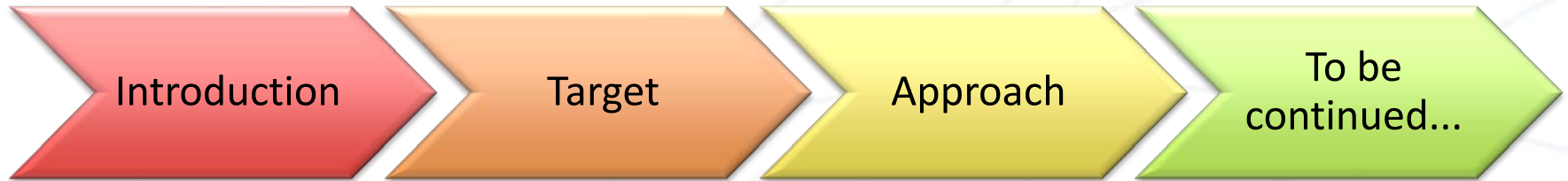


Towards Scalable Non-monotonic Stream Reasoning via Input Dependency Analysis

Thu-Le Pham¹, Alessandra Mileo², Muhammad Intizar Ali¹

1. INSIGHT, National University of Ireland, Galway
 2. INSIGHT, Dublin City University
- {thule.pham, alessandra.mileo, ali.intizar}@insight-centre.org

