2) mod 7,free)
(tn,(tk+1) mod 7)
1`(A,0)++
1`(B,3)
track
TS
move
train
TT
2
1`(A,0)++
1`(B,3)
1`(tk,free)++
1`((tk+1) mod 7,busy)++
1`((tk+2) mod 7,free)
(tn,tk)

State Space

Nodes: 28

Arcs: 42

Secs: 0

Status: Full

Scc Graph

Nodes: 1

Arcs: 0

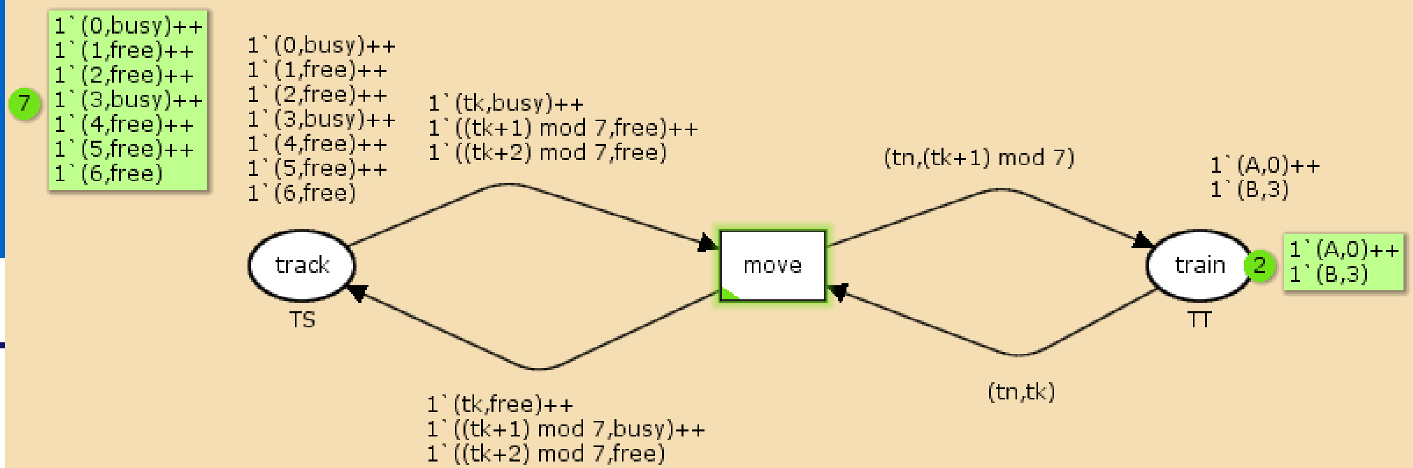
Secs: 0

Boundedness Properties

--

Best Integer Bounds

	Upper	Lower
New_Page'track 1	7	7
New_Page'train 1	2	2



Home Properties

Home Markings
All

Liveness Properties

Dead Markings
None

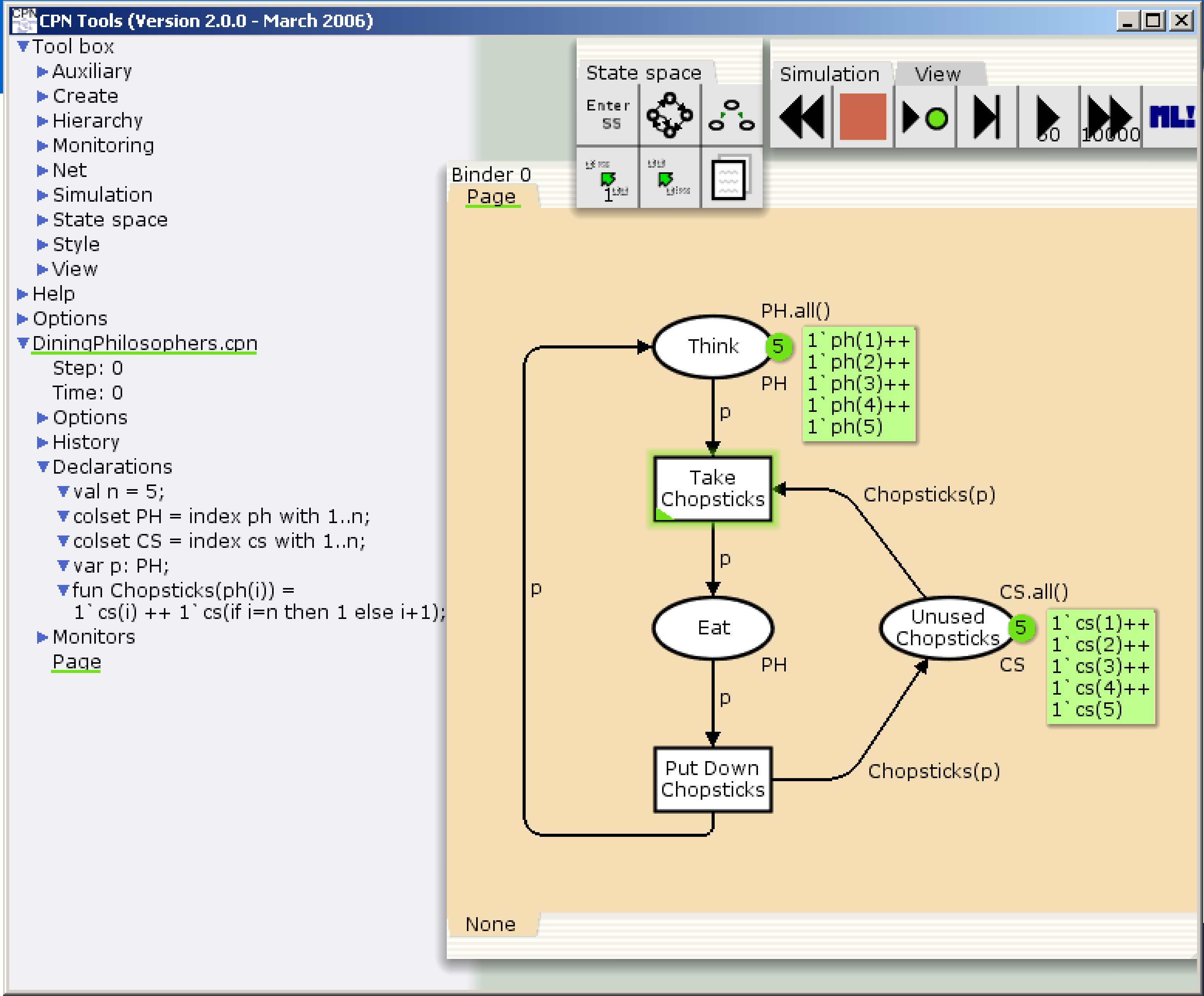
Dead Transition Instances
None

Live Transition Instances
All

Fairness Properties

New_Page'move 1 Impartial

Another example



Report (1)

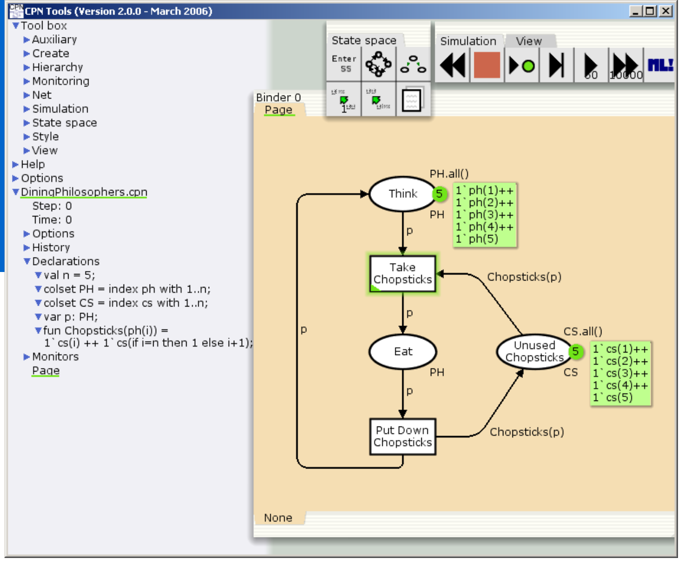
CPN Tools state space report for:
C:\Program Files\CPN Tools\Samples\DiningPhilosophers\DiningPhilosophers.cpn
Report generated: Thu Nov 02 10:42:53 2006

```
 Statistics
------------------------------------------------------------------------

  State Space
     Nodes:  11
     Arcs:   30
     Secs:   0
     Status: Full

  Scc Graph
     Nodes:  1
     Arcs:   0
     Secs:   0
```

Report (2)



Boundedness Properties

Best Integer Bounds

	Upper	Lower
Page'Eat 1	2	0
Page'Think 1	5	3
Page'Unused_Chopsticks 1	5	1

Best Upper Multi-set Bounds

Page'Eat 1 1\`ph(1)++ 1\`ph(2)++ 1\`ph(3)++ 1\`ph(4)++
1\`ph(5)
Page'Think 1 1\`ph(1)++ 1\`ph(2)++ 1\`ph(3)++ 1\`ph(4)++
1\`ph(5)
Page'Unused_Chopsticks 1 1\`cs(1)++ 1\`cs(2)++ 1\`cs(3)++ 1\`cs(4)++ 1\`cs(5)

Best Lower Multi-set Bounds

Page'Eat 1 empty
Page'Think 1 empty
Page'Unused_Chopsticks 1 empty

Report (3)

Home Properties

--

Home Markings

All

Liveness Properties

--

Dead Markings

None

Dead Transition Instances

None

Live Transition Instances

All

Fairness Properties

--

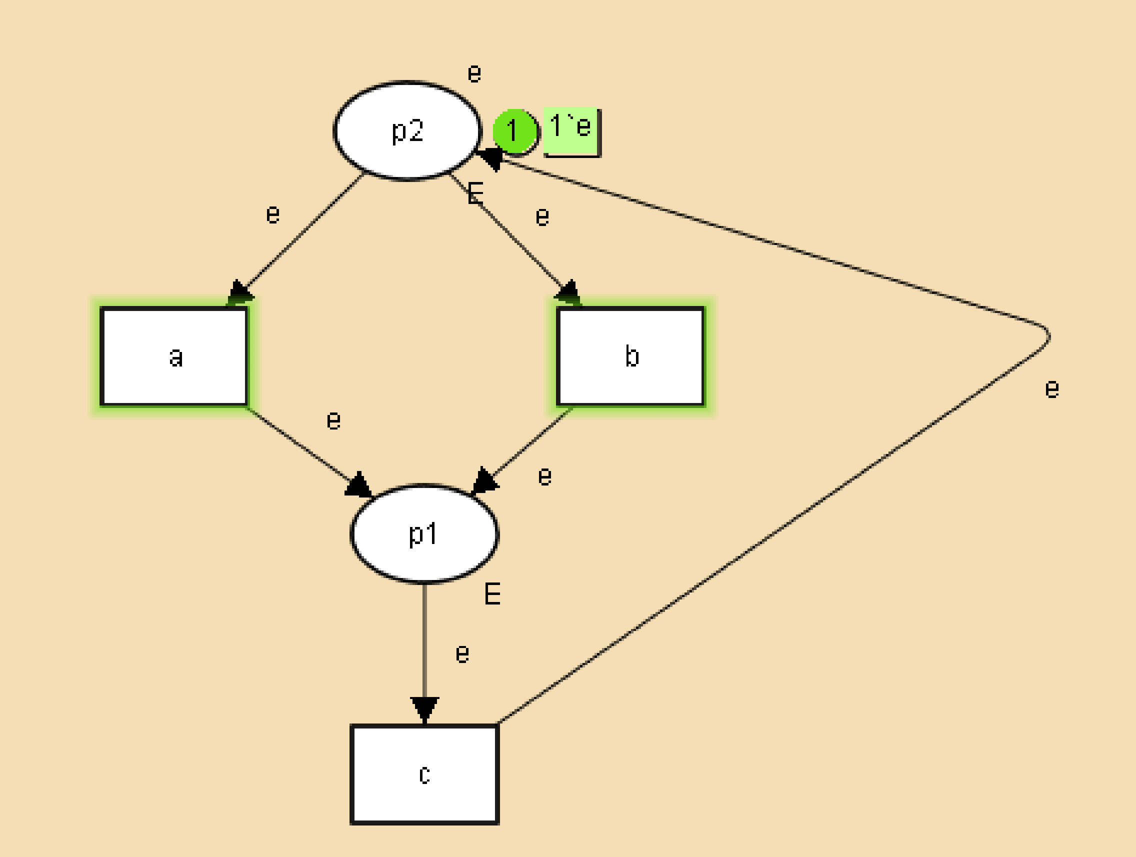
Page'Put_Down_Chopsticks 1 Impartial
Page'Take_Chopsticks 1 Impartial

strongest fairness property, i.e., there are infinite firing sequences and in each infinite firing sequence t occurs infinitely often

Fairness properties

- Are only relevant if there are Infinite Firing Sequences (IFS), otherwise CPN Tools reports: "no infinite occurrence sequences".
- Given a transition t it is often desirable that t appears infinitely often in an IFS.
- Properties reported by CPN Tools
  - t is impartial: t occurs infinitely often in every IFS.
  - t is fair: t occurs infinitely often in every IFS where t is enabled infinitely often.
  - t is just: t occurs infinitely often in every IFS where t is continuously enabled from some point onward
  - No fairness: not just, i.e., there is an IFS where t is continuously enabled from some point onward and does not fire anymore

Example

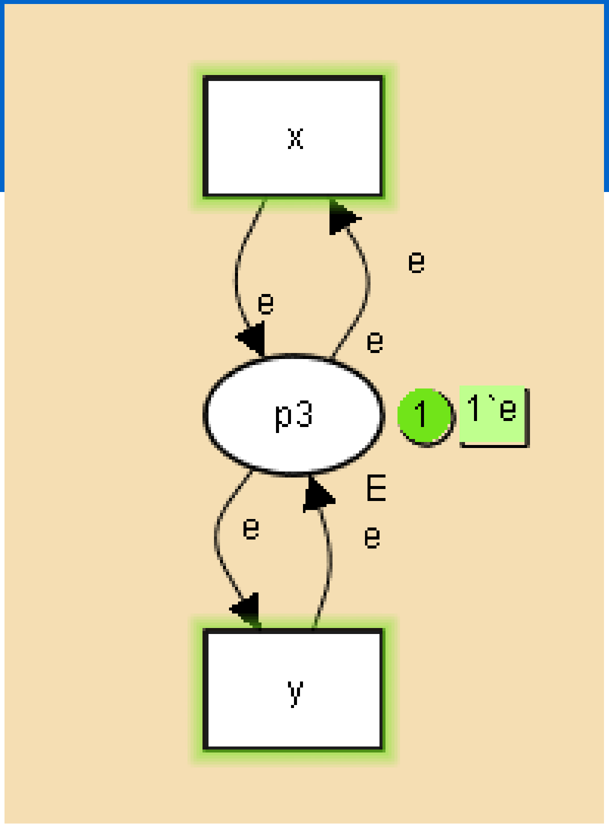


Fairness Properties

---

main1'a 1	**Just**
main1'b 1	**Just**
main1'c 1	**Impartial**

Example

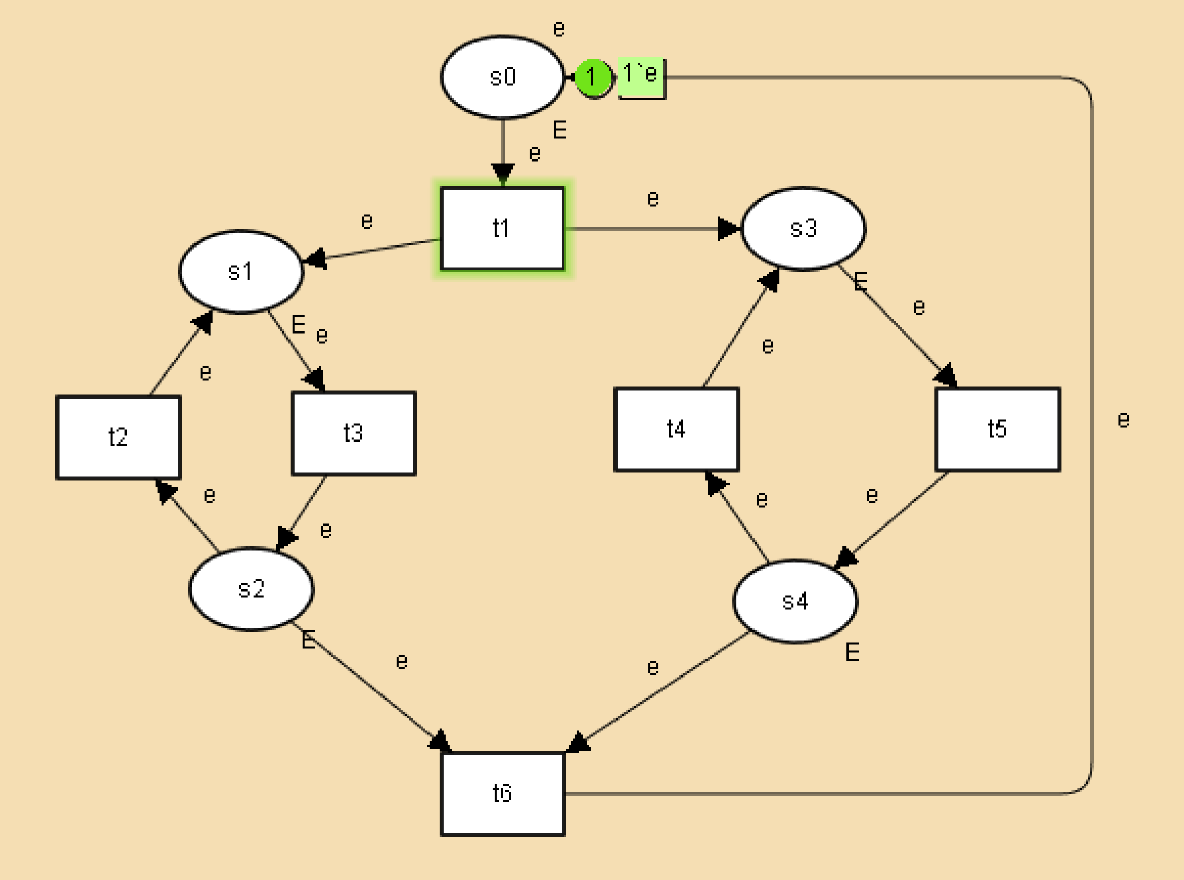


Fairness Properties

---

main2'x 1 No Fairness
main2'y 1 No Fairness

Example

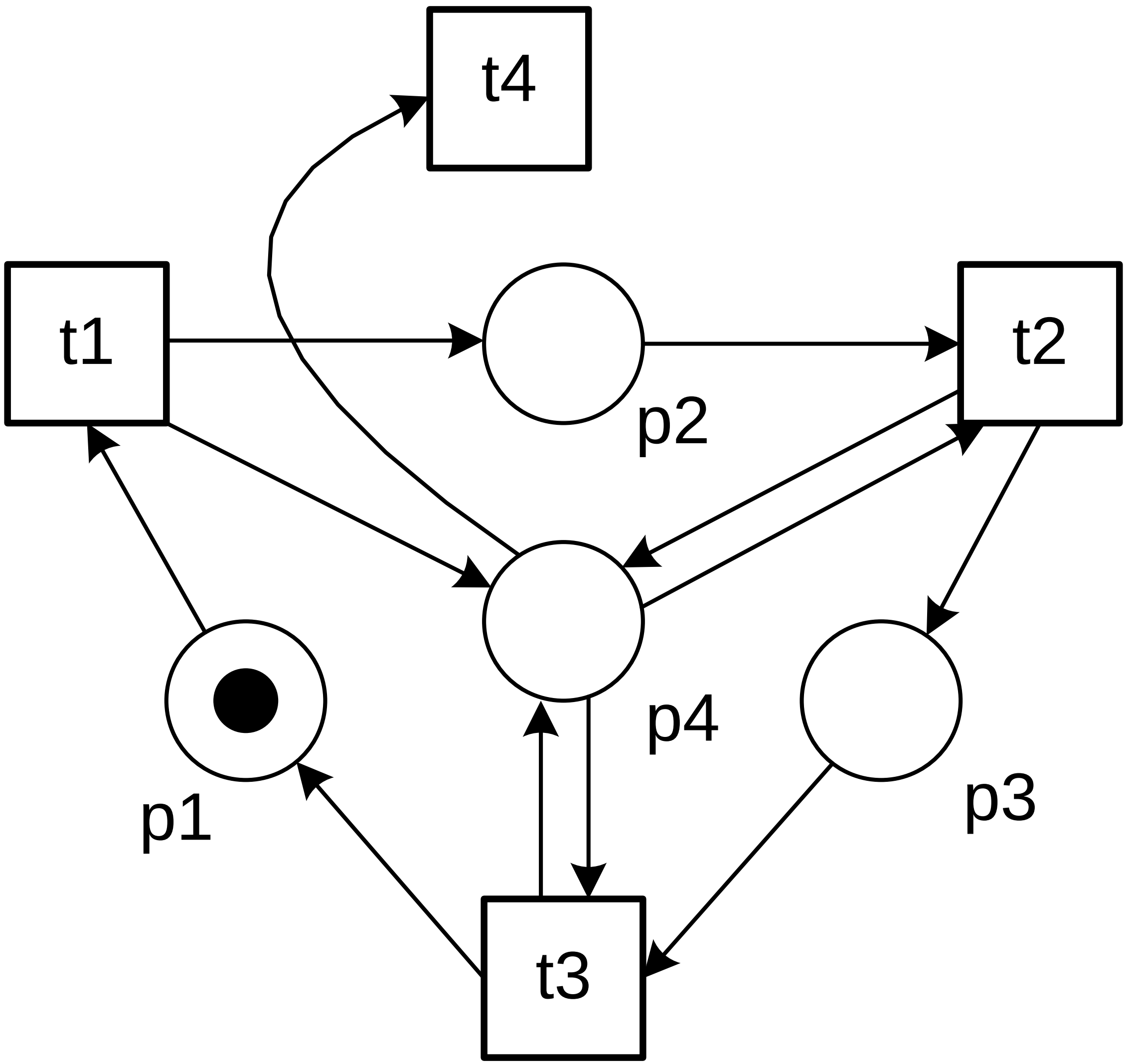


Fairness Properties

---

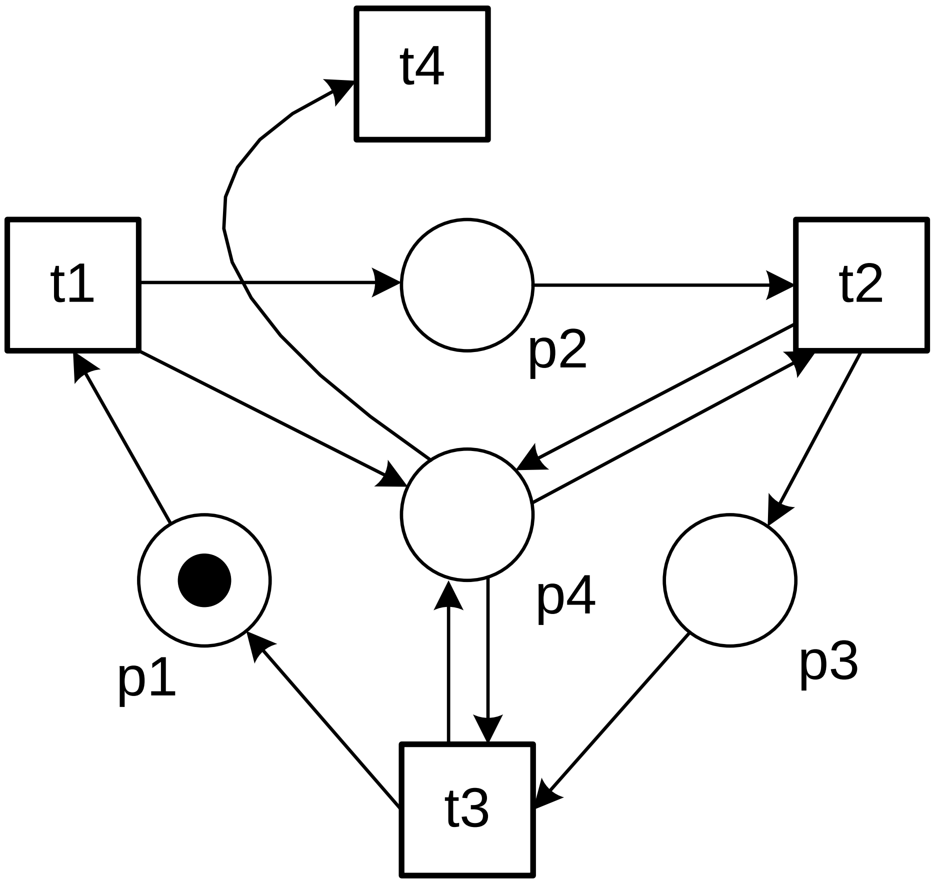
main3't1 1	**Fair**
main3't2 1	**No Fairness**
main3't3 1	**No Fairness**
main3't4 1	**No Fairness**
main3't5 1	**No Fairness**
main3't6 1	**Just**

Exercise



Indicate for each transition whether it is *impartial*, *fair*, or *just* (or satisfies *no fairness property*)

- **t1, t2, and t3 are all impartial because it is not possible to construct an infinite firing sequence where not all of these transitions appear infinitely often. If one stops executing one of these transitions, the system will block after a while.**
- **t4 has no fairness as it is possible to construct an infinite firing sequence where t4 remains enabled but never fires.**



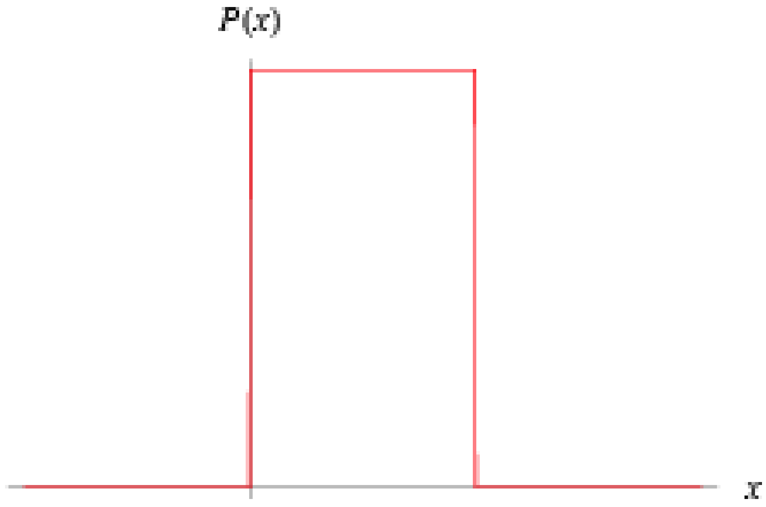
Simulation

- Most widely used analysis technique.
- From a technical point of view just a "walk" in the reachability graph.
- By making many "walks" (in case of transient behavior) or a very "long walk" (in case of steady-state) behavior, it is possible to make reliable statements about properties/ performance indicators.
- Used for validation and performance analysis.
- Cannot be used to prove correctness!

