Real-Time Stream Monitoring
by Maximilian Schwenger
book review

>500 reviewed
Pyhton?

⭐⭐⭐⭐⭐ book review

>500 reviewed
Lola!

book review

>500 reviewed
book review

>500 reviewed
Lola

in stars, pid

out isBook :=
    if pid = then 1 else 0

out #books :=
    #books[-1,0] + isBook[0,0]

trigger #books >= 5
trigger badRev

>500 reviewed
in stars, pid

out isBook :=
  if pid = then 1 else 0

out #books :=
  #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev

>500 reviewed
Lola

in stars, pid

out isBook :=
  if pid = 🌟🌟 then 1 else 0

out #books :=
  #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
in stars, pid

out isBook :=
    if pid =    then 1 else 0

out #books :=
    #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
in stars, pid

out isBook :=
    if pid = 1 then 1 else 0

out #books :=
    #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
in stars, pid

out isBook :=
  if pid = 1 then 1 else 0

out #books :=
  #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
in stars, pid

out isBook :=
  if pid = 🌟🌟🌟🌟🌟 then 1 else 0

out #books :=
  #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
in stars, pid

\[\text{out isBook := if pid = } \text{ then 1 else 0}\]

\[\text{out #books := #books[-1,0]+isBook[0,0]}\]

\[\text{trigger #books >= 5}\]
\[\text{trigger badRev}\]
in stars, pid

out isBook :=
  if pid = 📖 then 1 else 0

out #books :=
  #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
Lola

in stars, pid

out isBook :=
    if pid = 📚 then 1 else 0

out #books :=
    #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
in stars, pid

out isBook :=
if pid = 📚 then 1 else 0

out #books :=
#books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
in stars, pid

out isBook :=
  if pid = then 1 else 0

out #books :=
  #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
Lola

in stars, pid

out isBook :=
   if pid = 
   then 1
   else 0

out #books :=
   #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
in stars, pid

out isBook :=
    if pid = 📚 then 1 else 0

out #books :=
    #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
in stars, pid

out isBook :=
  if pid = 📖 then 1 else 0

out #books :=
  #books[-1,0]+isBook[0,0]

trigger #books >= 5
trigger badRev
in stars, pid

out isBook :=
    if pid = 
    then 1 else 0

out #books :=
    #books[-1,0]+isBook[0,0]

trigger #books >= 5
in stars, pid

out isBook :=
    if pid =  📚 then 1 else 0

out #books :=
    #books[-1,0] + isBook[0,0]

out badRev :=
    isBook[0,0] = 1
    and stars = 0

trigger #books >= 5
trigger badRev
Lola 2.0

in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)[−1, 0] = 3
  and stars = 4

trigger any(inc3to4)
in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)\[-1, 0\] = 3
  and stars = 4

trigger any(inc3to4)
in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)[-1, 0] = 3
  and stars = 4

trigger any(inc3to4)
in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)[−1, 0] = 3
  and stars = 4

trigger any (inc3to4)
**Lola 2.0**

in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)[−1, 0] = 3
  and stars = 4

trigger any(inc3to4)
Lola 2.0

in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)[-1, 0] = 3
      and stars = 4

trigger any(inc3to4)
in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)[{-1, 0}] = 3
  and stars = 4

trigger any/inc3to4
in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)[{-1, 0}] = 3
  and stars = 4

trigger any(any3to4)
in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)[-1, 0] = 3
  and stars = 4

trigger any(inc3to4)
in stars, pid

out split \langle \text{prod} \rangle 
    inv: pid 
    ext: pid = prod 
    := stars

out inc3to4 \langle \text{prod} \rangle 
    inv: pid 
    ext: pid = prod 
    := \text{split}(\text{prod})[-1, 0] = 3 
    and stars = 4

trigger any(inc3to4)
**Lola 2.0**

in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)[-1, 0] = 3
  and stars = 4

trigger any(inc3to4)
in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)[-1, 0] = 3
  and stars = 4

trigger any(inc3to4)
in stars, pid

out split <prod>
  inv: pid
  ext: pid = prod
  := stars

out inc3to4 <prod>
  inv: pid
  ext: pid = prod
  := split(prod)[−1, 0] = 3
  and stars = 4

trigger any(inc3to4)
Lola 2.0

\[ \text{in stars, pid} \]

\[ \text{out split } <\text{prod}> \]
\[ \text{inv: pid} \]
\[ \text{ext: pid} = \text{prod} \]
\[ := \text{stars} \]

\[ \text{out inc3to4 } <\text{prod}> \]
\[ \text{inv: pid} \]
\[ \text{ext: pid} = \text{prod} \]
\[ := \text{split(prod)}[-1, 0] = 3 \]
\[ \text{and stars} = 4 \]

\[ \text{trigger any(inc3to4)} \]
Real-time Stream Monitoring

Peter Faymonville, Bernd Finkbeiner, Maximilian Schwenger, and Hazem Torfah

Saarland University
{faymonville, finkbeiner, schwenger, torfah}@react.uni-saarland.de

Abstract. We introduce a new stream-based specification language for the description of real-time properties of reactive systems. The key feature
Real-Time Lola 2.0

Rapid increase in page views?

Average number of reviews per month?

Hourly rate of new book reviews?
Real-Time Lola 2.0

Rapid increase in page views?

Average number of reviews per month?