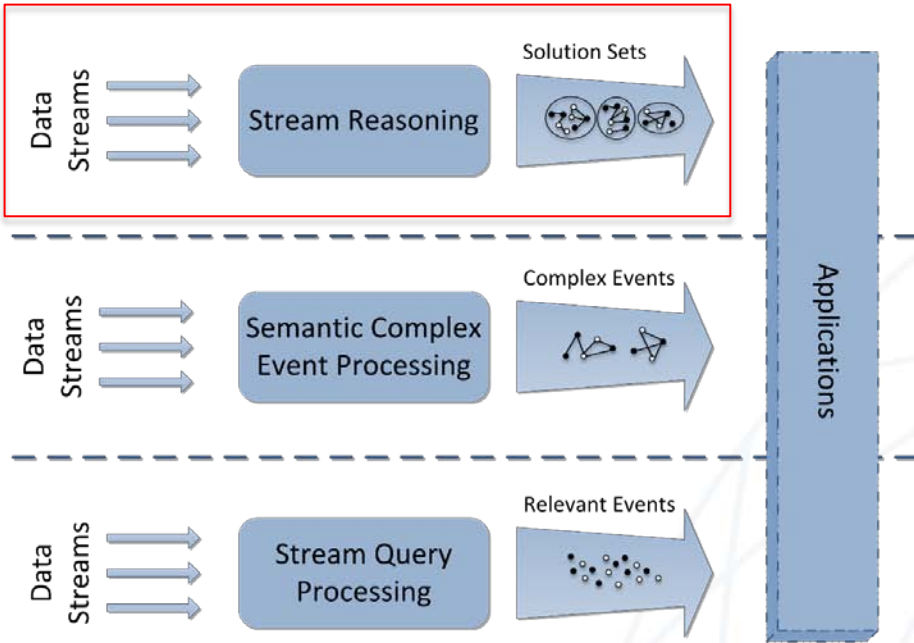
Outline



Introduction

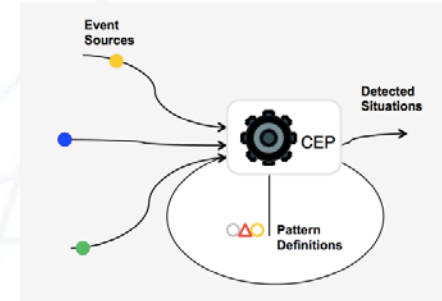
Stream reasoning

Introduction



Non-monotonic Reasoning

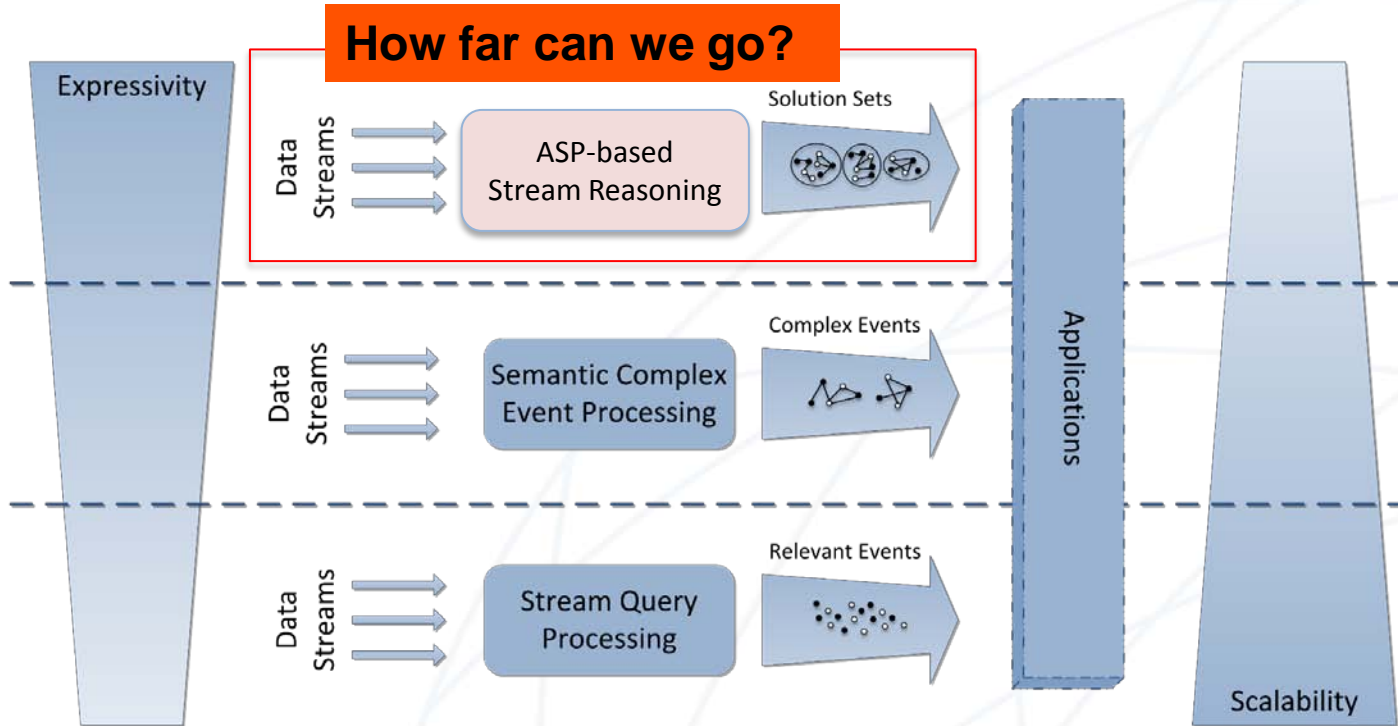
ETALIS/
EP-SPAQRL

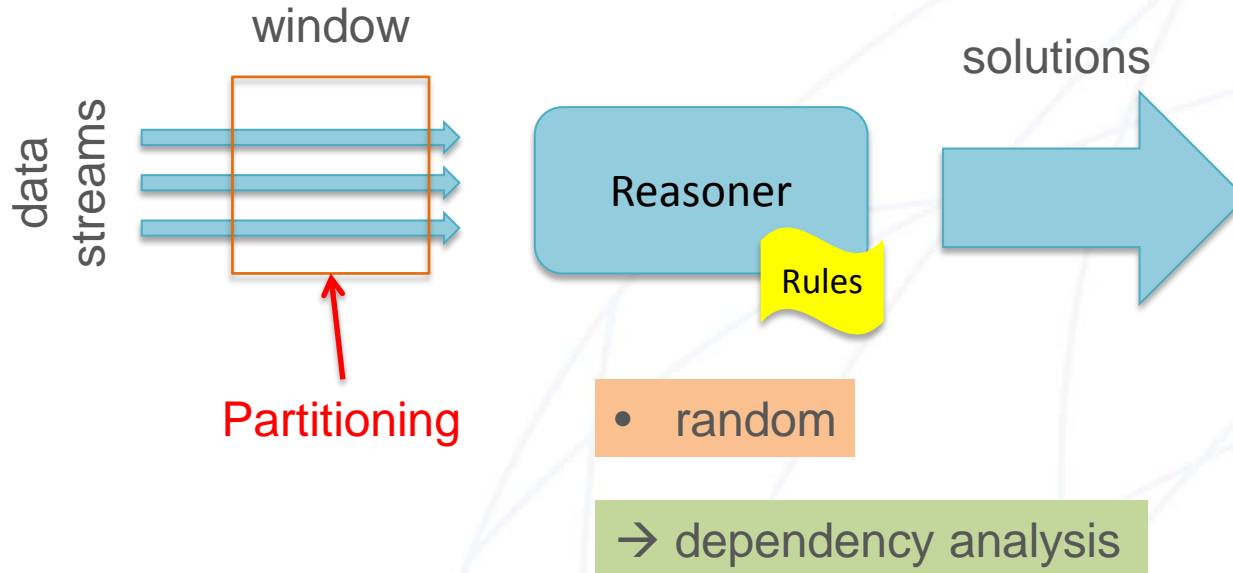


StreamRule



reprinted from <http://streamreasoning.org/events/sr4Id2015>



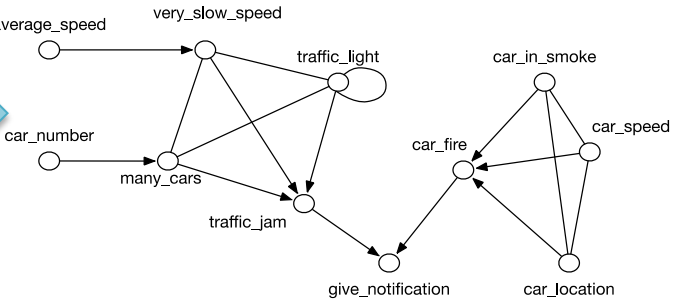




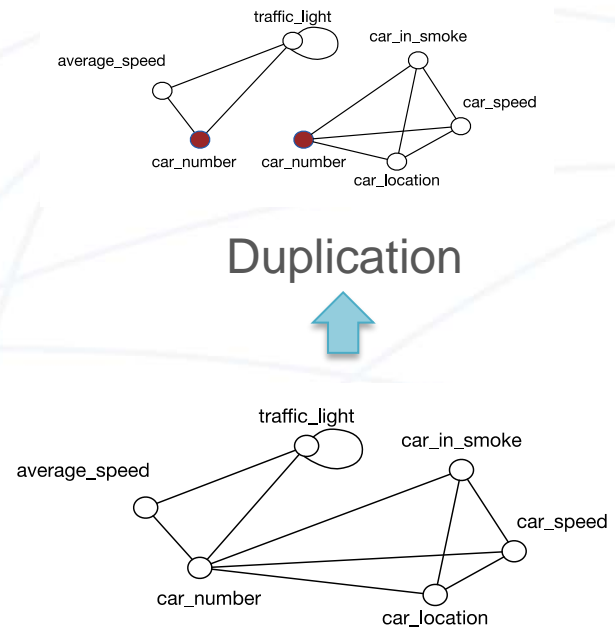
```

very_slow_speed(X) :-
average_speed(X,Y), Y < 20.
many_cars(X) :- car_number(X,Y), Y >
40.
traffic_jam(X) :- very_slow_speed(X),
many_cars(X), not traffic_light(X).
car_fire(X) :- car_in_smoke(C,high),
car_speed(C,0), car_location(C,X).
.....
    
```

Rules



Extended Dependency Graph