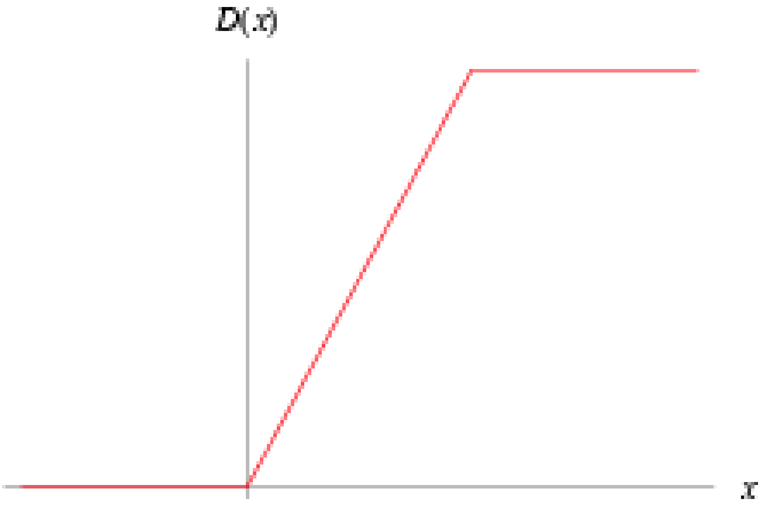
Stochastic process

- Simulation of a deterministic system is not very interesting.
- Simulation of an untimed system is not interesting.
- In a timed and non-deterministic system, durations and probabilities are described by some probability distribution.
- In other words, we simulate a stochastic process!
- CPN allows for the use of distributions using some internal random generator.

Uniform distribution

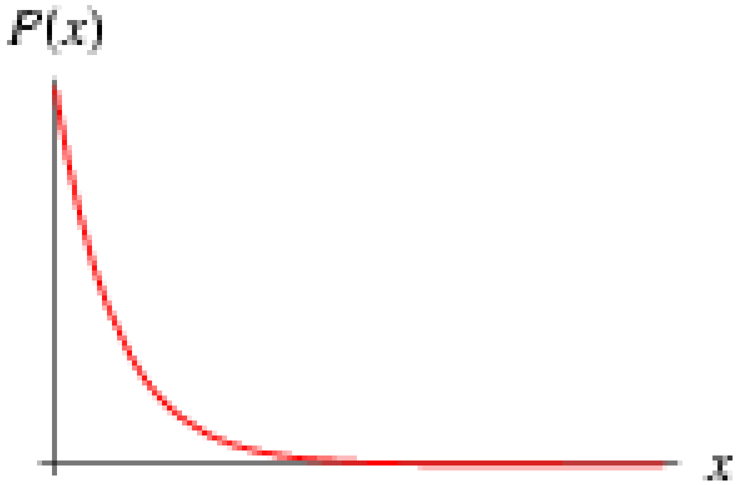


probability density function (PDF)

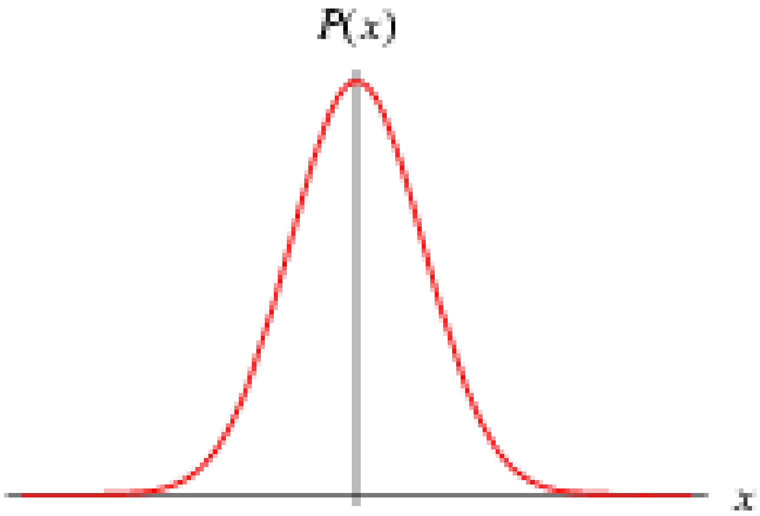


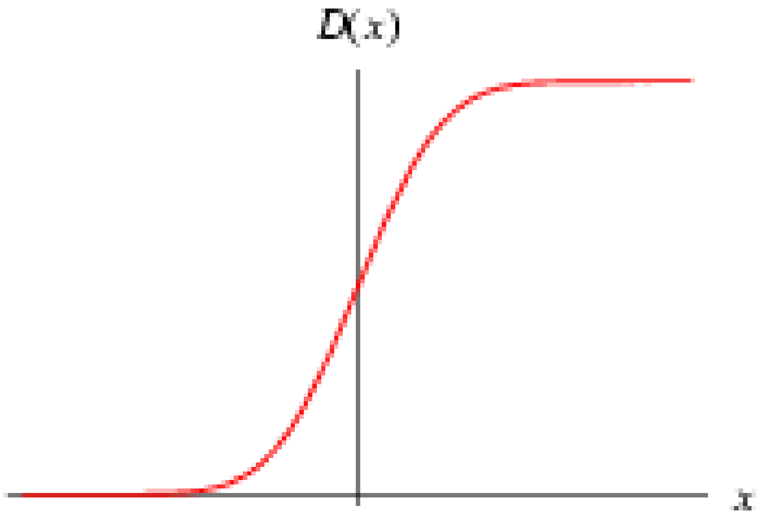
cumulative distribution function (CDF)

Negative exponential distribution



Normal distribution





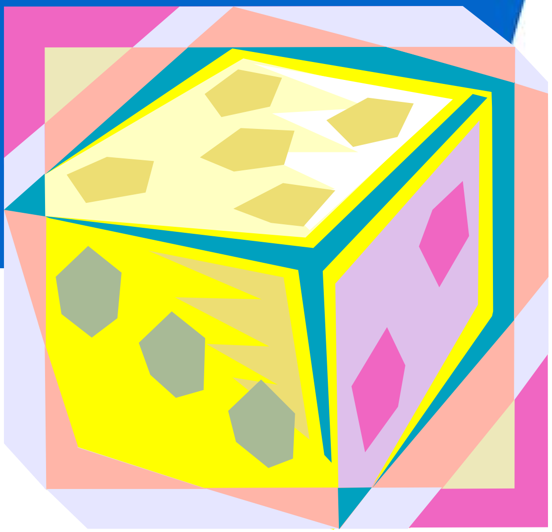
Distributions in CPN Tools

There is standard library with stochastic functions:

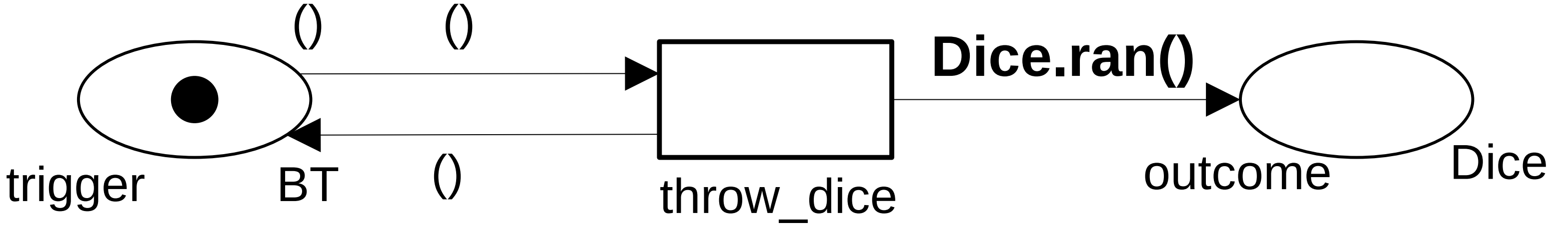
- **uniform(a:real, b:real) : real**
- **exponential(r:real) : real**
- **normal(n:real, v:real) : real**
- **erlang (n:int, r:real) : real**
- **Etc.**

A nice additional function is also C.ran() which returns a randomly selected element of finite color set C, e.g.,
color C = int with 1..5;
fun select1to5() = C.ran()
returns a number between 1 and 5

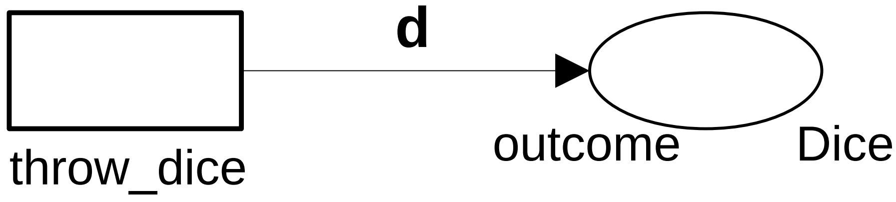
Example



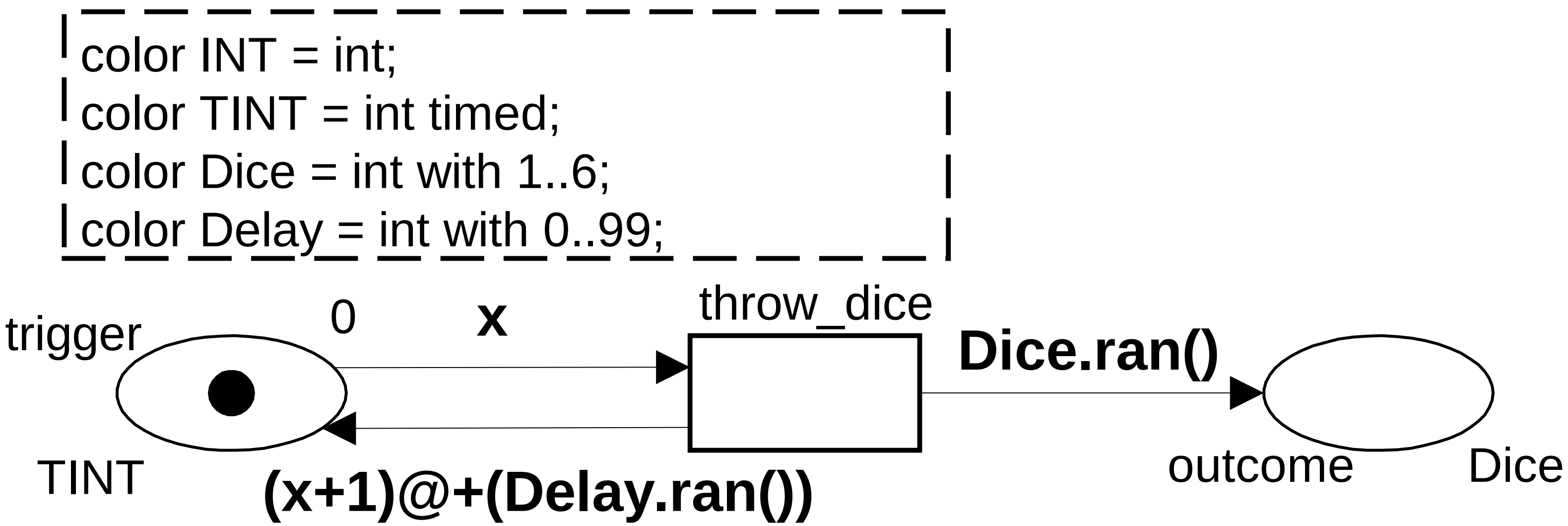
```
color BT = unit;
color Dice = int with 1..6;
var d : Dice;
```



or even simpler …

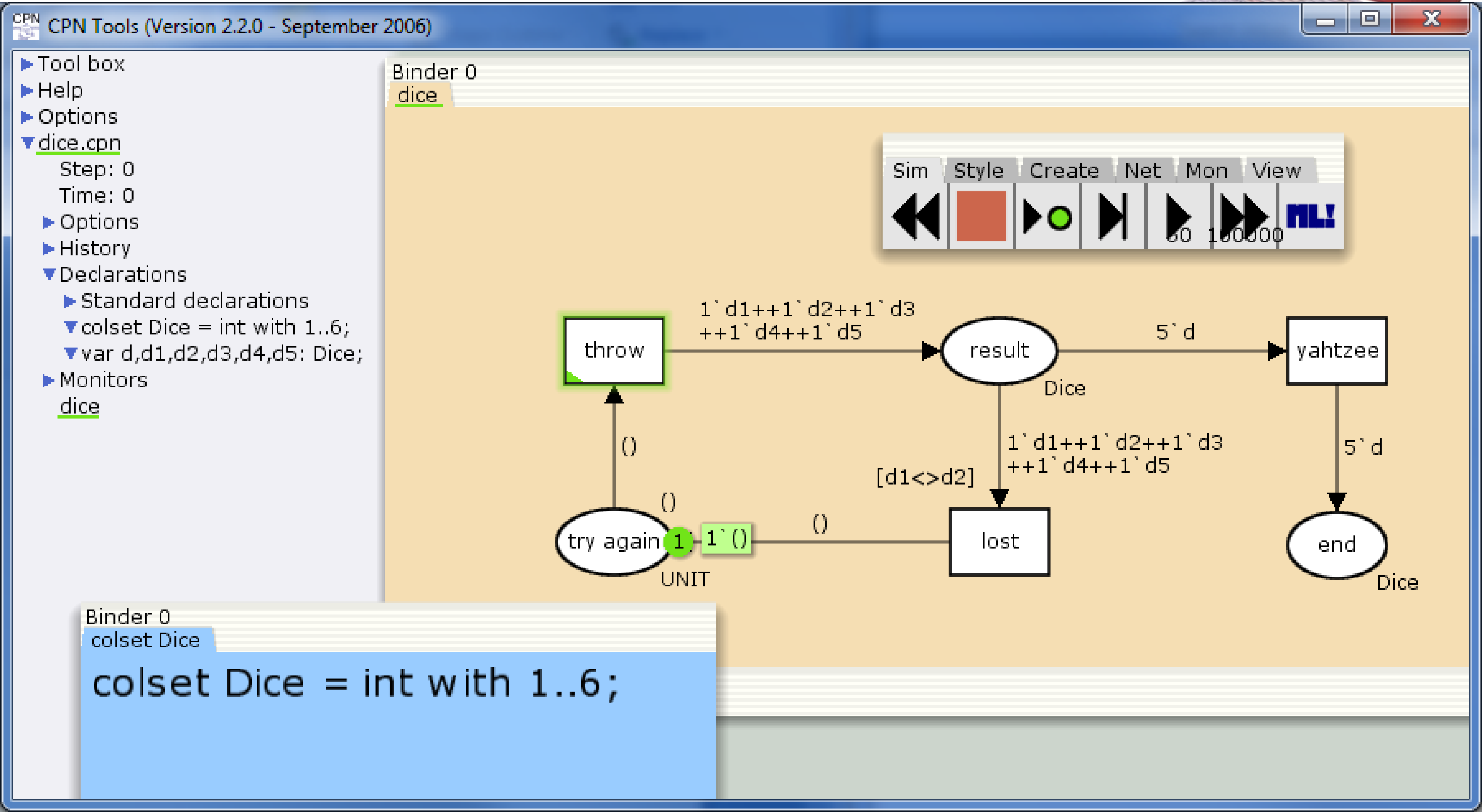


Example(2)



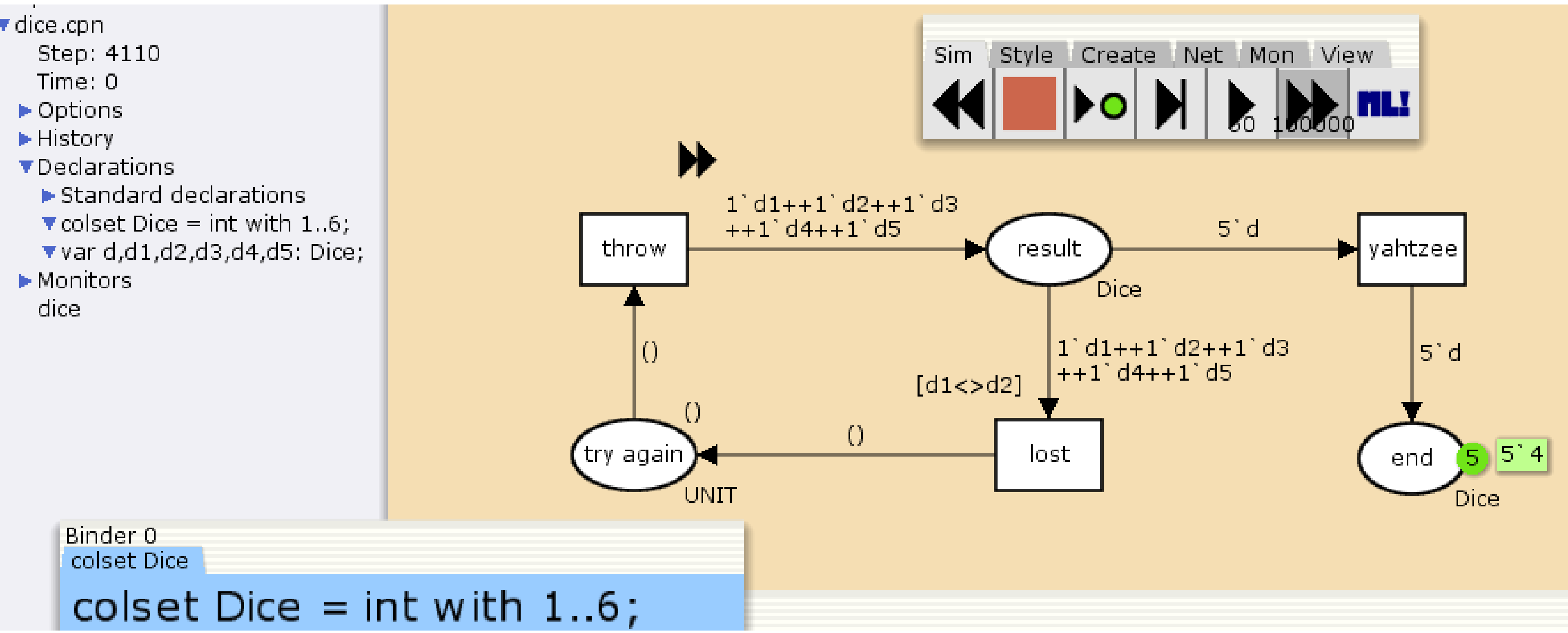
color INT = int;
color TINT = int timed;
color Dice = int with 1..6;
color Delay = int with 0..99;
trigger
0
x
throw_dice
Dice.ran()
TINT
(x+1)@+(Delay.ran())
outcome
Dice