Hourly rate of new book reviews?
Real-Time Lola 2.0
Real-Time Lola 2.0
Real-Time Lola 2.0
Real-Time Lola 2.0
Real-Time Lola 2.0

10s
Aggregation $\gamma$
Aggregation $\gamma$
Aggregation $\gamma$
Aggregation $\gamma$

(\begin{array}{cccccc}
\circ & \circ & \circ & \circ & \circ & \circ \\
\end{array})\circ
Aggregation $\gamma$
Aggregation $\gamma$

$\ast$  $\circ$  $F$

$(\circ \circ \circ \circ \circ)$

$\equiv$  $\text{Sack}$
Sound Split
Sound Split

Example: count
Sound Split

Example: count
Sound Split

Example: count
Sound Split

Example: count
Sound Split

count

\[ x \mapsto 1 + \text{id} \]
<table>
<thead>
<tr>
<th>Operation</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>count</td>
<td>$x \mapsto 1$ + id</td>
</tr>
<tr>
<td>average</td>
<td>$x \mapsto (x,1)$ + $(a,b) \mapsto a/b$</td>
</tr>
<tr>
<td>Operation</td>
<td>x ↦ (x, 1)</td>
</tr>
<tr>
<td>-------------</td>
<td>------------</td>
</tr>
<tr>
<td>count</td>
<td>x ↦ 1</td>
</tr>
<tr>
<td>average</td>
<td></td>
</tr>
<tr>
<td>median</td>
<td></td>
</tr>
<tr>
<td>Function</td>
<td>Formula</td>
</tr>
<tr>
<td>---------------</td>
<td>----------------------------------------------</td>
</tr>
<tr>
<td><strong>count</strong></td>
<td>$x \mapsto 1$</td>
</tr>
<tr>
<td><strong>average</strong></td>
<td>$x \mapsto (x,1)$</td>
</tr>
<tr>
<td><strong>median</strong></td>
<td>impossible</td>
</tr>
<tr>
<td><strong>“integral”</strong></td>
<td>trapezoid</td>
</tr>
</tbody>
</table>
Real-Time Lola 2.0
Real-Time Lola 2.0
Real-Time Lola 2.0
Aggregation $\gamma$

$\ast \ O \ F$

\[ o(\ldots) \]

\[ = \]
Aggregation $\gamma$

\[ F(\circ(\bigcirc \bigcirc \bigcirc \bigcirc \bigcirc )) = \text{bag} \]
Left-Invertibility of $\circ$

$$a \circ b = c \iff \circ^{-1}(c,a) = b$$
Left-Invertibility of $\circ$

\[ a \circ b = c \iff \circ^{-1}(c,a) = b \]

\[ 3 + 4 = 7 \iff +^{-1}(7,3) = 4 \]
Left-Invertibility of $\circ$

$$a \circ b = c \iff \circ^{-1}(c,a) = b$$

$$3 + 4 = 7 \iff +^{-1}(7,3) = 4$$

$$\max(12,9) = 12 \iff \max^{-1}(12,12) = ?$$
time

t_0 t_0 + x t_0 + 2x t_0 + 3x t_0 + 4x t_0 + 5x t_0 + 6x t_0 + 7x t_0 + 8x t_0 + 9x t_0 + 10x
10s
\[ \text{time} \]

\[
\begin{array}{cccccccccccc}
0 & 0 & 1 & 2 & 2 & 1 & 3 & 9 & 4 & 6 & 5 & 2 \\
\bot & \bot & \bot & \bot & \bot & \bot & \top & \bot & \bot & \bot & \bot & \bot \\
0 & 0 & 1 & 2 & 2 & 3 & 3 & 3 & 4 & 5 & 5 & 5
\end{array}
\]
Example: Amazon
Example: Amazon

Does the star average of any product increase significantly?
in string pid
in double stars
time in in int timestamp

out double star_avg <string prod>
inv: pid
ter: low_freq
:= arr_freq(prod)[0, 0] < 20

out int time_p <string prod>
inv: pid
ext: pid = prod
ter: low_freq
:= timestamp

out int arr_freq <string prod>
inv: pid
ext: pid = prod
ter: low_freq
:= time_p(prod)
    [slide, 1day, cnt, 0]

out bool low_freq <string prod>
inv: pid
ter: low_freq
:= arr_freq(prod)[0, 0] < 20

out double star_avg_d <string prod>
inv: pid
ext: pid = prod
ter: low_freq
:= stars[slide, 1day, avg, 0.0]

out double star_avg_d <string prod>
inv: pid
ext: pid = prod
ter: low_freq
:= (  
    star_avg(prod)[0, 0.0]  
    - star_avg(prod)[-1, 0.0]  
  ) / (  
    time_p(prod)[0, 0]  
    - time_p(prod)[-1, 0]  
  )

trigger any(star_avg_d > 2.5)
in double sensor
time in double timestamp
in double ref

double acc_sensor := sensor[slide, 1min, integral, 0]
double acc_ref := ref[slide, 1min, integral, 0]

double error := acc_sensor - acc_ref

trigger any(error > 5)
in double sensor
time in double timestamp
in double ref

out double acc_sensor
  := sensor[slide, 1min, integral, 0]

out double acc_ref
  := ref[slide, 1min, integral, 0]

out double error
  := acc_sensor - acc_ref

trigger any(error > 5)
Time and Space

Time: $O(freq \times window\ size \times O(\circ))$
Time and Space

Time: $O(\text{freq} \times \text{window size} \times O(\circ))$

$O(\text{freq} \times \text{window size})$  $O\left(\frac{\text{window size}}{\text{bucket size}}\right)$
Time and Space

Time: $O(freq \times window\ size \times O(\circ))$

$O(freq \times window\ size)$

$O\left(\frac{window\ size}{bucket\ size}\right)$
in double sensor
time in double timestamp
in double ref

out double acc_sensor
:= sensor[slide, 1min, integral, 0]

out double acc_ref
:= ref[slide, 1min, integral, 0]