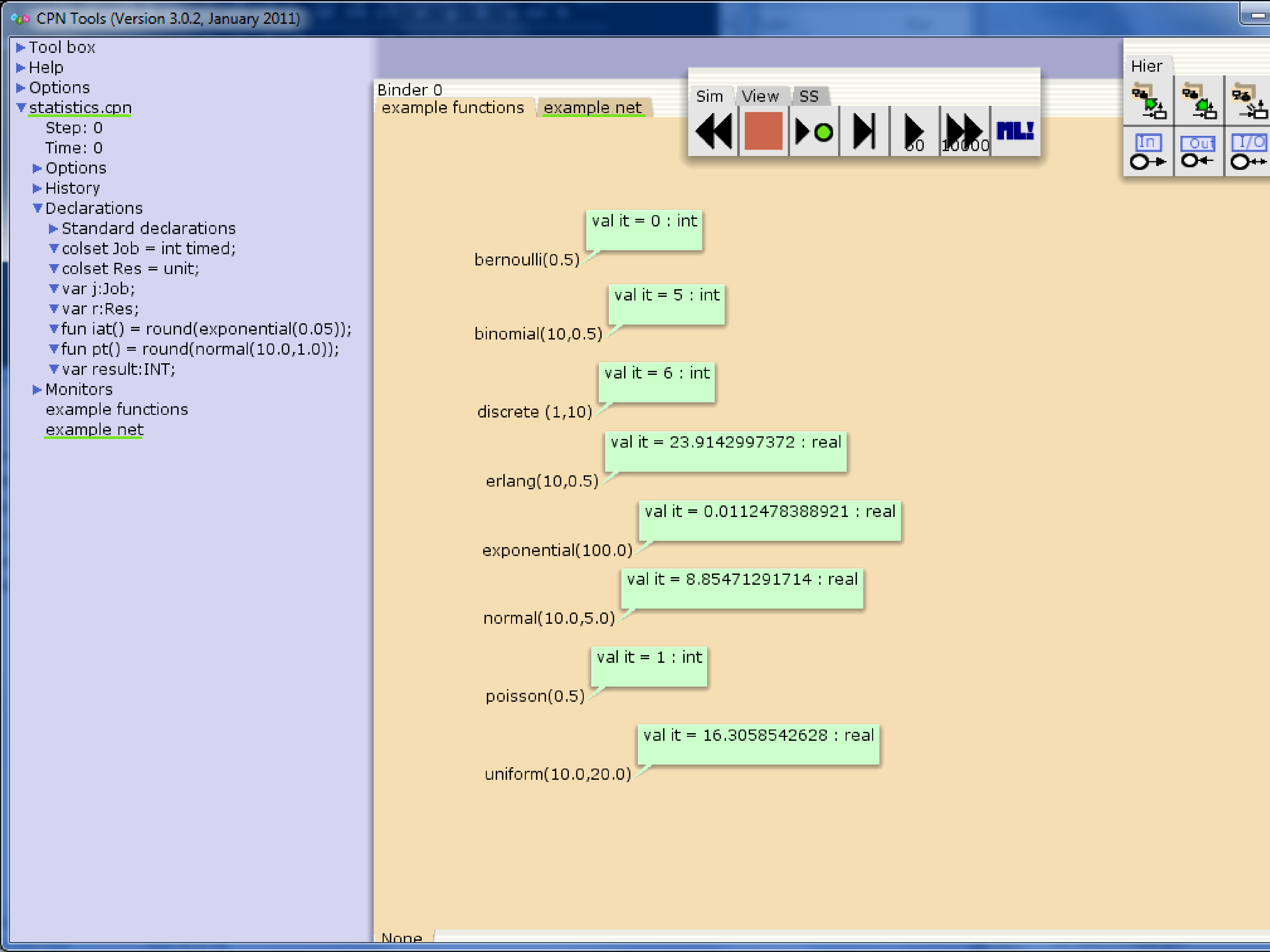
Yahtzee



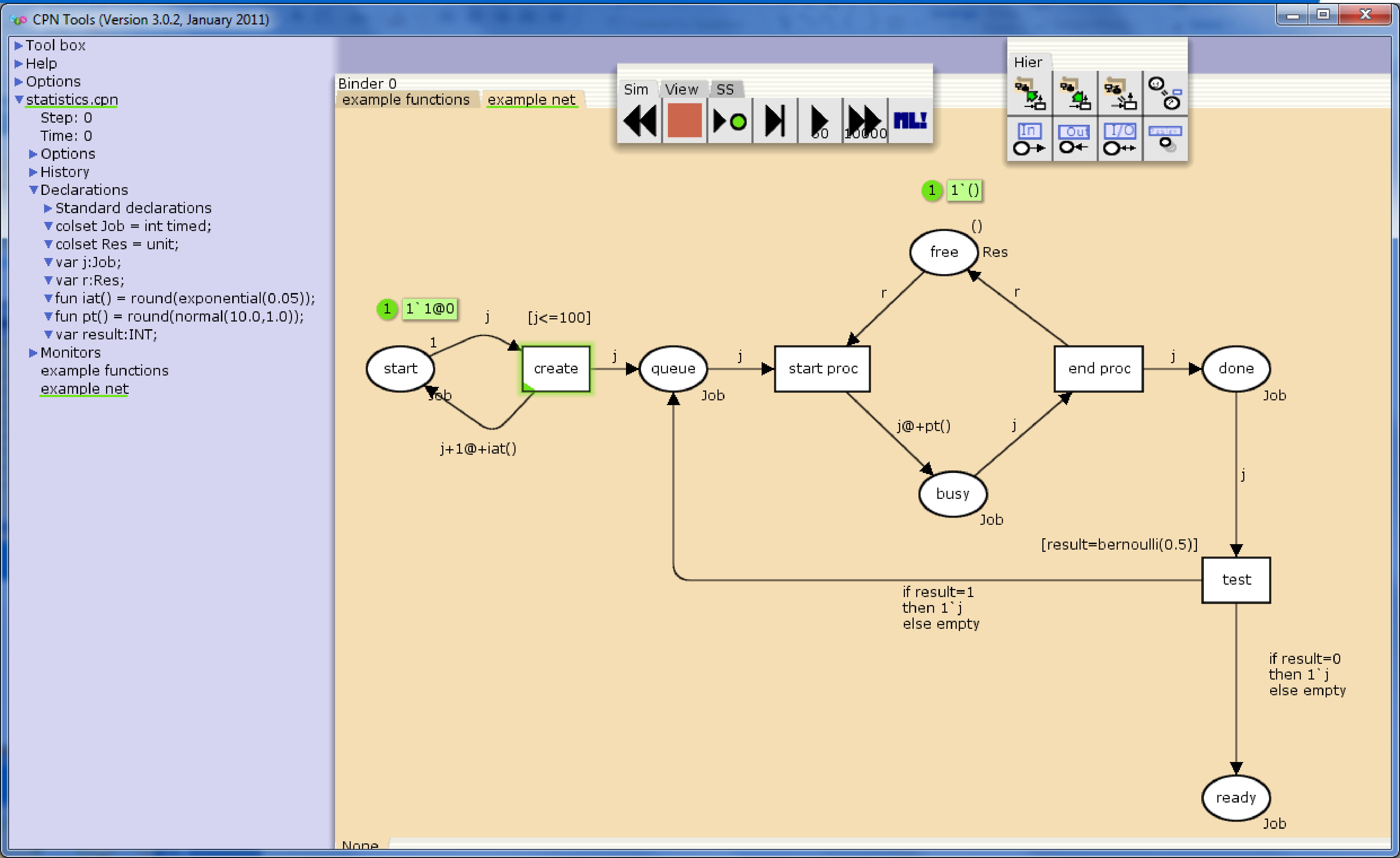
CPN Tools (Version 2.2.0 - September 2006)
Tool box
Help
Options
dice.cpn
Step: 0
Time: 0
Options
History
Declarations
Standard declarations
colset Dice = int with 1..6;
var d,d1,d2,d3,d4,d5: Dice;
Monitors
dice
Binder 0
dice
Sim
Style
Create
Net
Mon
View
50
100000
ML!
throw
1`d1++1`d2++1`d3
++1`d4++1`d5
result
Dice
5`d
yahtzee
()
1`d1++1`d2++1`d3
++1`d4++1`d5
[d1<>d2]
5`d
()
try again
1
1`()
()
lost
end
UNIT
Dice
Binder 0
colset Dice
colset Dice = int with 1..6;

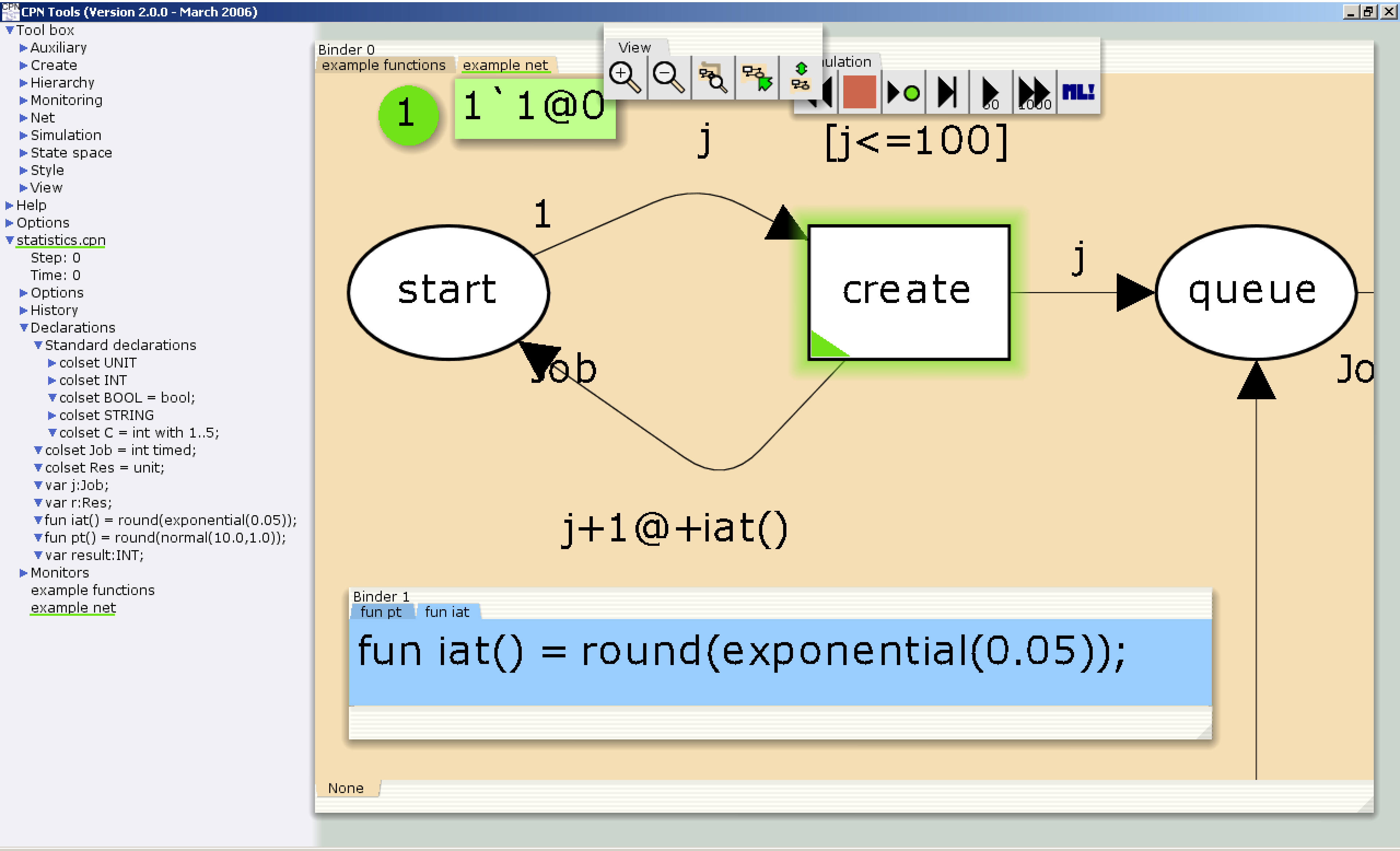
After 2055 times throwing the dices … five 4's



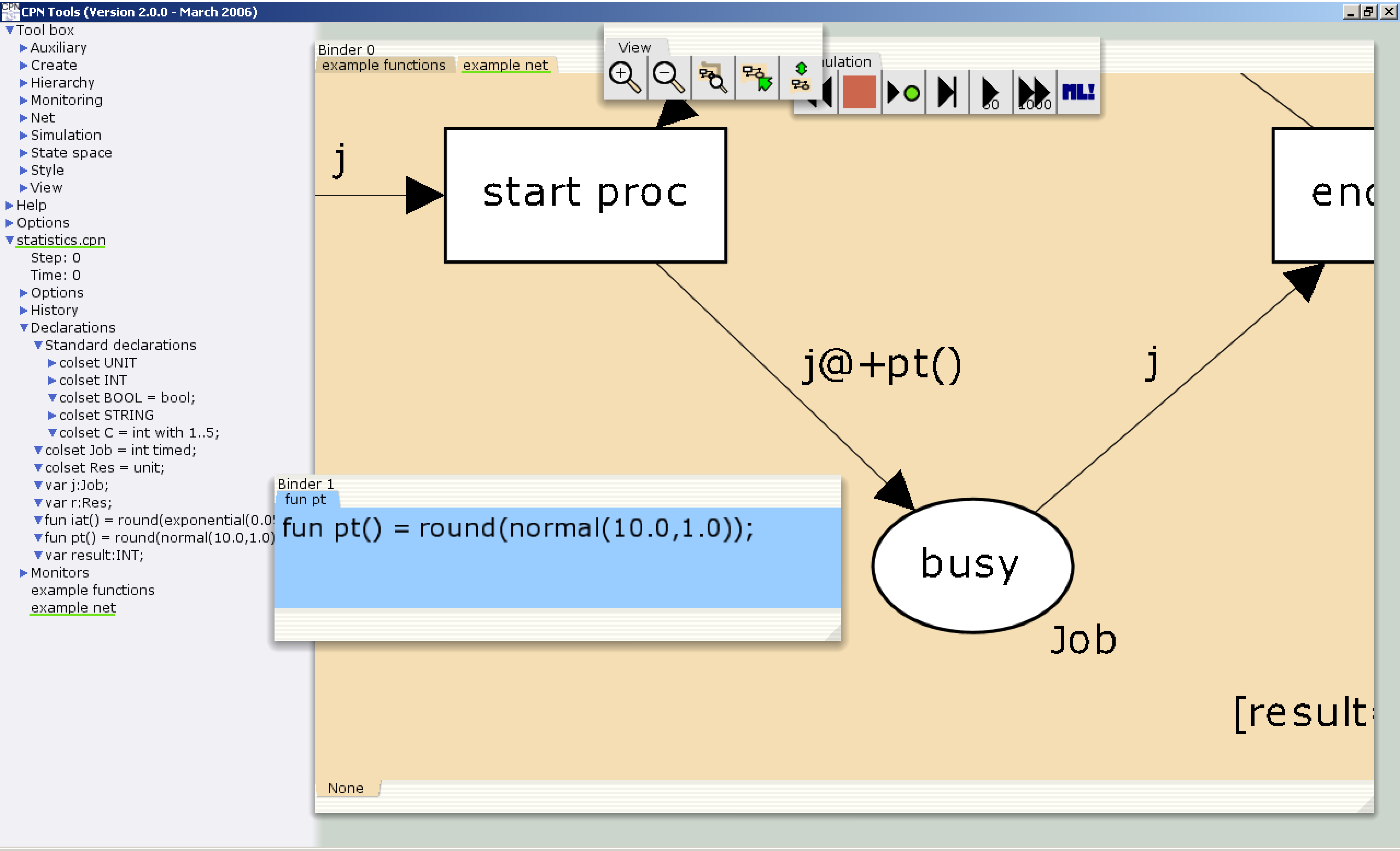

CPN Tools (Version 3.0.2, January 2011)
Tool box
Help
Options
statistics.cpn
Step: 0
Time: 0
Options
History
Declarations
Standard declarations
colset Job = int timed;
colset Res = unit;
var j:Job;
var r:Res;
fun iat() = round(exponential(0.05));
fun pt() = round(normal(10.0,1.0));
var result:INT;
Monitors
example functions
example net
Binder 0
example functions
example net
Sim
View
SS
50
10000
ML!
Hier
In
Out
I/O
val it = 0 : int
bernoulli(0.5)
val it = 5 : int
binomial(10,0.5)
val it = 6 : int
discrete (1,10)
val it = 23.9142997372 : real
erlang(10,0.5)
val it = 0.0112478388921 : real
exponential(100.0)
val it = 8.85471291714 : real
normal(10.0,5.0)
val it = 1 : int
poisson(0.5)
val it = 16.3058542628 : real
uniform(10.0,20.0)
None

Example

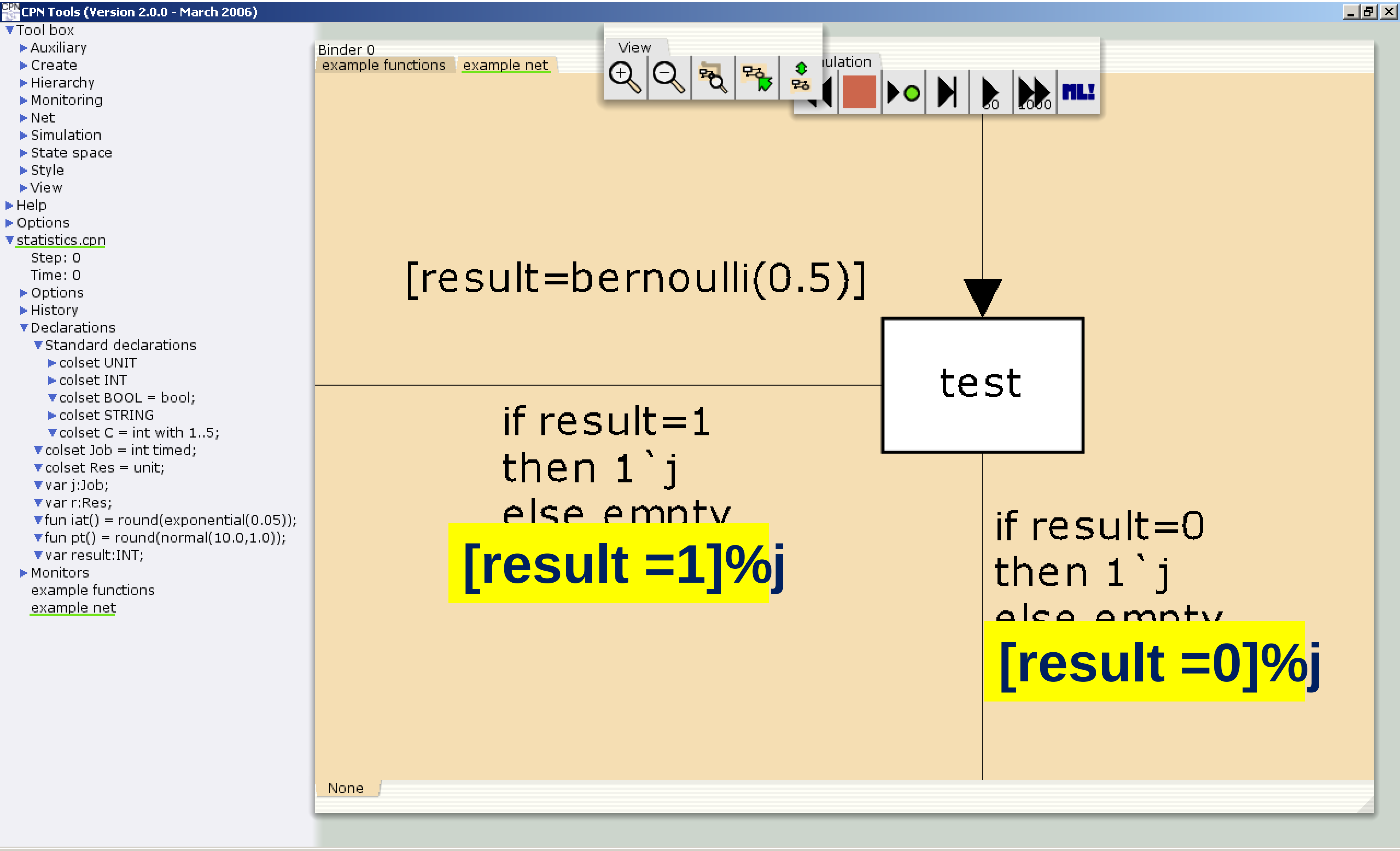

CPN Tools (Version 3.0.2, January 2011)
Tool box
Help
Options
statistics.cpn
Step: 0
Time: 0
Options
History
Declarations
Standard declarations
colset Job = int timed;
colset Res = unit;
var j:Job;
var r:Res;
fun iat() = round(exponential(0.05));
fun pt() = round(normal(10.0,1.0));
var result:INT;
Monitors
example functions
example net
Binder 0
example functions
example net
Sim
View
SS
Hier
1
1`()
()
free
Res
r
r
1
1`1@0
j
[j<=100]
1
start
create
j
queue
j
start proc
end proc
j
done
Job
Job
Job
j+1@+iat()
j@+pt()
j
busy
Job
j
[result=bernoulli(0.5)]
test
if result=1
then 1`j
else empty
if result=0
then 1`j
else empty
ready
Job
None



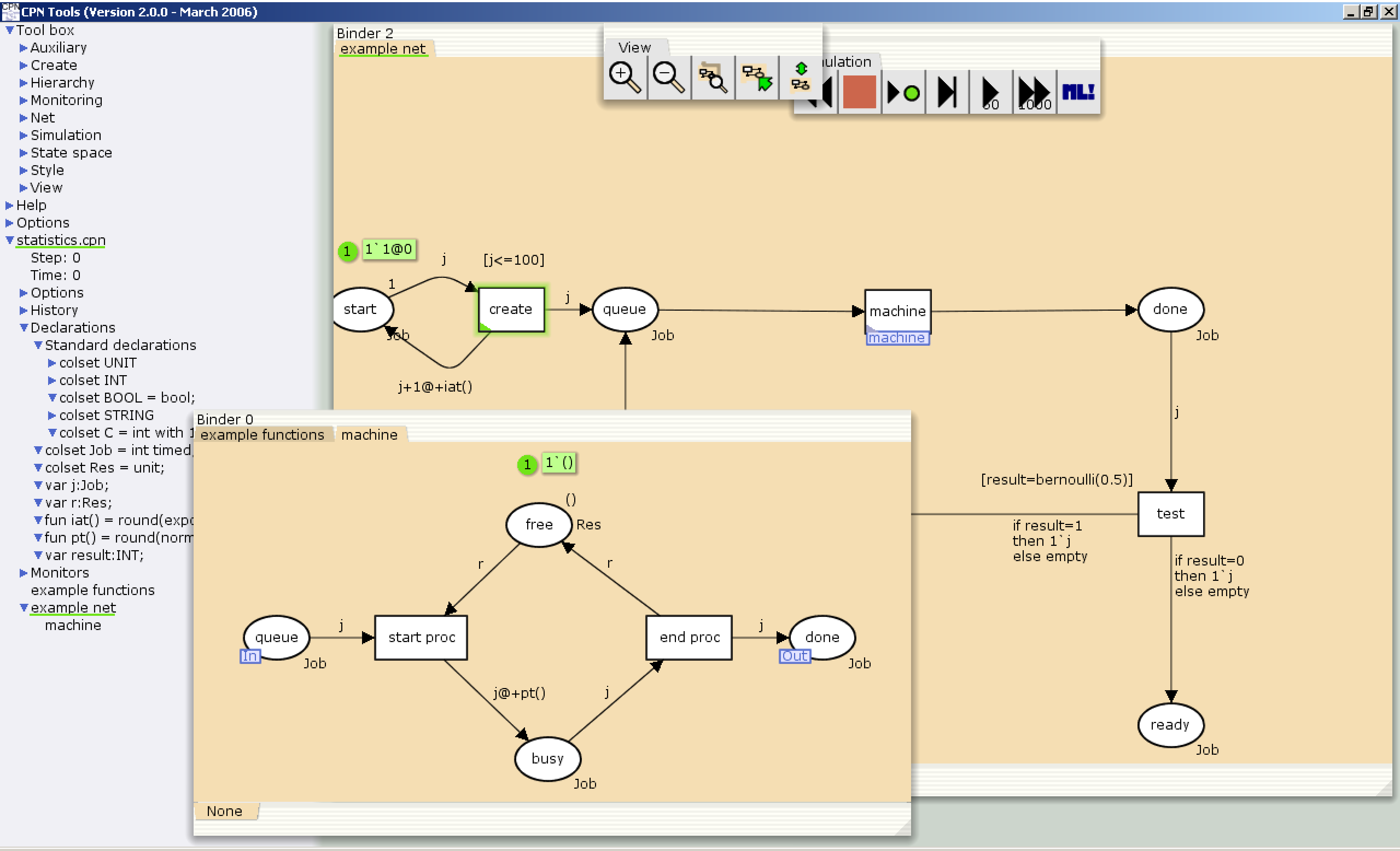
CPN Tools (Version 2.0.0 - March 2006)
Tool box
Auxiliary
Create
Hierarchy
Monitoring
Net
Simulation
State space
Style
View
Help
Options
statistics.cpn
Step: 0
Time: 0
Options
History
Declarations
Standard declarations
colset UNIT
colset INT
colset BOOL = bool;
colset STRING
colset C = int with 1..5;
colset Job = int timed;
colset Res = unit;
var j:Job;
var r:Res;
fun iat() = round(exponential(0.05));
fun pt() = round(normal(10.0,1.0));
var result:INT;
Monitors
example functions
example net
Binder 0
example functions
example net
View
1
1`1@0
j
[j<=100]
1
start
create
j
queue
Job
j+1@+iat()
Binder 1
fun pt
fun iat
fun iat() = round(exponential(0.05));
None


CPN Tools (Version 2.0.0 - March 2006)
Tool box
Auxiliary
Create
Hierarchy
Monitoring
Net
Simulation
State space
Style
View
Help
Options
statistics.cpn
Step: 0
Time: 0
Options
History
Declarations
Standard declarations
colset UNIT
colset INT
colset BOOL = bool;
colset STRING
colset C = int with 1..5;
colset Job = int timed;
colset Res = unit;
var j:Job;
var r:Res;
fun iat() = round(exponential(0.0
fun pt() = round(normal(10.0,1.0)
var result:INT;
Monitors
example functions
example net
Binder 0
example functions
example net
View
j
start proc
j@+pt()
j
busy
Job
[result
None
Binder 1
fun pt
fun pt() = round(normal(10.0,1.0));

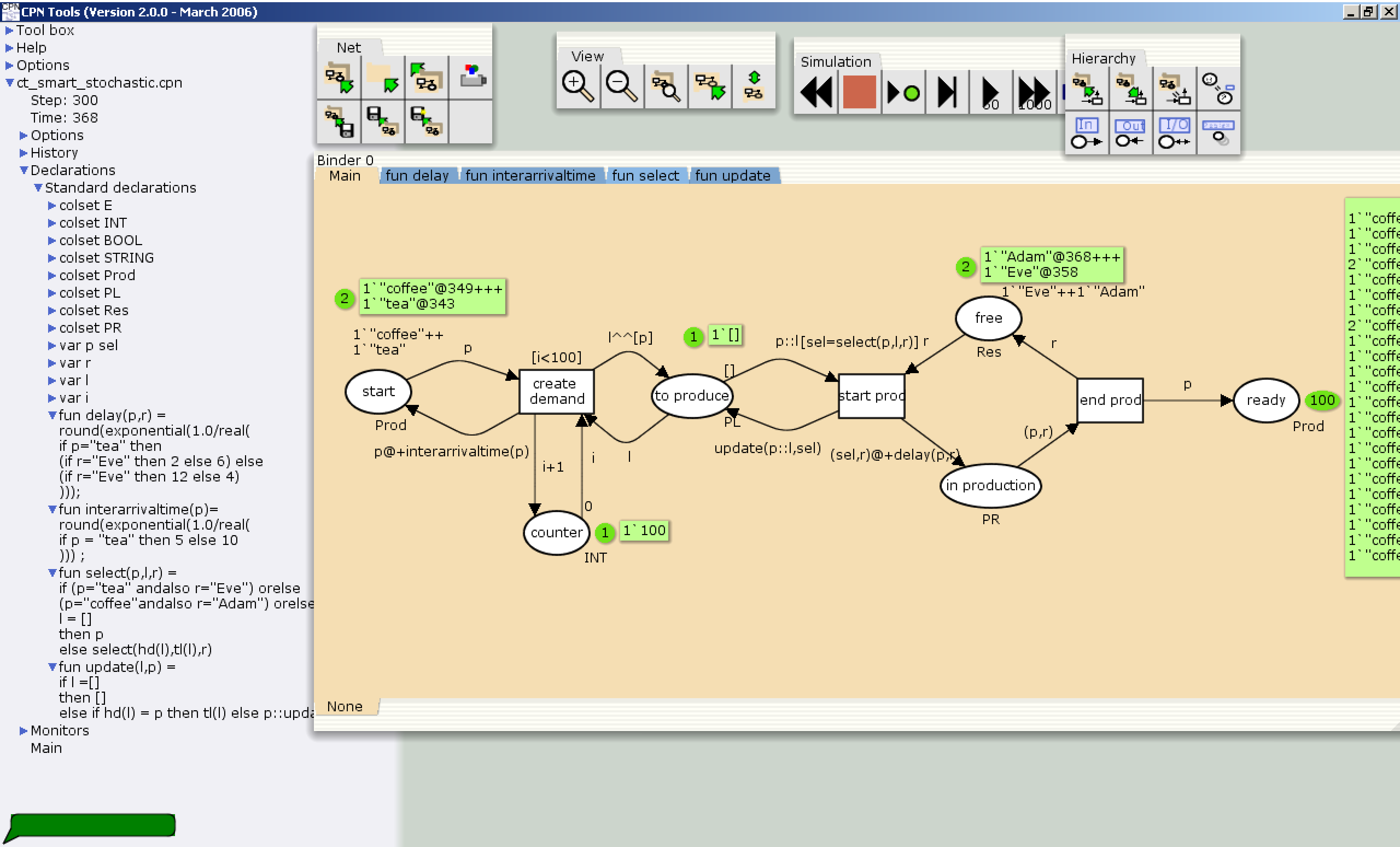
alternative notation [b]%v = if b then 1`v else empty



Adding hierarchy



Example revisited

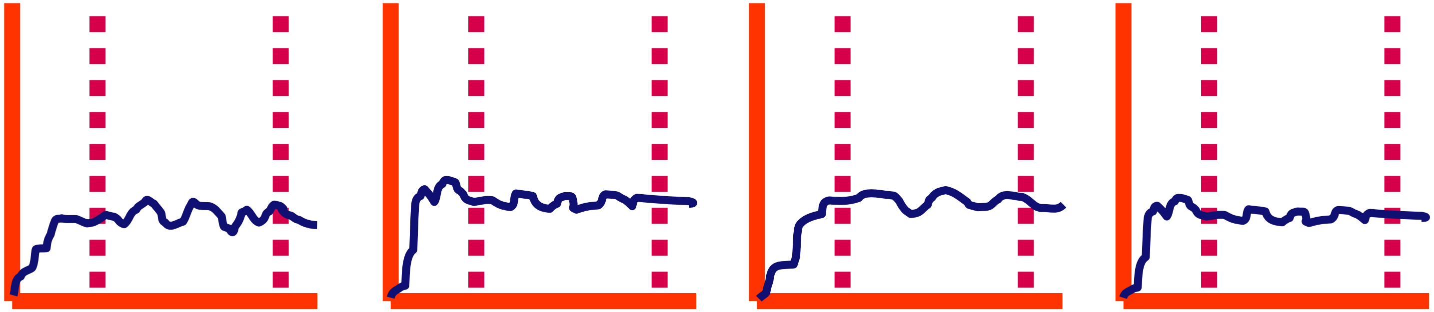


Subruns and confidence intervals

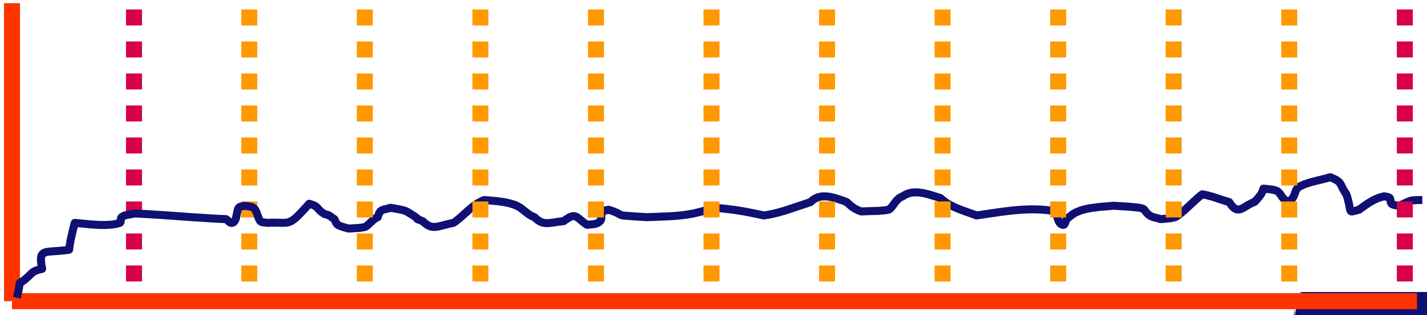
- **A single run does not provide information about reliability of results.**
- **Therefore, multiple runs or one run cut into parts: subruns.**
- **If the subruns are assumed to be mutually independent, one can calculate a confidence interval, e.g., the flow time is with 95% confidence within the interval 5.5+/-0.5 (i.e. [5,6]).**

Two possible settings

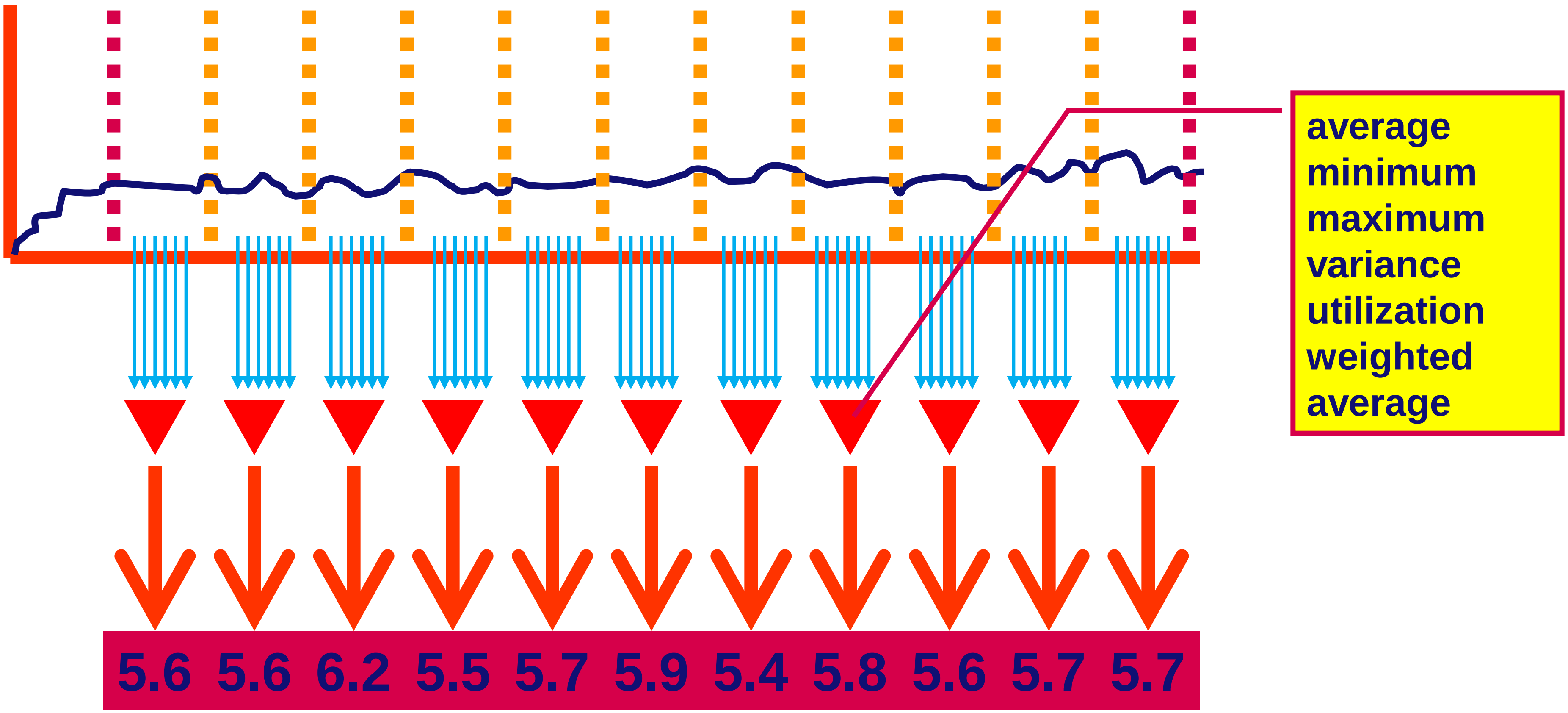
Steady-state analysis (I)



Steady-state analysis (II)



More on calculating confidence intervals

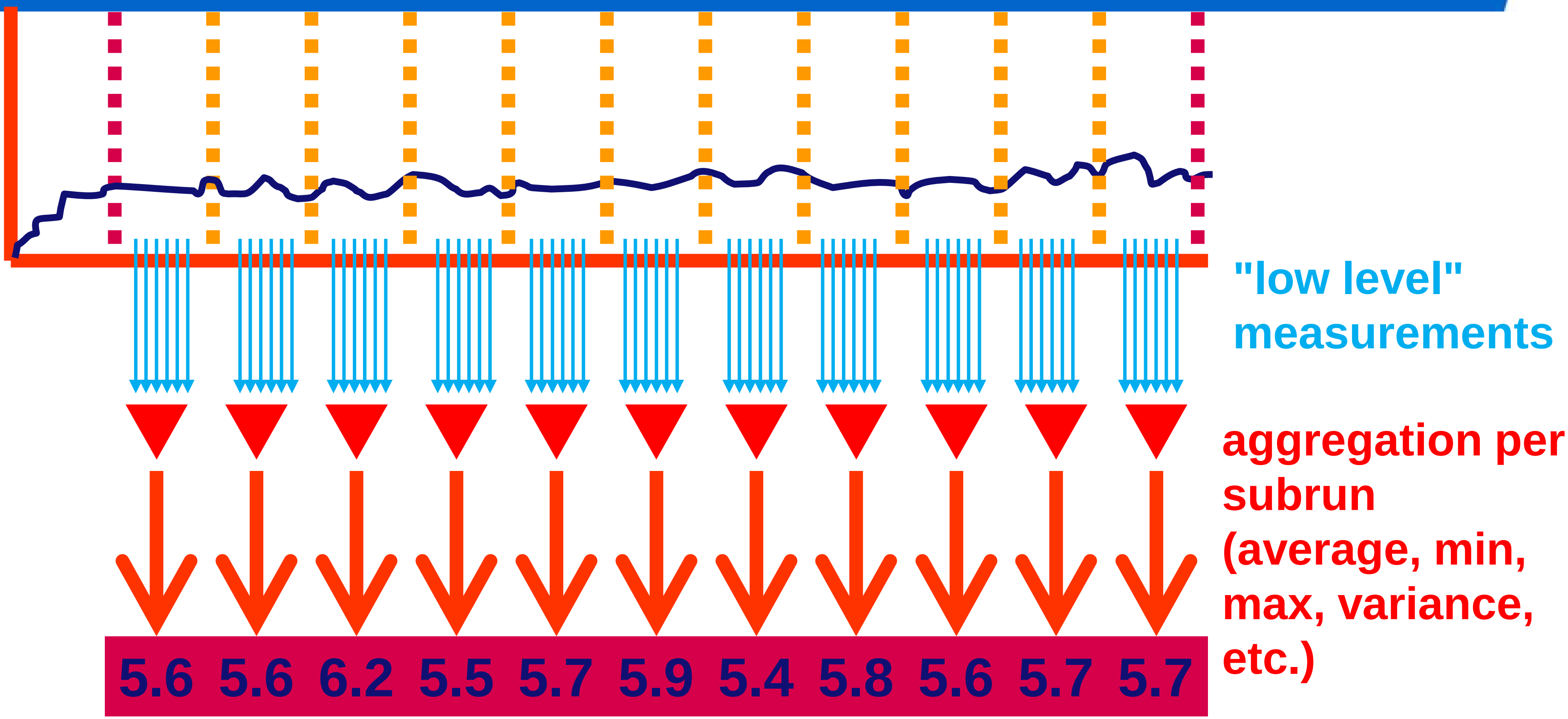


5.6	5.6	6.2	5.5	5.7	5.9	5.4	5.8	5.6	5.7	5.7

is not the same as

4.6	6.6	3.2	8.5	1.7	9.9	4.4	6.8	4.6	6.7	5.7

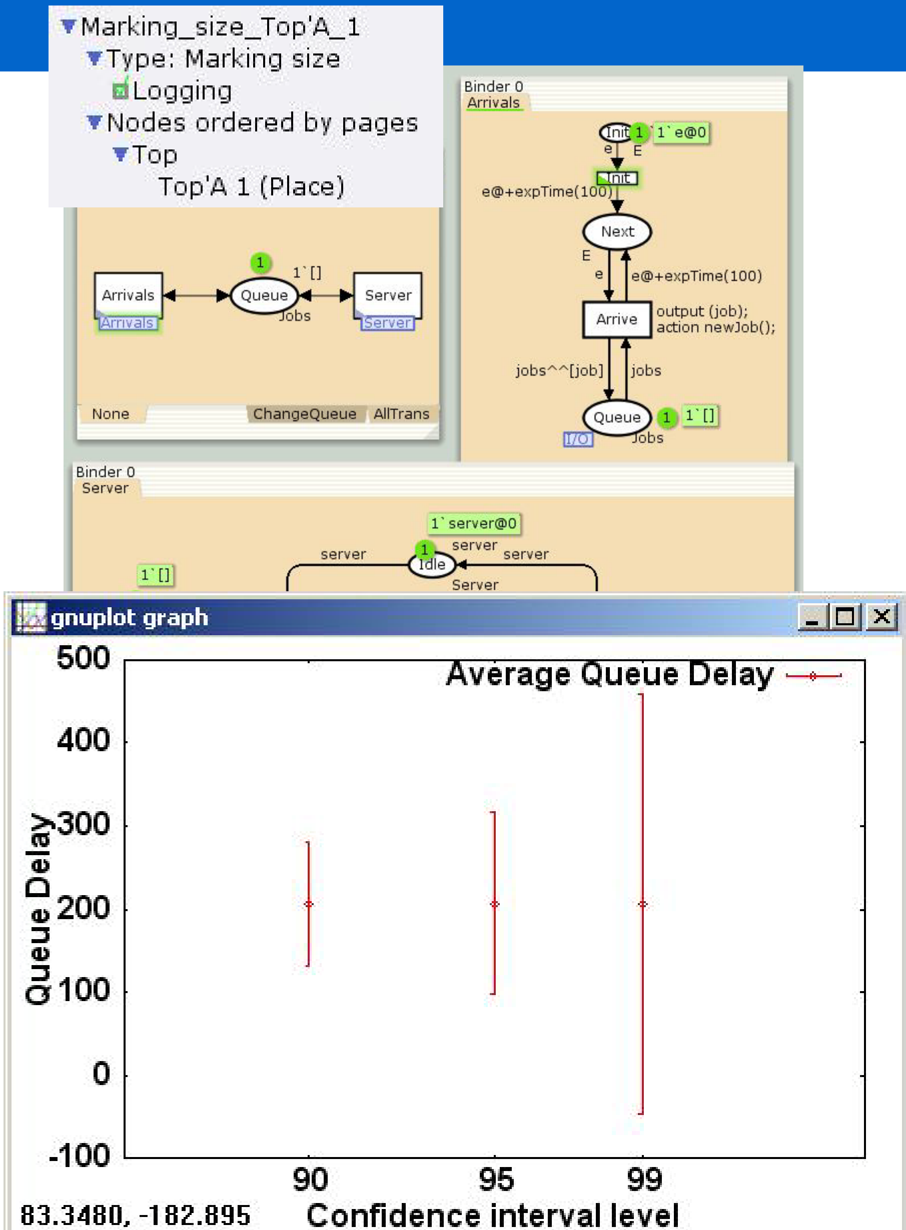
although the average over the subrun results is the same (5.7)



subruns = 11
average = 5.7
standard deviation = 0.21

confidence = 0.9
confidence interval = [5.7-0.117,5.7+0.117] = [5.58,5.82]

Using monitors in CPN Tools



CPN Tools Simulation Performance Report
Net: C:\nets\QueueSystem\QueueSystem.cpn

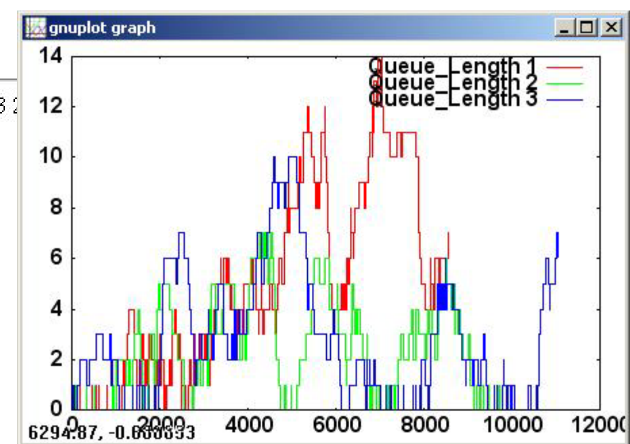
Note that these statistics have been calculated for data that is not necessarily independent or identically distributed.

Timed statistics					
Name	Count	Avrg	Min	Max	Time Interval
Marking_size_Server'Busy_1	201	0.828407	0	1	11131
Queue_Length	209	1.992993	0	9	11131
Server_Utilization	118	0.828407	0	1	11131

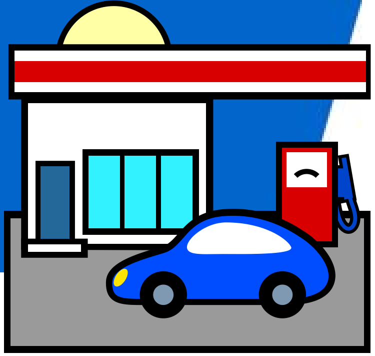
Untimed statistics						
Name	Count	Sum	Avrg	StD	Min	Max
Count_trans_occur_Arrivals'Arrive_1	107	107	1.000000	0.000000	1	1
Processed_A_Jobs	46	46	1.000000	0.000000	1	1
Queue_Delay	100	19933	199.330000	252.527148	0	1255

Simulation steps executed: 307
Model time: 11131

Generated: Wed Feb 15 15:11:03

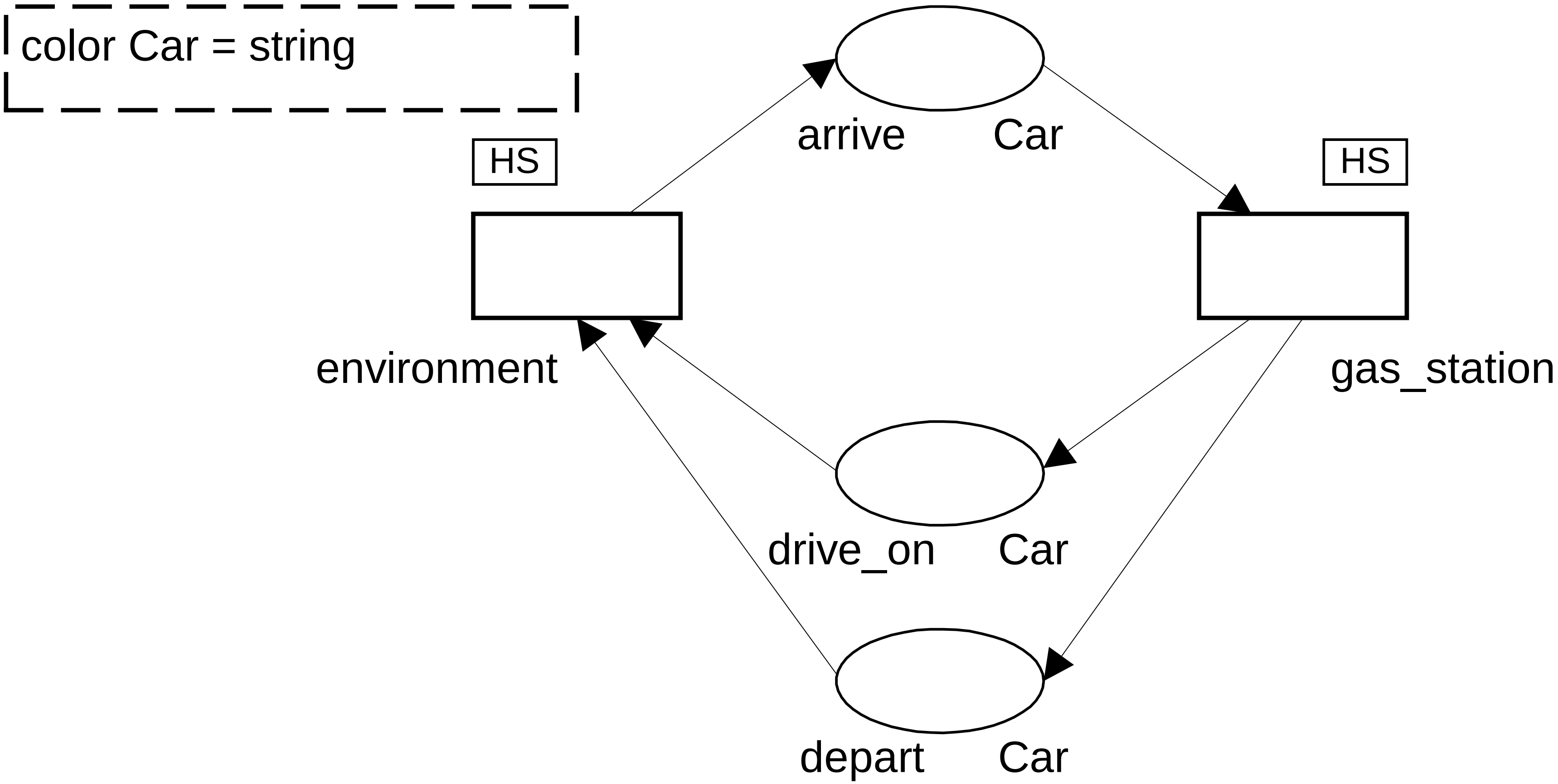


Example of a simulation model



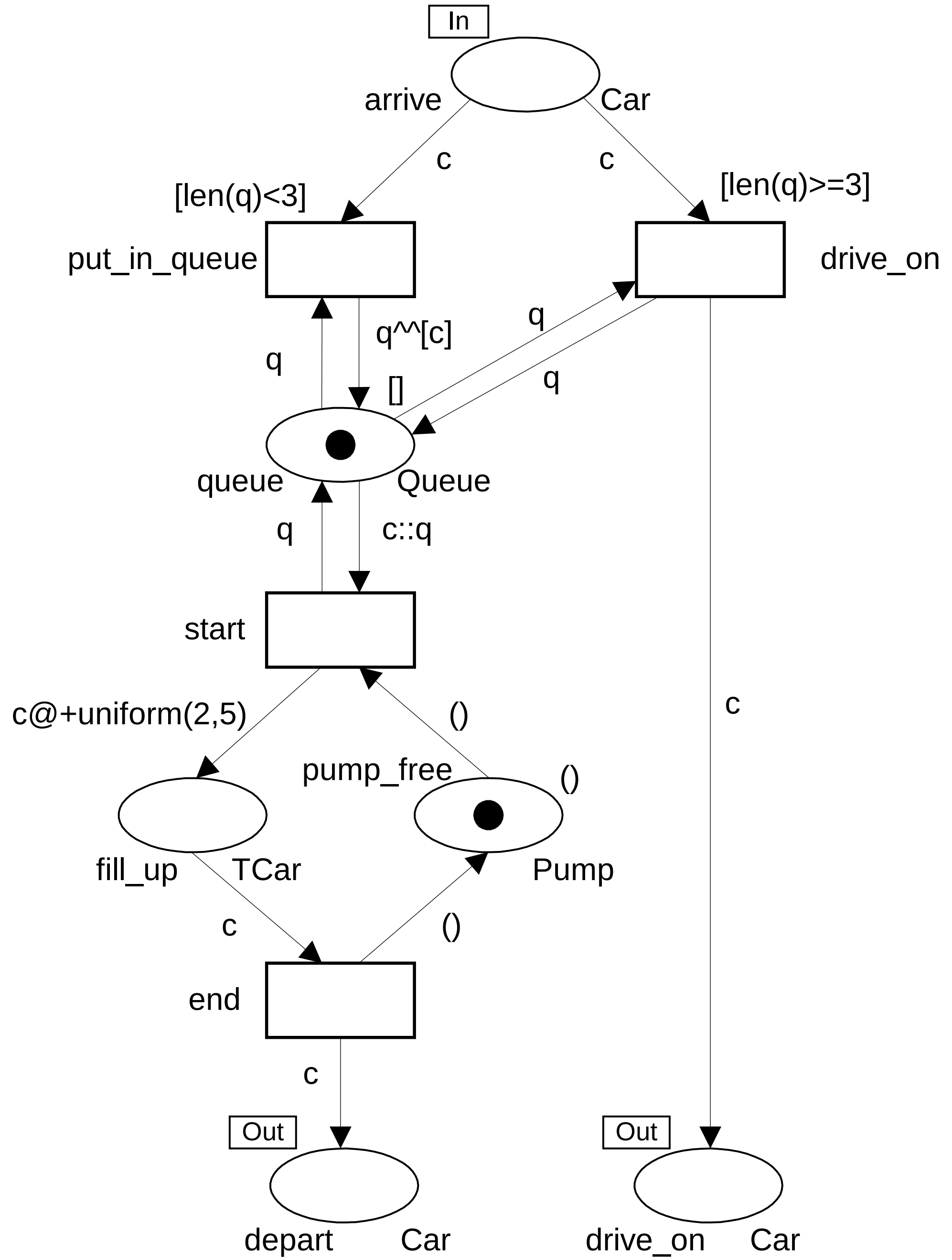
- Gas station with one pump and space for 4 cars (3 waiting and 1 being served).
- Service time: uniform distribution between 2 and 5 minutes.
- Poisson arrival process with mean time between arrivals of 4 minutes.
- If there are more than 3 cars waiting, the "sale" is lost.
- Questions: flow time, waiting time, utilization, lost sales, etc.

Top-level page: *main*



Subpage *gas_station*

```
color Car = string;
color Pump = unit;
color TCar = Car timed;
color Queue = list Car;
var c:Car;
var q:Queue;
fun len(q:Queue) = if q=[] then 0
  else 1+len(tl(q));
```

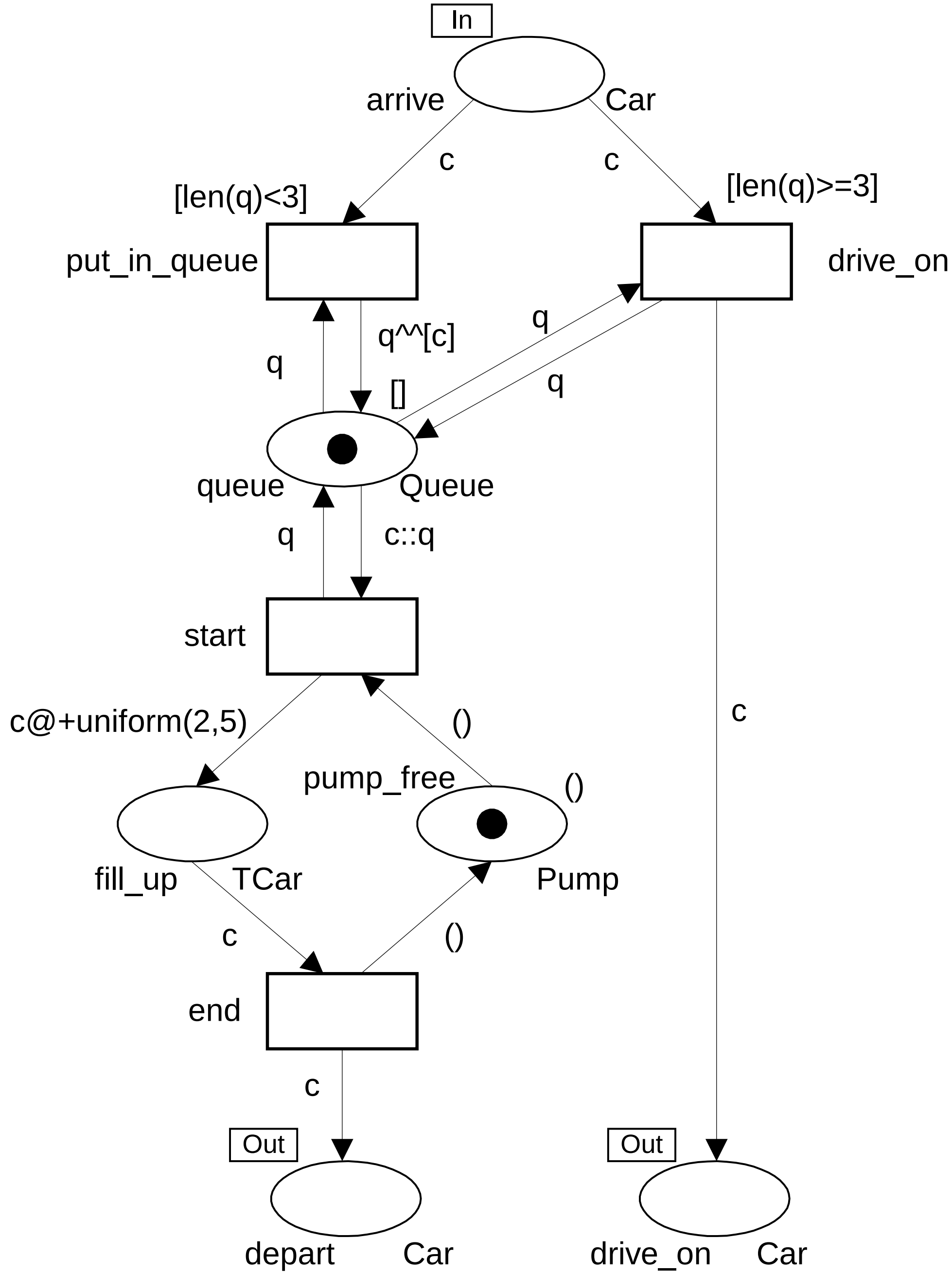


Interesting performance indicators:

- Calculation of flow time (average, variance, maximum, minimum, service level, etc.).
- Calculation of waiting times (average, variance, maximum, minimum, service level, etc.).
- Calculation of lost sales (average).
- Probability of no space left.
- Probability of no cars waiting.

Alternatives

```
color Car = string;
color Pump = unit;
color TCar = Car timed;
color Queue = list Car;
var c:Car;
var q:Queue;
fun len(q:Queue) = if q=[] then 0
  else 1+len(tl(q));
```

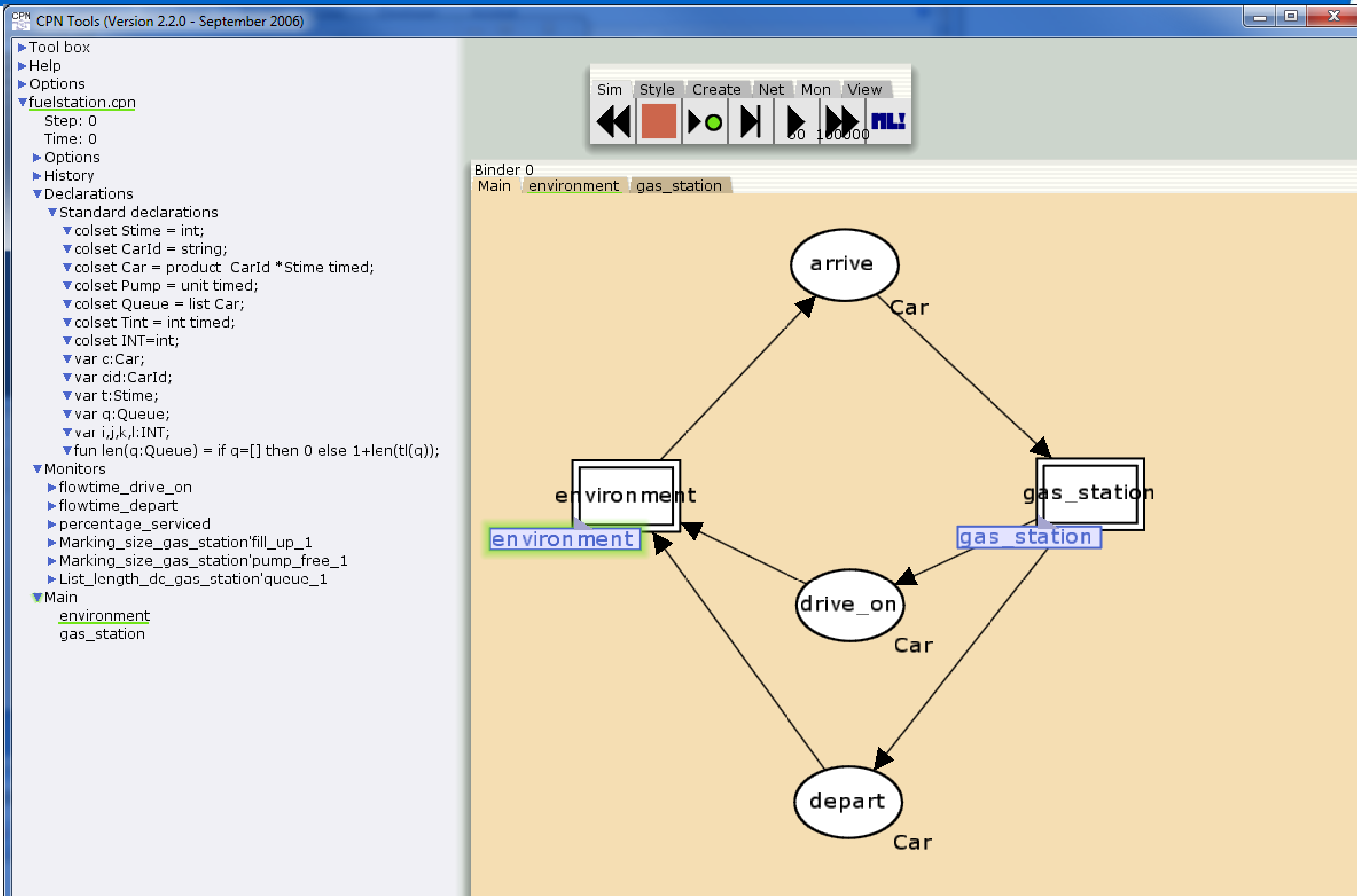


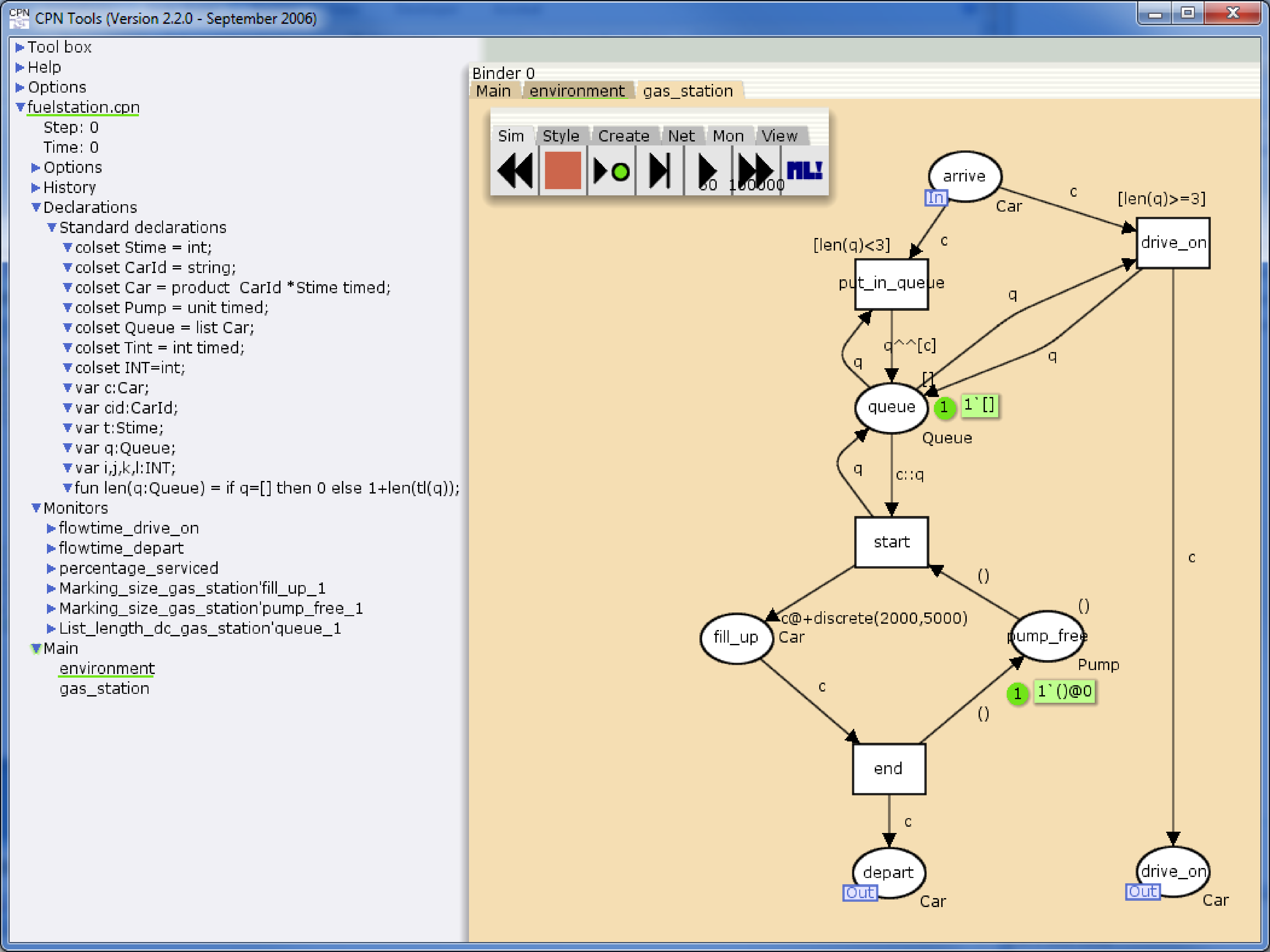
Model the following alternatives:

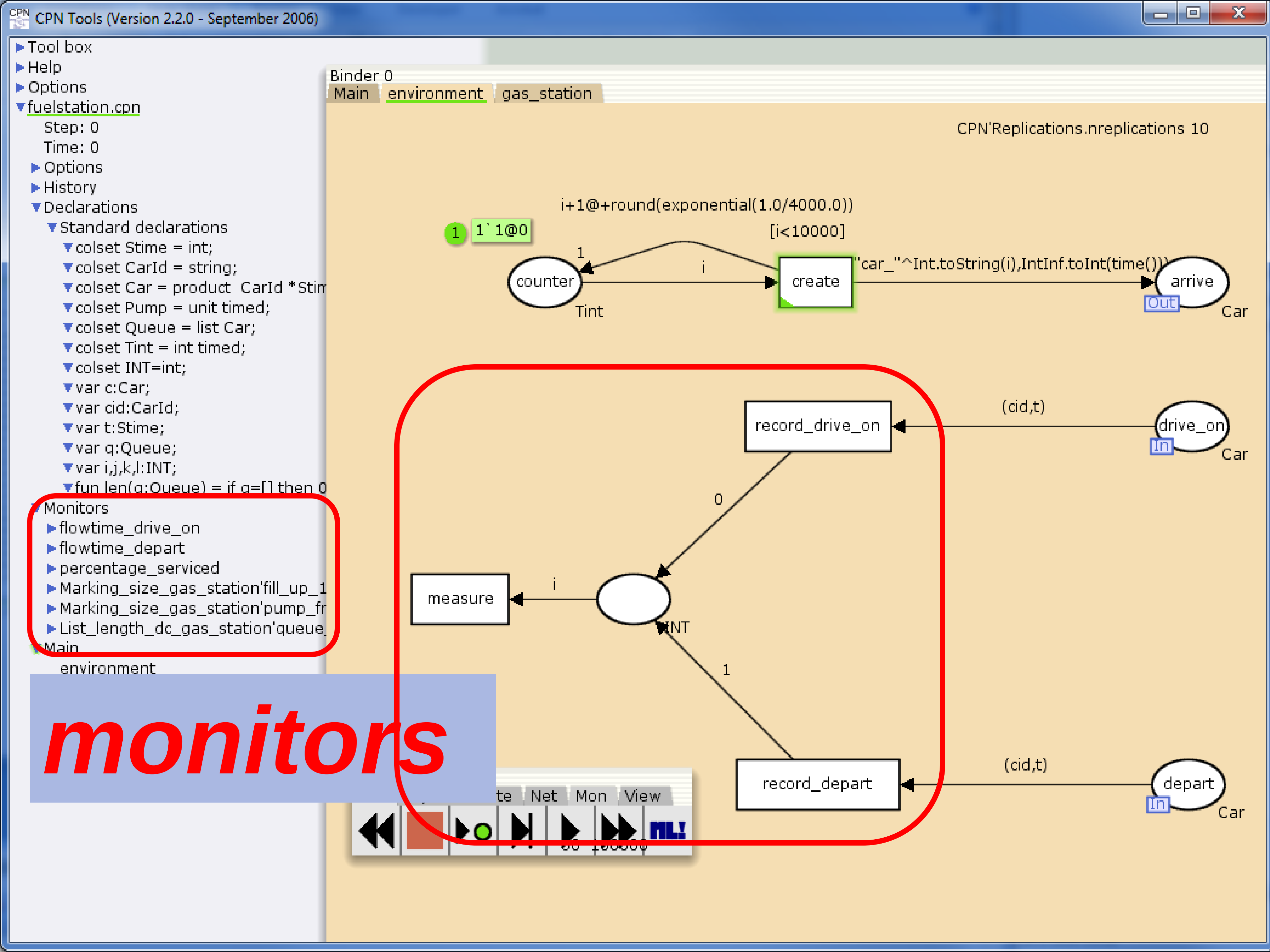
- **6 waiting spaces**
- **2 pumps**
- **1 faster pump**

Experiments

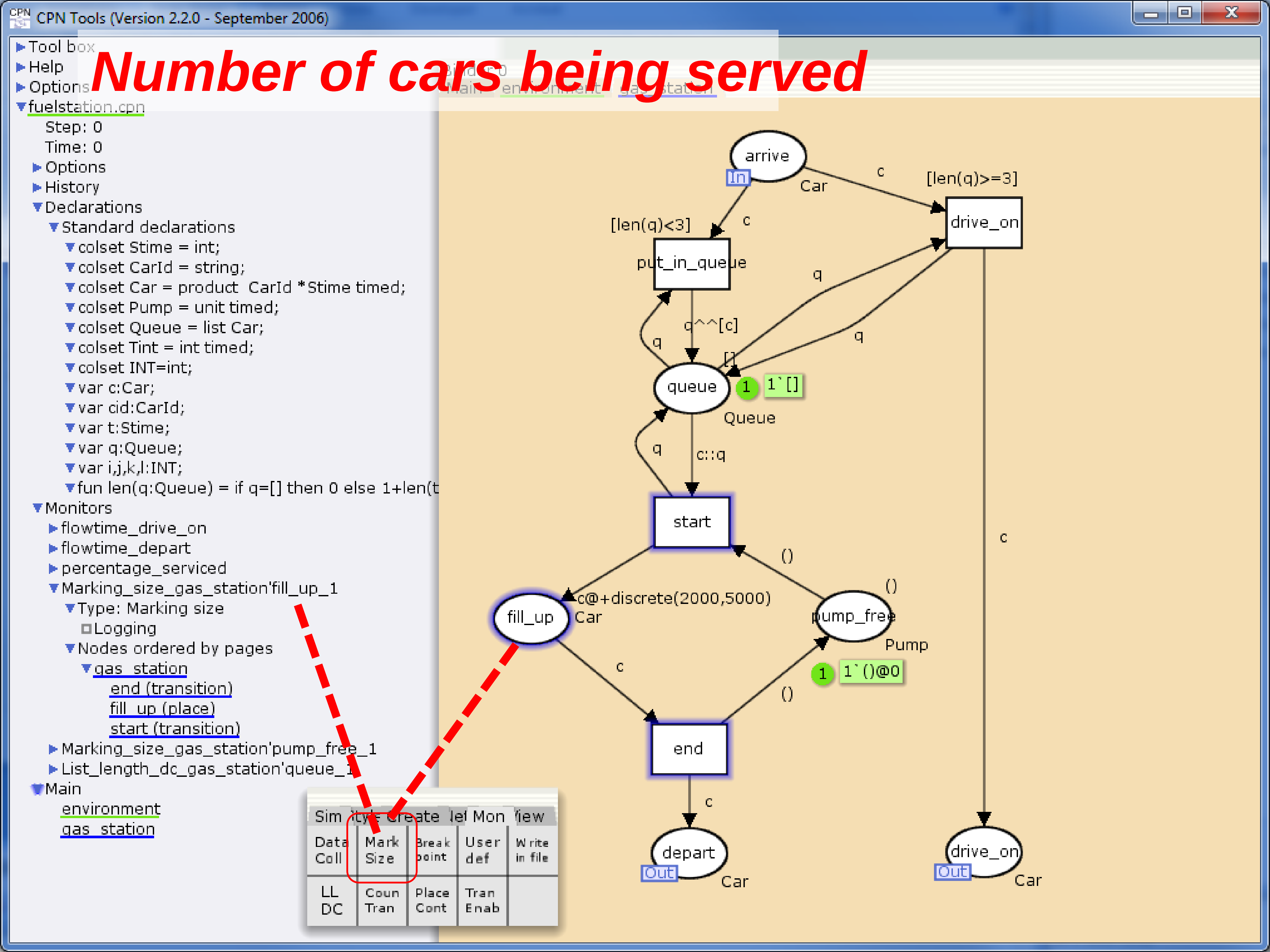
(note time dimension * 1000; not needed in CPN Tools Version 3)



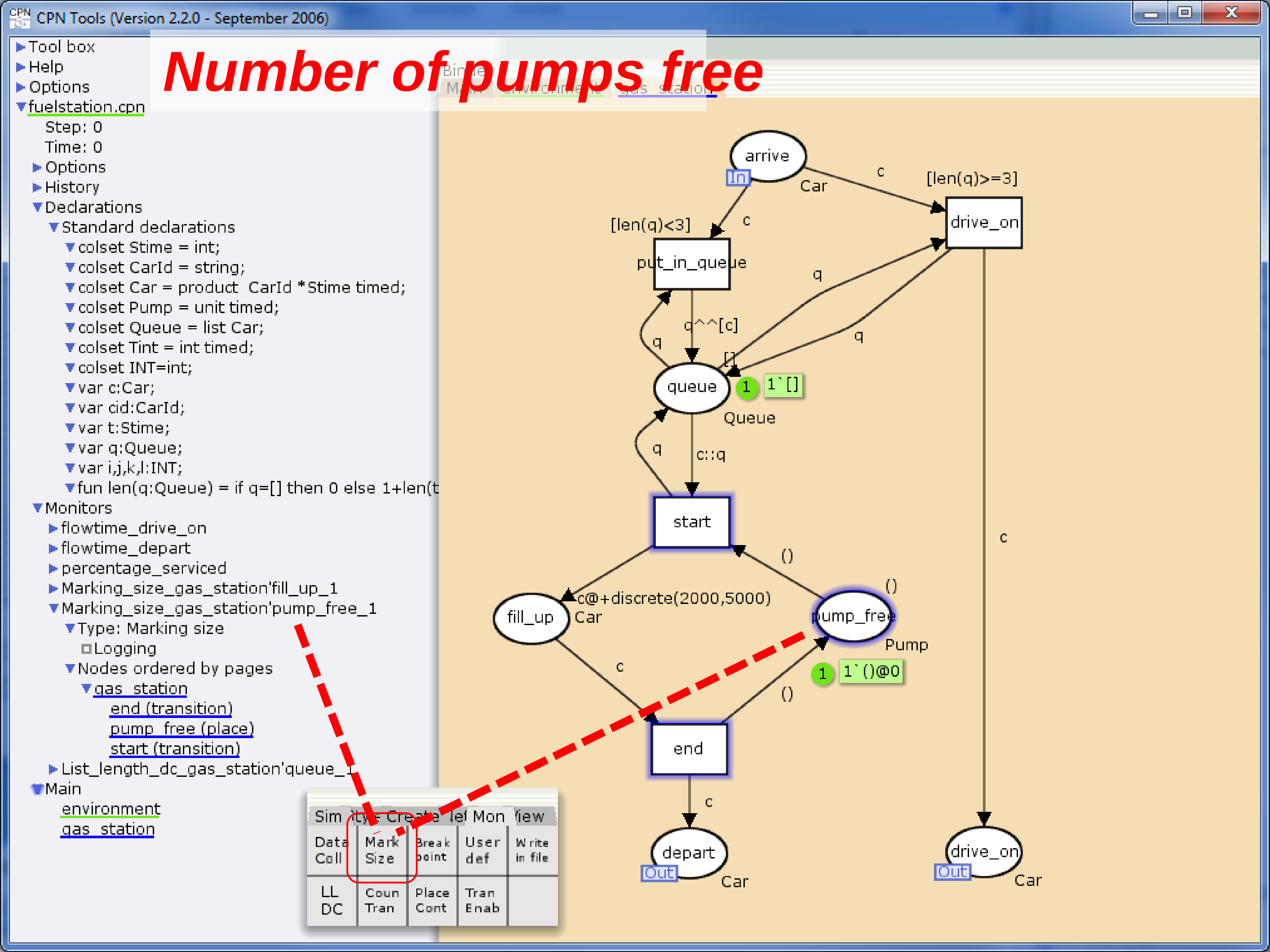

CPN Tools (Version 2.2.0 - September 2006)
Tool box
Help
Options
fuelstation.cpn
Step: 0
Time: 0
Options
History
Declarations
Standard declarations
colset Stime = int;
colset CarId = string;
colset Car = product CarId *Stime timed;
colset Pump = unit timed;
colset Queue = list Car;
colset Tint = int timed;
colset INT=int;
var c:Car;
var cid:CarId;
var t:Stime;
var q:Queue;
var i,j,k,l:INT;
fun len(q:Queue) = if q=[] then 0 else 1+len(tl(q));
Monitors
flowtime_drive_on
flowtime_depart
percentage_serviced
Marking_size_gas_station'fill_up_1
Marking_size_gas_station'pump_free_1
List_length_dc_gas_station'queue_1
Main
environment
gas_station
Binder 0
Main
environment
gas_station
Sim
Style
Create
Net
Mon
View
arrive
In
Car
c
[len(q)>=3]
drive_on
[len(q)<3]
c
put_in_queue
q
q^^[c]
q
q
[]
queue
1
1`[]
Queue
q
c::q
start
c
()
()
c@+discrete(2000,5000)
fill_up
Car
pump_free
Pump
c
1
1`()@0
()
end
c
depart
Out
Car
drive_on
Out
Car


CPN Tools (Version 2.2.0 - September 2006)
Tool box
Help
Options
fuelstation.cpn
Step: 0
Time: 0
Options
History
Declarations
Standard declarations
colset Stime = int;
colset CarId = string;
colset Car = product CarId * Stim
colset Pump = unit timed;
colset Queue = list Car;
colset Tint = int timed;
colset INT=int;
var c:Car;
var cid:CarId;
var t:Stime;
var q:Queue;
var i,j,k,l:INT;
fun len(q:Queue) = if q=[] then 0
Monitors
flowtime_drive_on
flowtime_depart
percentage_serviced
Marking_size_gas_station'fill_up_1
Marking_size_gas_station'pump_fr
List_length_dc_gas_station'queue_
Main
environment
Binder 0
Main
environment
gas_station
CPN'Replications.nreplications 10
i+1@+round(exponential(1.0/4000.0))
[i<10000]
1`1@0
counter
Tint
create
"car_"^Int.toString(i),IntInf.toInt(time())
arrive
Out
Car
record_drive_on
(cid,t)
drive_on
In
Car
measure
INT
record_depart
(cid,t)
depart
In
Car
Net
Mon
View
monitors

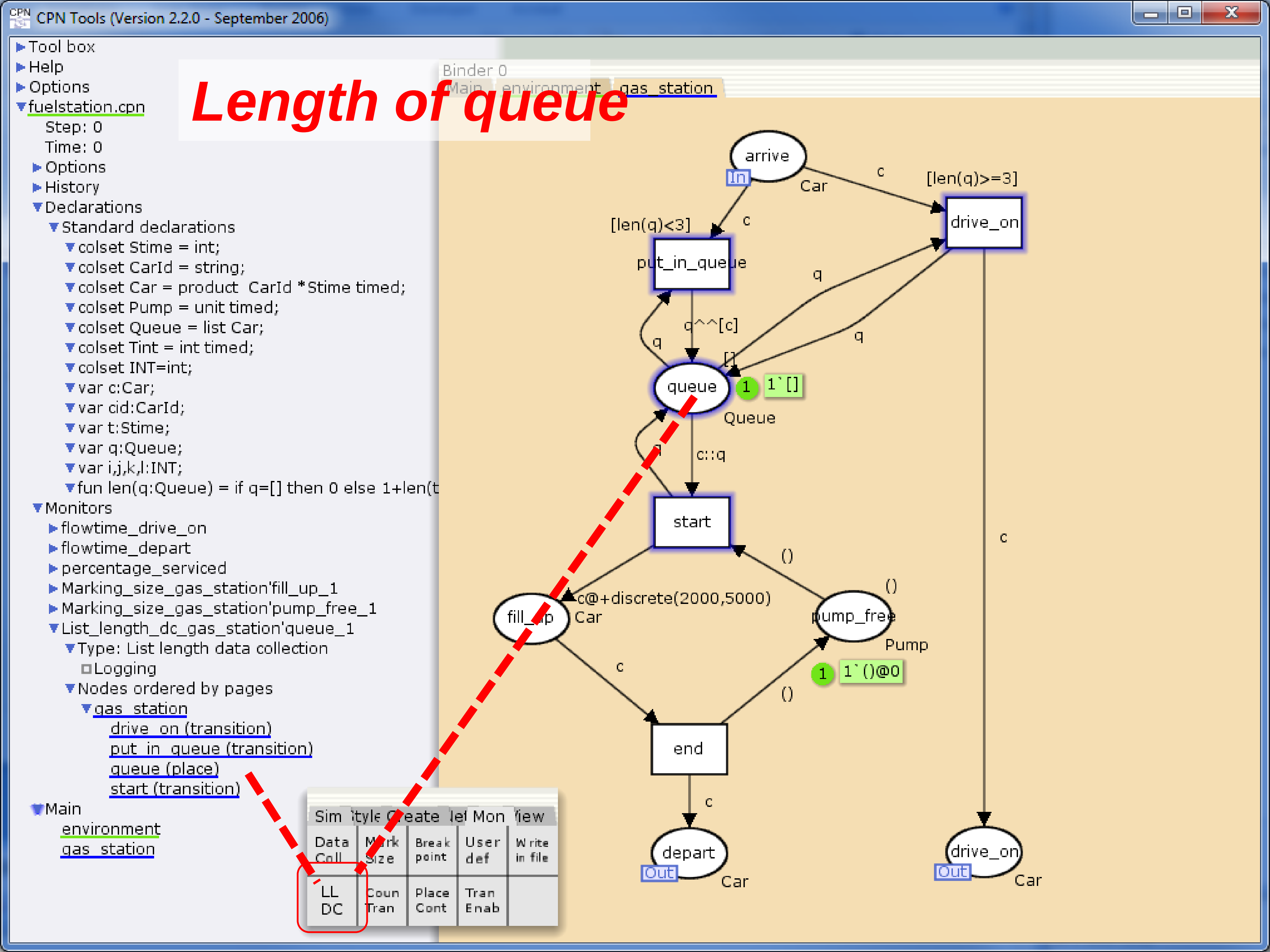
Number of cars being served

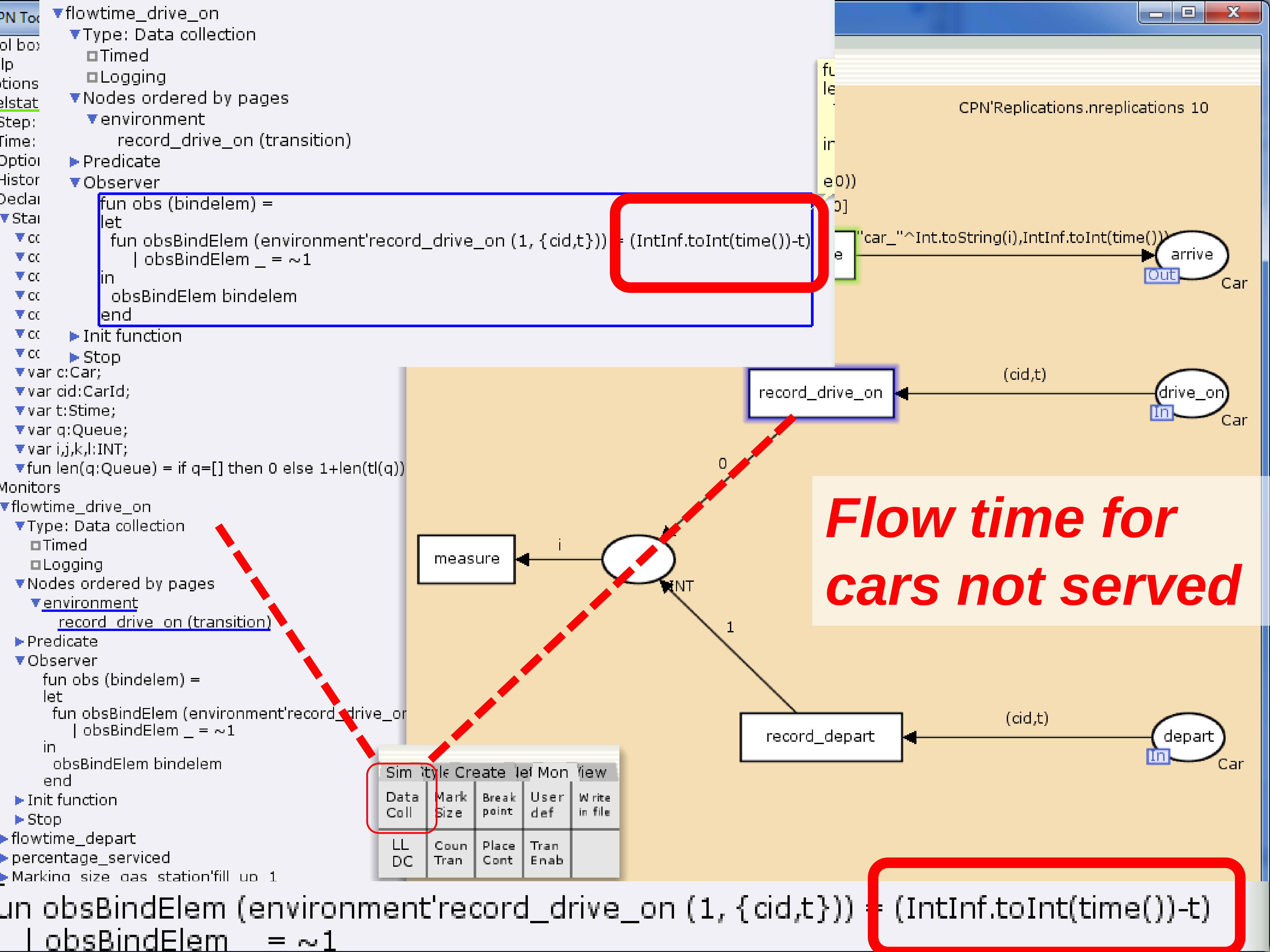


Number of pumps free

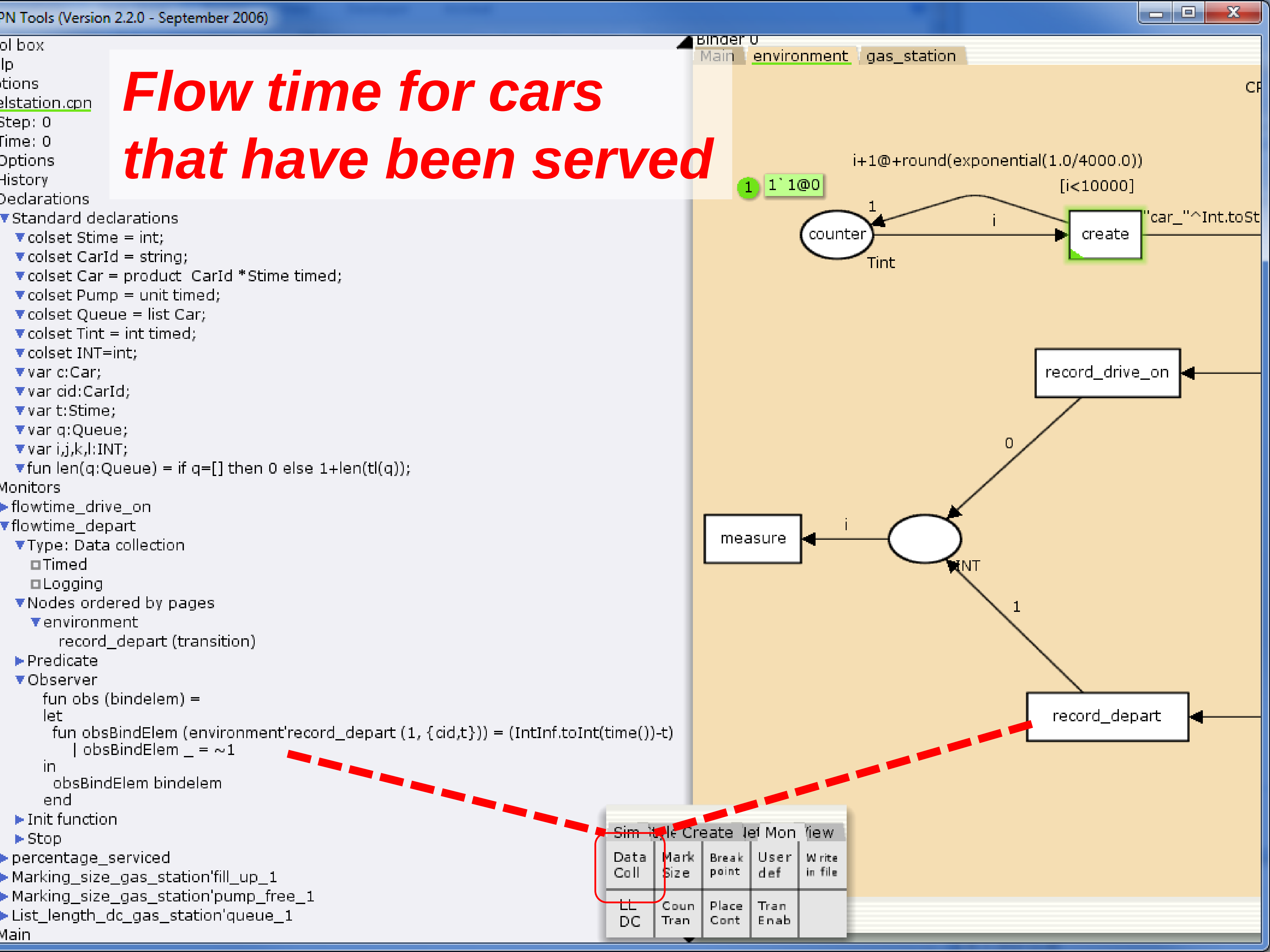


Length of queue

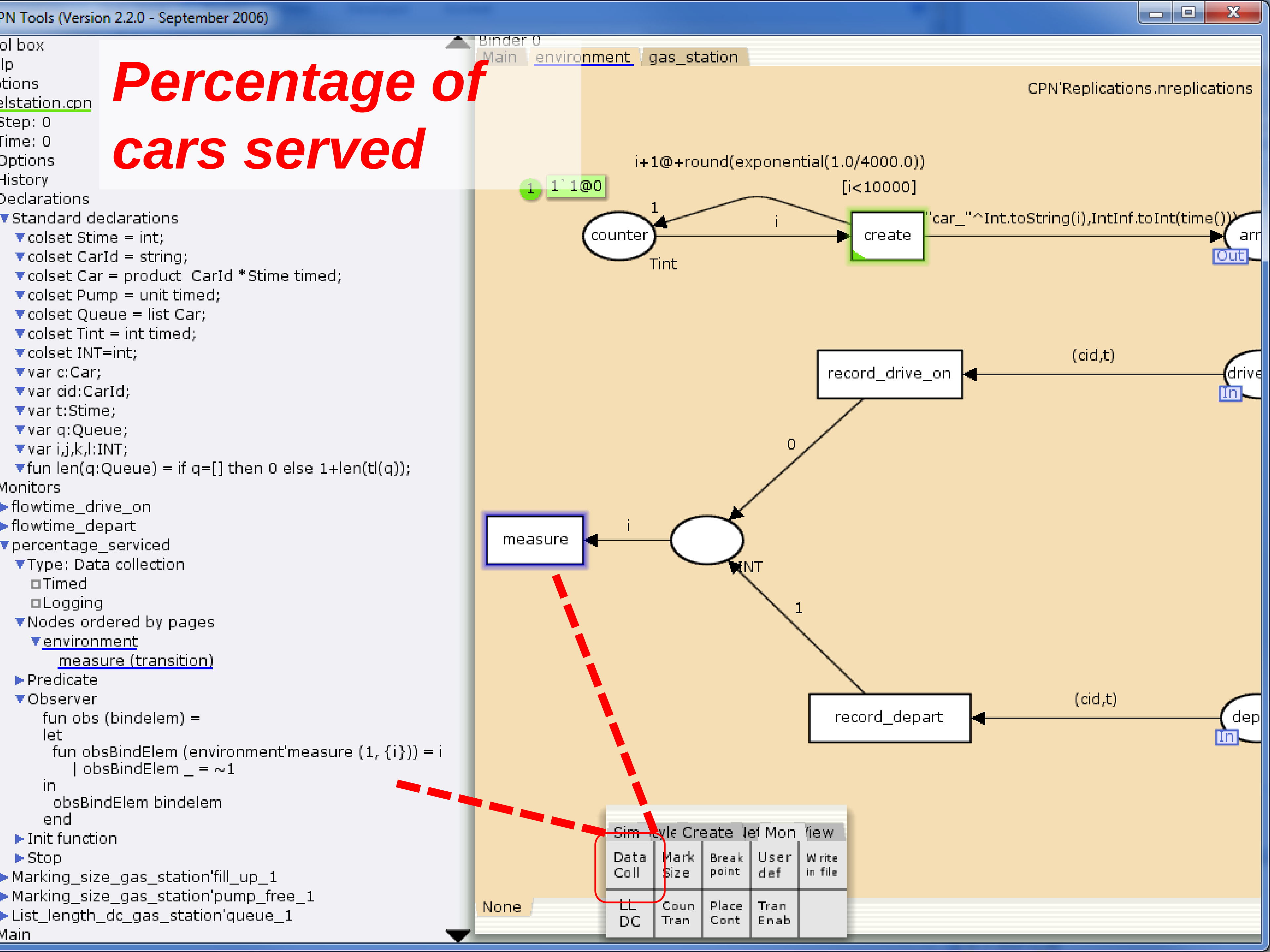



flowtime_drive_on
Type: Data collection
Timed
Logging
Nodes ordered by pages
environment
record_drive_on (transition)
Predicate
Observer
fun obs (bindelem) =
let
fun obsBindElem (environment'record_drive_on (1, {cid,t})) = (IntInf.toInt(time())-t)
| obsBindElem _ = ~1
in
obsBindElem bindelem
end
Init function
Stop
var c:Car;
var cid:CarId;
var t:Stime;
var q:Queue;
var i,j,k,l:INT;
fun len(q:Queue) = if q=[] then 0 else 1+len(tl(q))
Monitors
flowtime_depart
percentage_serviced
CPN'Replications.nreplications 10
"car_"^Int.toString(i),IntInf.toInt(time())
arrive
Out
Car
(cid,t)
record_drive_on
drive_on
In
Car
0
measure
i
INT
1
record_depart
(cid,t)
depart
In
Car
Flow time for
cars not served
Sim
Create
Mon
Data Coll
Mark Size
Break point
User def
Write in file
LL DC
Coun Tran
Place Cont
Tran Enab
fun obsBindElem (environment'record_drive_on (1, {cid,t})) = (IntInf.toInt(time())-t)
| obsBindElem _ = ~1

Flow time for cars that have been served



Percentage of cars served



Executes the specified number of transitions without showing the intermediate markings.

Sim Style Create Net Mon

CPN Tools Simulation Pe...

file:///D:/courses/BIS-2010/CPN/newsimulation/output/PerfReport.html

CPN Tools Simulation Performance Report
Net: D:\courses\BIS-2010\CPN\newsimulation\fuelstation.cpn

Note that these statistics have been calculated for data that is not necessarily independent or identically distributed.

Timed statistics				
Name	**Count**	**Avrg**	**Min**	**Max**
List_length_dc_gas_station'queue_1	19130	0.888529	0	3
Marking_size_gas_station'fill_up_1	18258	0.799214	0	1
Marking_size_gas_station'pump_free_1	18258	0.200786	0	1

Untimed statistics					
Name	**Count**	**Sum**	**Avrg**	**Min**	**Max**
flowtime_depart	9128	67178542	7359.612401	2005	17997
flowtime_drive_on	872	0	0.000000	0	0
percentage_serviced	10000	9128	0.912800	0	1

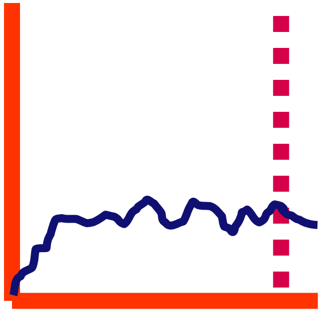
Simulation steps executed: 58256
Model time: 39803766

Generated: Sun Mar 28 19:04:26 2010

Average flow is time 7.359

Just one run ...

Subruns in CPN Tools



x 10

CPN'Replications.nreplications 10

CPN Tools Simulation Pe... | CPN Tools Performance ...

file:///D:/courses/BIS-2010/CPN/newsimulation/output/reps_1/PerfReportIID.html

CPN Tools Performance Report
Net: D:\courses\BIS-2010\CPN\newsimulation\fuelstation.cpn
Number of replications: 10

Statistics							
Name	**Avrg**	**90% Half Length**	**95% Half Length**	**99% Half Length**	**StD**	**Min**	**Max**
List_length_dc_gas_station'queue_1							
count_iid	19128.800000	25.132533	31.014615	44.561228	43.358454	19056	19181
max_iid	3.000000	0.000000	0.000000	0.000000	0.000000	3	3
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.891917	0.014062	0.017353	0.024932	0.024259	0.841207	0.924649
Marking_size_gas_station'fill_up_1							
count_iid	18255.600000	50.265065	62.029230	89.122456	86.716909	18110	18360
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.797933	0.003853	0.004755	0.006832	0.006648	0.790479	0.810488
Marking_size_gas_station'pump_free_1							
count_iid	18255.600000	50.265065	62.029230	89.122456	86.716909	18110	18360
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.202067	0.003853	0.004755	0.006832	0.006648	0.189512	0.209521

Average queue length [0.878,0.906]

Average # pumps busy [0.794,0.801]

Average # pumps free [0.198,0.206]

CPN Tools Simulation Pe... | CPN Tools Performance ...

file:///D:/courses/BIS-2010/CPN/newsimulation/output/reps_1/PerfReportIID.html

avrg_iid	0.202067	0.003853	0.004755	0.006832	0.006648	0.189512	0.209521
flowtime_depart							
count_iid	9126.800000	25.132533	31.014615	44.561228	43.358454	9054	9179
max_iid	18192.300000	181.666042	224.183626	322.102911	313.408874	17593	18615
min_iid	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	2000	[illegible]
sum_iid	67592080.300000	438346.885657	540938.709960	777210.790172	756232.713539	66432681	68943753
avrg_iid	7406.069819	53.540469	66.071218	94.929910	92.367611	7237.463885	7523.325295
flowtime_drive_on							
count_iid	873.200000	25.132533	31.014615	44.561228	43.358454	821	946
max_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
percentage_serviced							
count_iid	10000.000000	0.000000	0.000000	0.000000	0.000000	10000	10000
max_iid	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	9126.800000	25.132533	31.014615	44.561228	43.358454	9054	9179
avrg_iid	0.912680	0.002513	0.003101	0.004456	0.004336	0.905400	0.917900

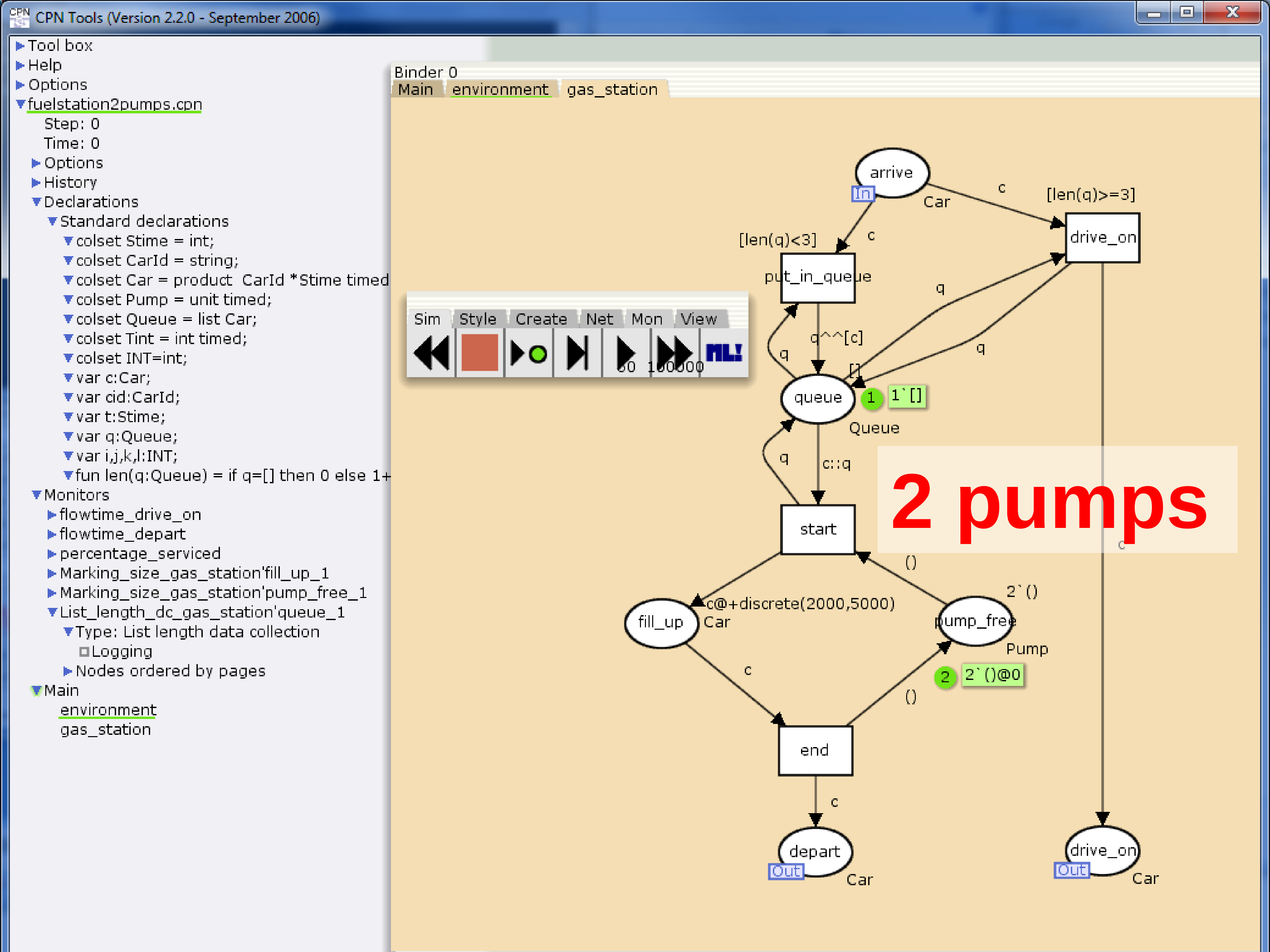
Generated: Sun Mar 28 19:14:39 2010

Average flow time [7.352,7.460]

Average fraction served [0.910,0.915]

Results

	Average flow time	Average fraction served
Base case	[7.352,7.460]	[0.910,0.915]



CPN Tools (Version 2.2.0 - September 2006)
Tool box
Help
Options
fuelstation2pumps.cpn
Step: 0
Time: 0
Options
History
Declarations
Standard declarations
colset Stime = int;
colset CarId = string;
colset Car = product CarId * Stime timed
colset Pump = unit timed;
colset Queue = list Car;
colset Tint = int timed;
colset INT=int;
var c:Car;
var cid:CarId;
var t:Stime;
var q:Queue;
var i,j,k,l:INT;
fun len(q:Queue) = if q=[] then 0 else 1+
Monitors
flowtime_drive_on
flowtime_depart
percentage_serviced
Marking_size_gas_station'fill_up_1
Marking_size_gas_station'pump_free_1
List_length_dc_gas_station'queue_1
Type: List length data collection
Logging
Nodes ordered by pages
Main
environment
gas_station
Binder 0
Main
environment
gas_station
Sim
Style
Create
Net
Mon
View
50
100000
ML!
arrive
In
Car
c
[len(q)>=3]
drive_on
[len(q)<3]
c
put_in_queue
q
q^^[c]
q
q
[]
queue
1
1`[]
Queue
q
c::q
2 pumps
start
()
2`()
fill_up
c@+discrete(2000,5000)
Car
pump_free
Pump
2
2`()@0
c
()
end
c
depart
Out
Car
drive_on
Out
Car

CPN Tools Performance ...

file:///D:/courses/BIS-2010/CPN/newsimulation/output/reps_2/PerfReportIID.html

CPN Tools Performance Report
Net: D:\courses\BIS-2010\CPN\newsimulation\fuelstation2pumps.cpn
Number of replications: 10

Statistics

Name	Avrg	90% Half Length	95% Half Length	99% Half Length	StD	Min	Max
List_length_dc_gas_station'queue_1							
count_iid	19963.900000	3.318017	4.094574	5.883009	5.724218	19955	19972
max_iid	3.000000	0.000000	0.000000	0.000000	0.000000	3	[illegible]
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.107718	0.002875	0.003548	0.005098	0.004960	0.100403	0.116024

Average queue length [0.105,0.111]

Name	Avrg	90% Half Length	95% Half Length	99% Half Length	StD	Min	Max
Marking_size_gas_station'fill_up_1							
count_iid	19925.800000	6.636034	8.189148	11.766017	11.448435	19908	19942
max_iid	2.000000	0.000000	0.000000	0.000000	0.000000	2	2
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.872762	0.005708	0.007044	0.010121	0.009848	0.853645	0.883708

Average # pumps busy [0.867,0.878]

Name	Avrg	90% Half Length	95% Half Length	99% Half Length	StD	Min	Max
Marking_size_gas_station'pump_free_1							
count_iid	19925.800000	6.636034	8.189148	11.766017	11.448435	19908	19942
max_iid	2.000000	0.000000	0.000000	0.000000	0.000000	2	2
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	1.127238	0.005708	0.007044	0.010121	0.009848	1.116292	1.146355

Average # pumps free [1.122,1.133]

CPN Tools Performance ...

file:///D:/courses/BIS-2010/CPN/newsimulation/output/reps_2/PerfReportIID.html

Average flow time [3.916,3.944]

Average fraction served[0.996,0.997]

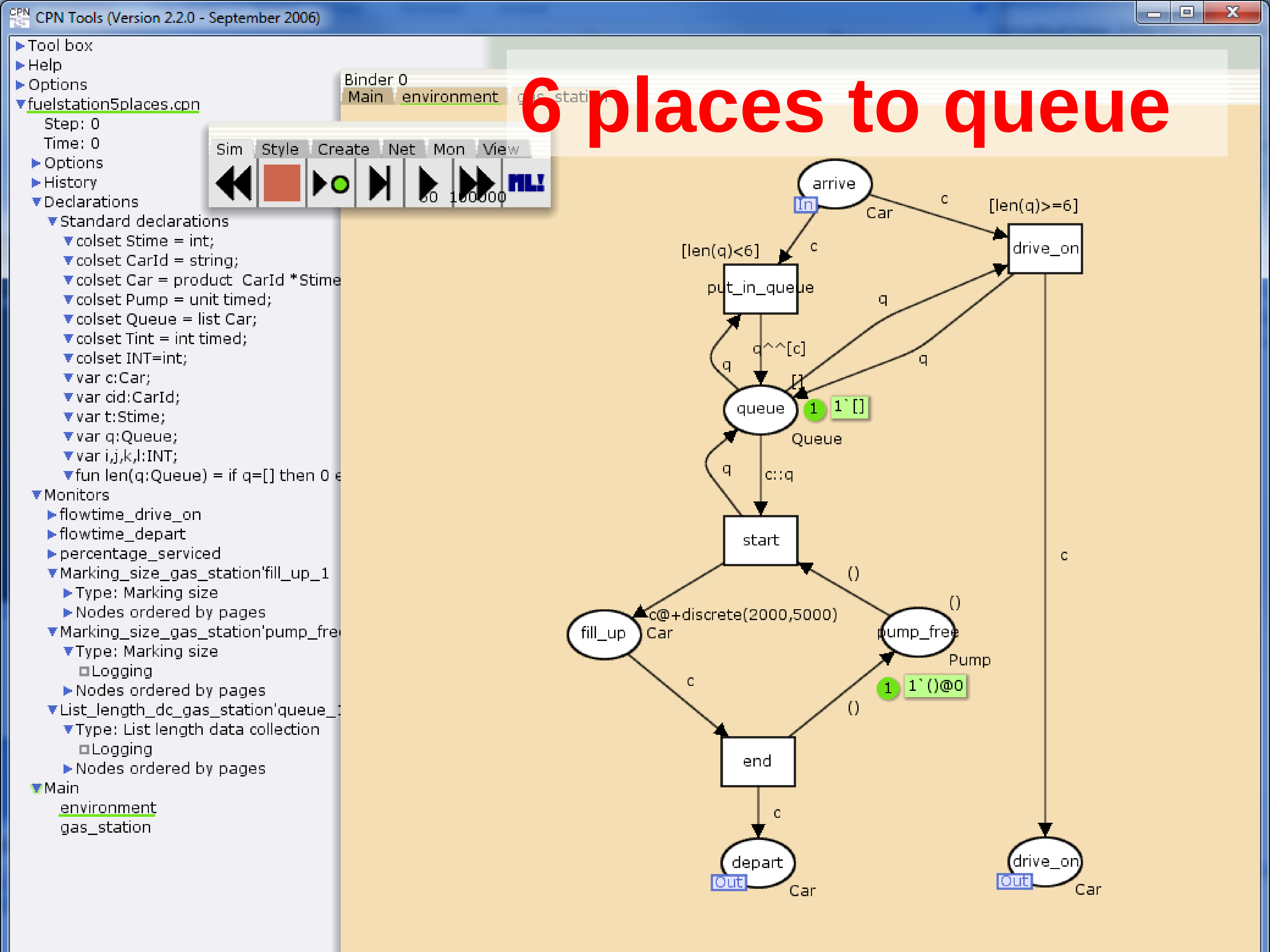
avrg_iid	1.127238	0.005708	0.007044	0.010121	0.009848	1.116292	1.146355
flowtime_depart							
count_iid	9961.900000	3.318017	4.094574	5.883009	5.724218	9953	9970
max_iid	11600.600000	343.702375	424.143357	609.401375	592.952724	10785	12662
min_iid	2009.200000	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
sum_iid	39154411.600000	131855.985224	162715.896659	233787.207843	227476.942953	38778392	39503646
avrg_iid	3930.421027	13.717449	16.927915	24.321718	23.665238	3893.412851	3966.627774
flowtime_drive_on							
count_iid	38.100000	3.318017	4.094574	5.883009	5.724218	30	47
max_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
percentage_serviced							
count_iid	10000.000000	0.000000	0.000000	0.000000	0.000000	10000	10000
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	9961.900000	3.318017	4.094574	5.883009	5.724218	9953	9970
avrg_iid	0.996190	0.000332	0.000409	0.000588	0.000572	0.995300	0.997000

Generated: Sun Mar 28 19:37:17 2010

Results

	Average flow time	Average fraction served
Base case	[7.352,7.460]	[0.910,0.915]
Two pumps	[3.916,3.944]	[0.996,0.997]

6 places to queue



CPN Tools Performance ... | CPN Tools Performance ...

file:///D:/courses/BIS-2010/CPN/newsimulation/output/reps_3/PerfReportIID.html

CPN Tools Performance Report
Net: D:\courses\BIS-2010\CPN\newsimulation\fuelstation5places.cpn
Number of replications: 10

Statistics							
Name	**Avrg**	**90% Half Length**	**95% Half Length**	**99% Half Length**	**StD**	**Min**	**Max**
List_length_dc_gas_station'queue_1							
count_iid	19680.900000	18.247226	22.517854	32.353238	31.479976	19632	19746
max_iid	6.000000	0.000000	0.000000	0.000000	0.000000	6	6
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	1.730337	0.039569	0.048830	0.070159	0.068265	1.648615	1.853813
Marking_size_gas_station'fill_up_1							
count_iid	19359.800000	36.494453	45.035707	64.706476	62.959952	19262	19490
max_iid	[illegible]	0.000000	[illegible]	[illegible]	0.000000	[illegible]	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.845927	0.004984	0.006150	0.008837	0.008598	0.830000	0.860005
Marking_size_gas_station'pump_free_1							
count_iid	19359.800000	36.494453	45.035707	64.706476	62.959952	19262	19490
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	[illegible]	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.154073	0.004984	0.006150	0.008837	0.008598	0.139995	0.170000

Average queue length [1.691,1.770]

Average # pumps busy [0.841,0.851]

Average # pumps free [0.149,0.159]

CPN Tools Performance ... | CPN Tools Performance ...

file:///D:/courses/BIS-2010/CPN/newsimulation/output/reps_3/PerfReportIID.html

avrg_iid	0.154073	0.004984	0.006150	0.008837	0.008598	0.139995	0.170000
flowtime_depart							
count_iid	9678.900000	18.247226	22.517854	32.353238	31.479976	9630	9744
max_iid	30006.900000	299.730152	369.879763	531.436441	517.092180	29374	31188
min_iid	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
sum_iid	103039273.400000	1101932.891192	1359832.078492	1953781.721972	1901046.243765	100848139	106400048
avrg_iid	10646.221288	127.439617	157.265911	225.956768	219.857858	10395.074418	11048.810800
flowtime_drive_on							
count_iid	321.100000	18.247226	22.517854	32.353238	31.479976	256	370
max_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
percentage_serviced							
count_iid	10000.000000	0.000000	0.000000	0.000000	0.000000	10000	10000
max_iid	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	9678.900000	18.247226	22.517854	32.353238	31.479976	9630	9744
avrg_iid	0.967890	0.001825	0.002252	0.003235	0.003148	0.963000	0.974400

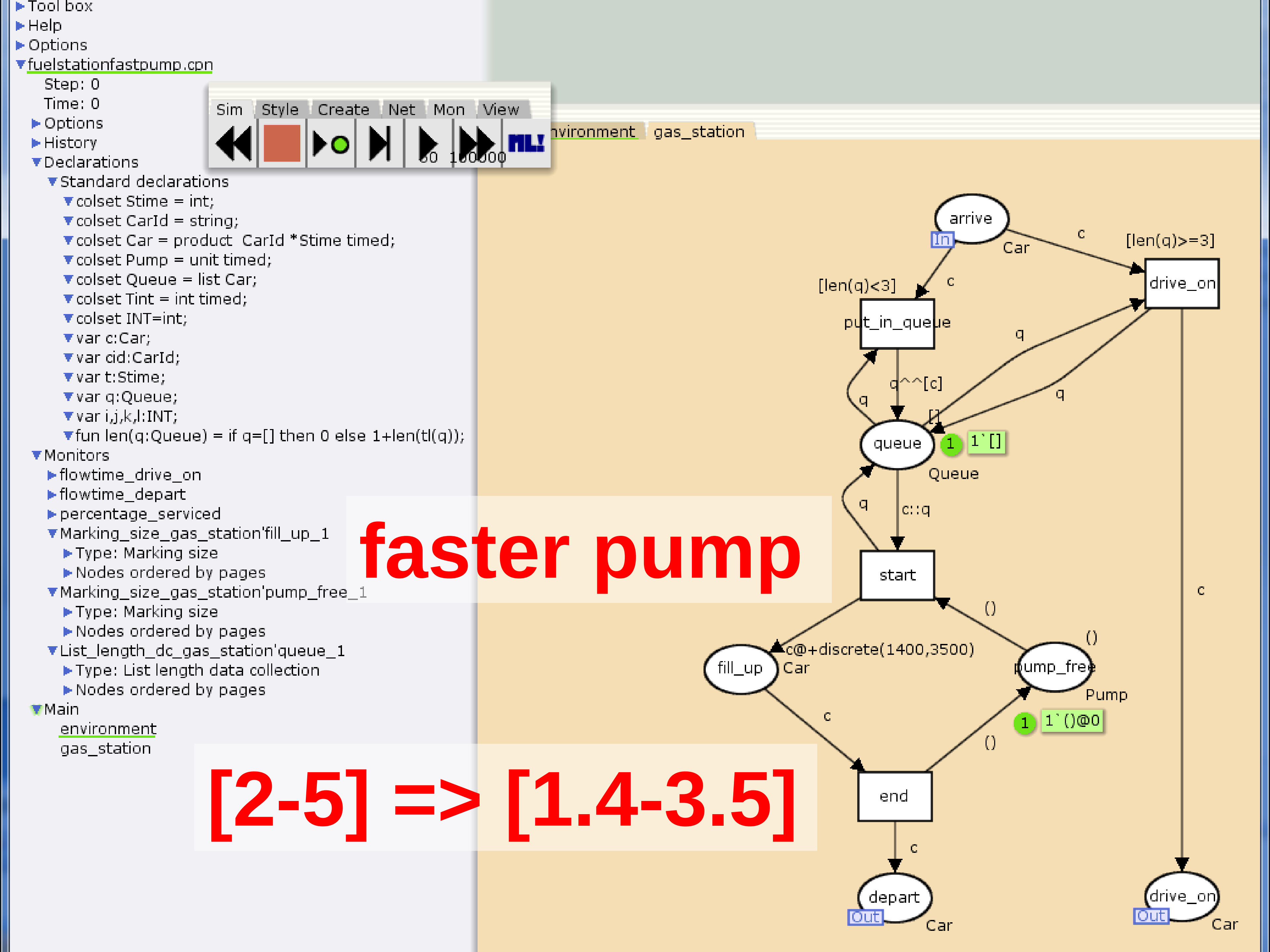
Average flow time [10.518,10.774]

Average fraction served [0.966,0.970]

Generated: Sun Mar 28 19:51:13 2010

Results

	Average flow time	Average fraction served
Base case	[7.352,7.460]	[0.910,0.915]
Two pumps	[3.916,3.944]	[0.996,0.997]
Six places	[10.518,10.774]	[0.966,0.970]



faster pump

[2-5] => [1.4-3.5]

CPN Tools Performance ... | CPN Tools Performance ... | CPN Tools Performance ...

file:///D:/courses/BIS-2010/CPN/newsimulation/output/reps_4/PerfReportIID.html

CPN Tools Performance Report
Net: D:\courses\BIS-2010\CPN\newsimulation\fuelstationfastpump.cpn
Number of replications: 10

Statistics

Name	Avrg	90% Half Length	95% Half Length	99% Half Length	StD	Min	Max
List_length_dc_gas_station'queue_1							
count_iid	19779.800000	15.691641	19.364153	27.822059	27.071100	19731	19827
max_iid	3.000000	0.000000	0.000000	0.000000	0.000000	3	[illegible]
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.388141	0.010423	0.012863	0.018481	0.017982	0.358039	0.419027
Marking_size_gas_station'fill_up_1							
count_iid	19557.600000	31.383282	38.728306	55.644117	54.142200	19460	19652
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.600104	0.004608	0.005687	0.008170	0.007950	0.586972	0.612357
Marking_size_gas_station'pump_free_1							
count_iid	19557.600000	31.383282	38.728306	55.644117	54.142200	19460	19652
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	[illegible]	[illegible]
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.399896	0.004608	0.005687	0.008170	0.007950	0.387643	0.413028

Average queue length [0.378,0.399]

Average # pumps busy [0.595,0.605]

Average # pumps free [0.395,0.405]

CPN Tools Performance ... | CPN Tools Performance ... | CPN Tools Performance ...

file:///D:/courses/BIS-2010/CPN/newsimulation/output/reps_4/PerfReportIID.html

Average flow time [4.000,4.065]

Average fraction served [0.976,0.979]

avrg_iid	0.399896	0.004608	0.005687	0.008170	0.007950	0.387643	0.413028
flowtime_depart							
count_iid	9777.800000	15.691641	19.364153	27.822059	27.071100	9729	9825
max_iid	12359.400000	177.506636	219.050743	314.728079	306.233099	11721	12731
min_iid	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
sum_iid	39430335.100000	272477.110796	336248.349493	483115.444674	470075.439377	38765863	40232718
avrg_iid	4032.742007	32.360712	39.934495	57.377149	55.828454	3945.634911	4135.339500
flowtime_drive_on							
count_iid	222.200000	15.691641	19.364153	27.822059	27.071100	175	271
max_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
percentage_serviced							
count_iid	10000.000000	0.000000	0.000000	0.000000	0.000000	10000	10000
max_iid	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	1	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	9777.800000	15.691641	19.364153	27.822059	27.071100	9729	9825
avrg_iid	0.977780	0.001569	0.001936	0.002782	0.002707	0.972900	0.982500

Generated: Sun Mar 28 20:03:31 2010

Results

	Average flow time	Average fraction served
Base case	[7.352,7.460]	**[0.910,0.915]**
Two pumps	**[3.916,3.944]**	[0.996,0.997]
Six places	**[10.518,10.774]**	[0.966,0.970]
Faster pump	**[4.000,4.065]**	**[0.976,0.979]**

Insights obtained from simulation

- **Adding a pump significantly reduces the flow time (from approx. 7.4 to approx. 3.9 minutes) and reduces the percentage not served (from approx. 9% to approx. 1%).**
- **Adding more waiting places significantly increases the flow time (from approx. 7.4 to approx. 10.6 minutes) but reduces the percentage not served (approx. 9% to approx. 3%).**
- **Installing a faster pump significantly reduces the flow time (from approx. 7.4 to approx. 4.0 minutes) and reduces the percentage not served (from approx. 9% to approx. 3%).**

Analytical models versus Simulation models

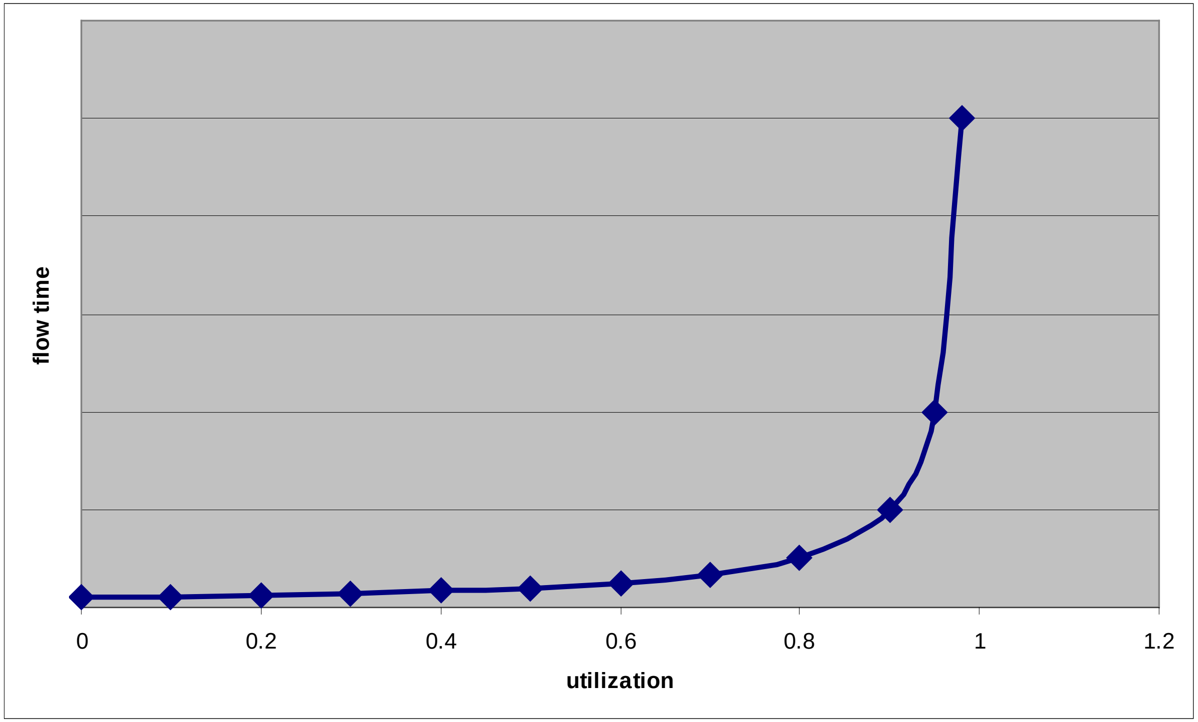
Example: M/M/1 queue

- arrival rate λ (average interarrival time = $1/\lambda$)
- service rate μ (average interarrival time = $1/\mu$)
- utilization $\rho = \lambda/\mu$
- average nof cases in system $L = \rho/(1 - \rho)$
- average flow time $S = 1/(\mu-\lambda)$
- Example:
  - $\lambda = 1/100$ and $\mu = 1/50$
  - $\rho = 0.5$
  - $L = 1$
  - $S = 100$

M/M/1 queue

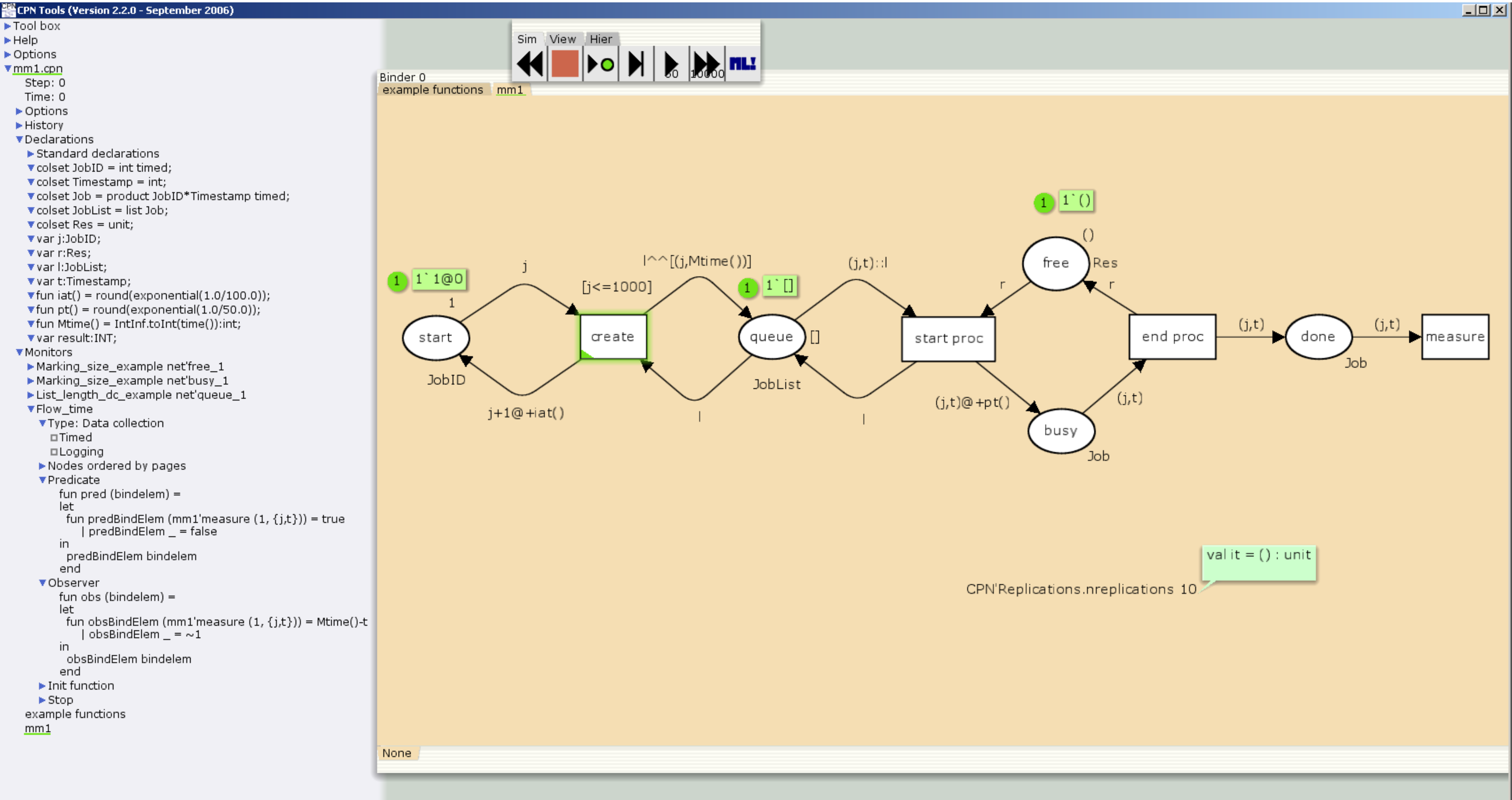
- λ = 1/100
- μ = 1/50
- ρ = 0.5
- L = 1
- S = 100

- λ = 1/100
- μ = 1/80
- ρ = 0.8
- L = 4
- S = 400

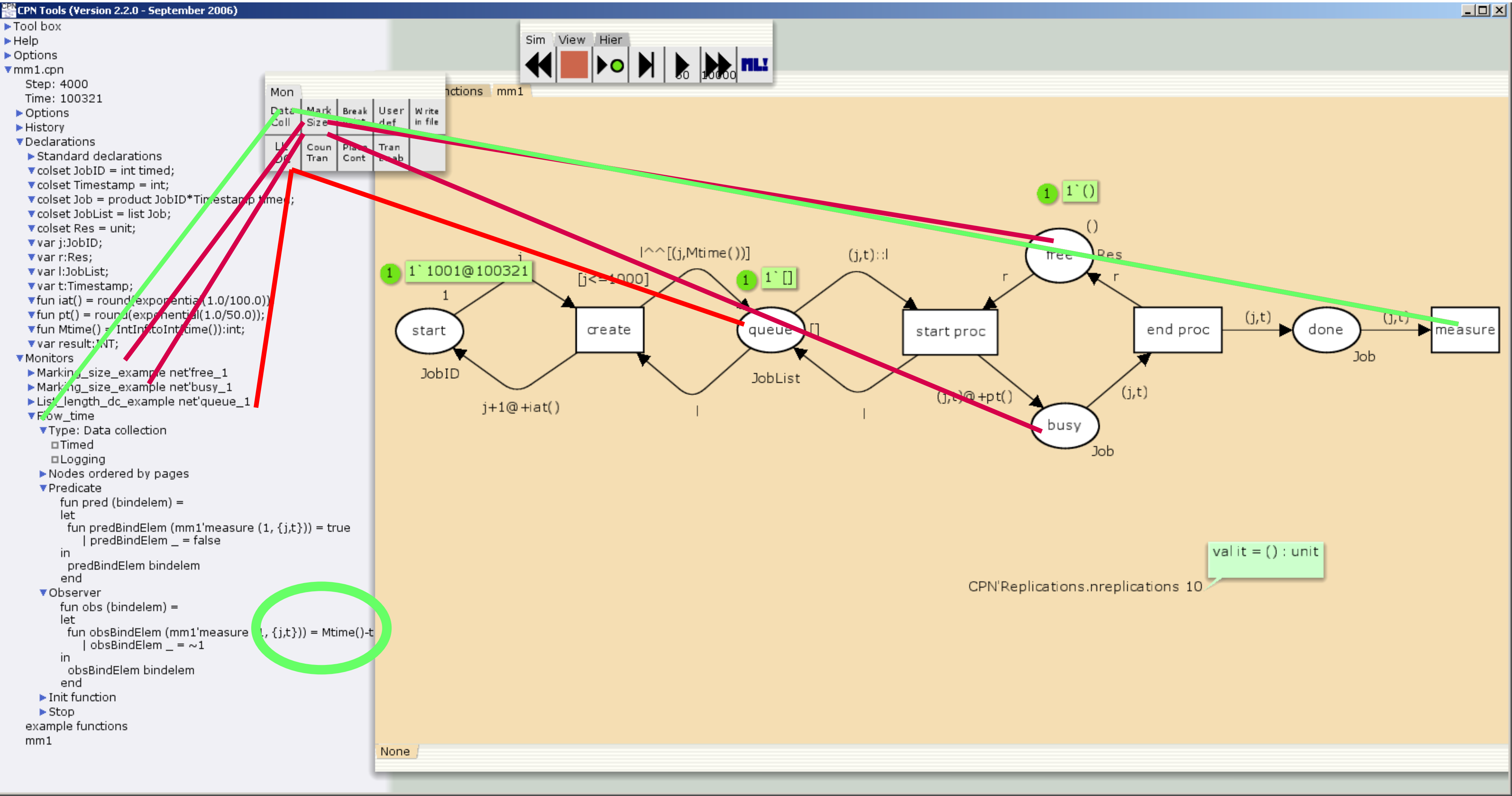


- λ = 1/100
- μ = 1/99
- ρ = 0.99
- L = 99
- S = 9900

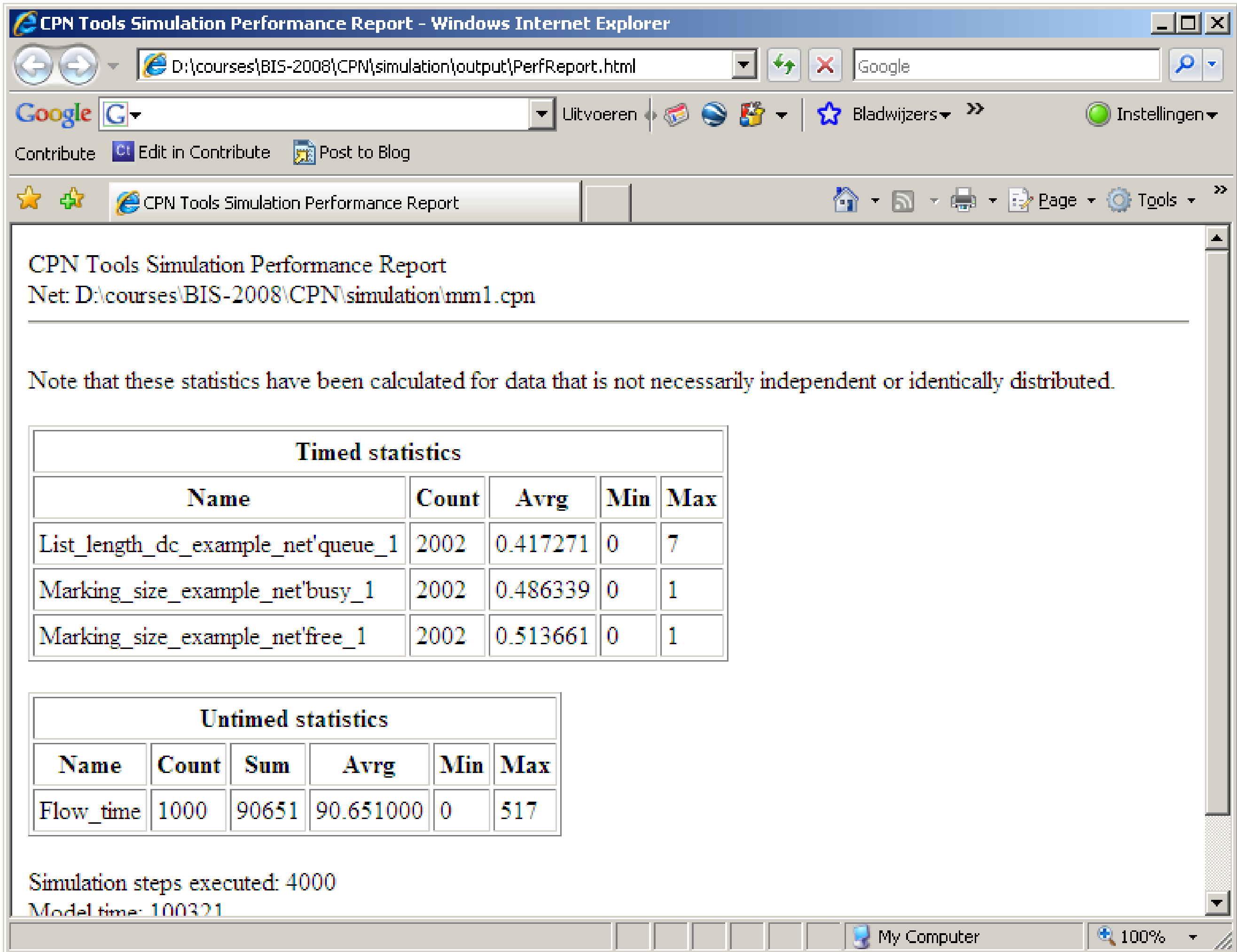
CPN model with monitors



Creating monitors



Single run



CPN Tools Simulation Performance Report
Net: D:\courses\BIS-2008\CPN\simulation\mm1.cpn

Note that these statistics have been calculated for data that is not necessarily independent or identically distributed.

Timed statistics				
Name	Count	Avrg	Min	Max
List_length_dc_example_net'queue_1	2002	0.417271	0	7
Marking_size_example_net'busy_1	2002	0.486339	0	1
Marking_size_example_net'free_1	2002	0.513661	0	1

Untimed statistics					
Name	Count	Sum	Avrg	Min	Max
Flow_time	1000	90651	90.651000	0	517

Simulation steps executed: 4000
Model time: 100321

CPN Tools Performance Report - Windows Internet Explorer

D:\courses\BIS-2008\CPN\simulation\output\reps_2\PerfReportIID.html

CPN'Replications.nreplications 10

90% [96.703-4.01059,96703+4.0159]

CPN Tools Performance Rep[illegible]
Net: D:\courses\BIS-2008\CP[illegible]
Number of replications: 10

Statistics							
Name	**Avrg**	**90% Half Length**	**95% Half Length**	**99% Half Length**	**StD**	**Min**	**Max**
Flow_time							
count_iid	1000.000000	0.00000[illegible]	0.000000	0.000000	0.000000	1000	1000
max_iid	587.300000	68.[illegible]02365	84.534834	121.458094	118.179760	440	780
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
sum_iid	96703.90[illegible]00	4010.589995	4949.238717	7110.975169	6919.039359	86441	10756[illegible]
avrg_iid	96.703900	4.010590	4.949239	7.110975	6.919039	86.4[illegible]	107.564000
List_length_dc_example_net'queue_1							
count_iid	2002.000000	0.000000	0.000000	0.00[illegible]	0.00[illegible]00	2002	2002
max_iid	8.400000	0.828801	1.0227[illegible]	1.469505	1.429841	6	10
min_iid	0.000000	0.000000	0.000000	0.000[illegible]	[illegible]000000	0	0
avrg_iid	0.464261	[illegible].031128	0.038413	[illegible].05519[illegible]	0.053702	0.391320	0.559613
Marking_size_example_net'busy_1							
count_iid	2002.000000	0.000000	[illegible].000000	0.000000	0.000000	2002	2002
max_iid	1.000000	0.000000	[illegible].000000	0.000000	0.000000	1	1
min_iid	0.000000	[illegible]	0.000000	0.000000	0.000000	0	0
avrg_iid	0.484061	0.013468	0.016620	0.023880	0.023235	0.445626	0.519824
Marking_size_example_net'free_1							
count_iid	2002.000000	0.000000	0.000000	0.000000	0.000000	2002	2002
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.515939	0.013468	0.016620	0.023880	0.023235	0.480176	0.554374

- $\lambda = 1/100$
- $\mu = 1/50$
- $\rho = 0.5$
- $L = 1$
- $S = 100$

CPN'Replications.nreplications 10

CPN Tools Performance Report
Net: D:\courses\BIS-2008\CPN\simulation\mm1.cpn
Number of replications: 10

Statistics							
Name	**Avrg**	**90% Half Length**	**95% Half Length**	**99% Half Length**	**StD**	**Min**	**Max**
Flow_time							
count_iid	1000.000000	0.000000	0.000000	0.000000	0.000000	1000	1000
max_iid	1845.200000	320.566238	395.592378	568.379854	553.038435	1211	3088
min_iid	0.400000	0.299328	0.369383	0.530723	0.516398	0	1
sum_iid	424854.700000	47563.021217	58694.792140	84331.597902	82055.362490	293109	540346
avrg_iid	424.854700	47.563021	58.694792	84.331598	82.055362	293.109000	540.346000
List_length_dc_example_net'queue_1							
count_iid	2002.000000	0.000000	0.000000	0.000000	0.000000	2002	2002
max_iid	19.800000	3.040301	3.751860	5.390604	5.245104	15	33
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	3.526256	0.491455	0.606477	0.871374	0.847855	2.214570	4.693034
Marking_size_example_net'busy_1							
count_iid	2002.000000	0.000000	0.000000	0.000000	0.000000	2002	2002
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.821571	0.014726	0.018172	0.026109	0.025405	0.792455	0.867871
Marking_size_example_net'free_1							
count_iid	2002.000000	0.000000	0.000000	0.000000	0.000000	2002	2002
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.178429	0.014726	0.018172	0.026109	0.025405	0.132129	0.207545

- $\lambda = 1/100$
- $\mu = 1/80$
- $\rho = 0.8$
- $L = 4$
- $S = 400$

CPN'Replications.nreplications 10

CPN Tools Performance Report - Windows Internet Explorer

D:\courses\BIS-2008\CPN\simulation\output\reps_6\PerfReportIID.html

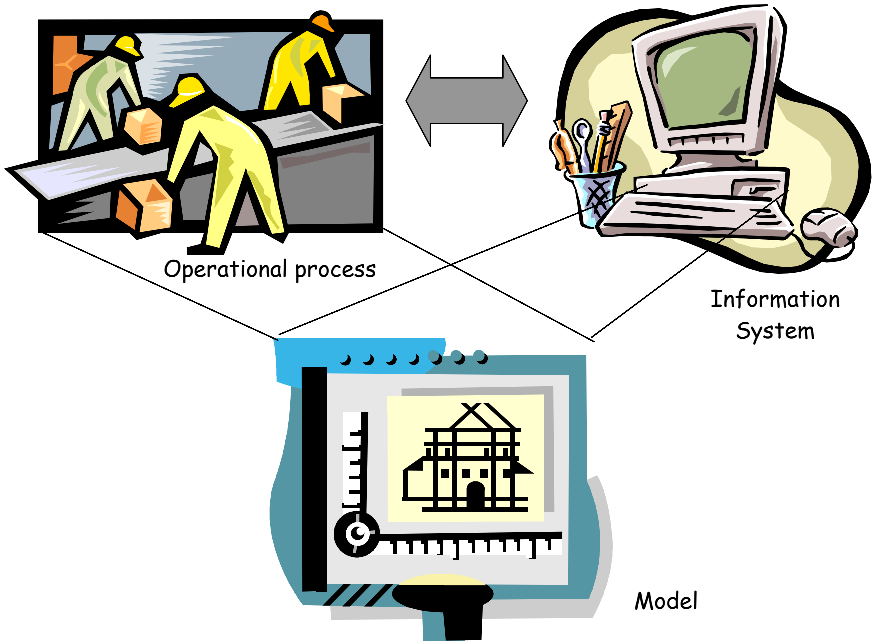
Statistics							
Name	**Avrg**	**90% Half Length**	**95% Half Length**	**99% Half Length**	**StD**	**Min**	**Max**
Flow_time							
count_iid	1000.000000	0.000000	0.000000	0.000000	0.000000	1000	1000
max_iid	4899.700000	1252.193365	1545.259897	2220.201002	2160.274471	2216	9389
min_iid	3.300000	2.186834	2.698646	3.877365	3.772709	1	13
sum_iid	1709043.500000	507550.790075	626339.272858	899912.748359	875622.763150	958796	3460629
avrg_iid	1709.043500	507.550790	626.339273	899.912748	875.622763	958.796000	3460.629000
List_length_dc_example_net'queue_1							
count_iid	2002.000000	0.000000	0.000000	0.000000	0.000000	2002	2002
max_iid	46.800000	11.543858	14.245612	20.467834	19.915377	26	92
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	15.676766	4.935593	6.090732	8.751052	8.514848	8.628088	33.015949
Marking_size_example_net'busy_1							
count_iid	2002.000000	0.000000	0.000000	0.000000	0.000000	2002	2002
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.954594	0.014241	0.017574	0.025250	0.024569	0.928707	0.995219
Marking_size_example_net'free_1							
count_iid	2002.000000	0.000000	0.000000	0.000000	0.000000	2002	2002
max_iid	1.000000	0.000000	0.000000	0.000000	0.000000	1	1
min_iid	0.000000	0.000000	0.000000	0.000000	0.000000	0	0
avrg_iid	0.045406	0.014241	0.017574	0.025250	0.024569	0.004781	0.071293

Generated: Mon Oct 27 22:38:41 2008

- $\lambda = 1/100$
- $\mu = 1/99$
- $\rho = 0.99$
- $L = 99$
- $S = 9900$

Note deviations. Why?

Conclusion analysis



- Analysis is typically model-driven to allow e.g. what-if questions.
- Models of both operational processes and/or the information systems can be analyzed.
- Types of analysis:
  - *validation (interactive simulation/gaming)*
  - *verification (state-space analysis, place and transition invariants, siphons, traps, etc.)*
  - *performance analysis (simulation)*