Duplication



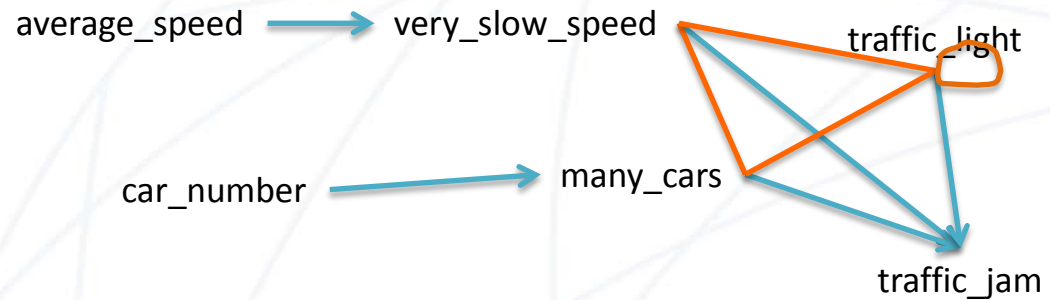
Input Dependency graph



Rules

```
(r1) very_slow_speed(X) :- average_speed(X,Y), Y < 20.  
(r2) many_cars(X) :- car_number(X,Y), Y > 40.  
(r3) traffic_jam(X) :- very_slow_speed(X), many_cars(X), not traffic_light(X).  
....
```

Extended Dependency Graph

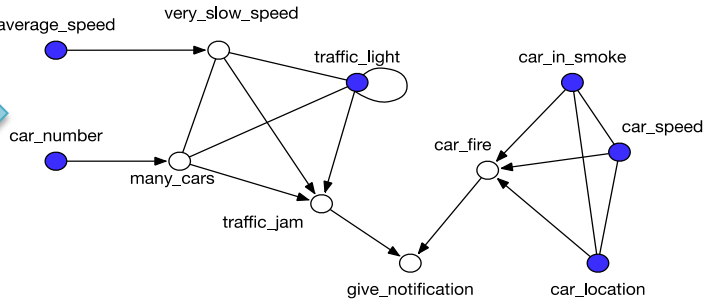




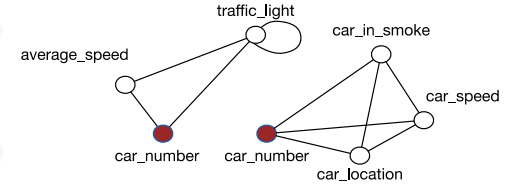
```

very_slow_speed(X) :-
average_speed(X,Y), Y < 20.
many_cars(X) :- car_number(X,Y), Y >
40.
traffic_jam(X) :- very_slow_speed(X),
many_cars(X), not traffic_light(X).
car_fire(X) :- car_in_smoke(C,high),
car_speed(C,0), car_location(C,X).
.....
    
```

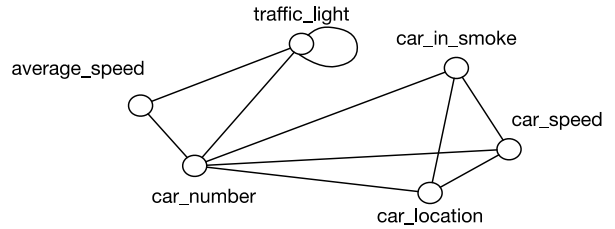
Rules



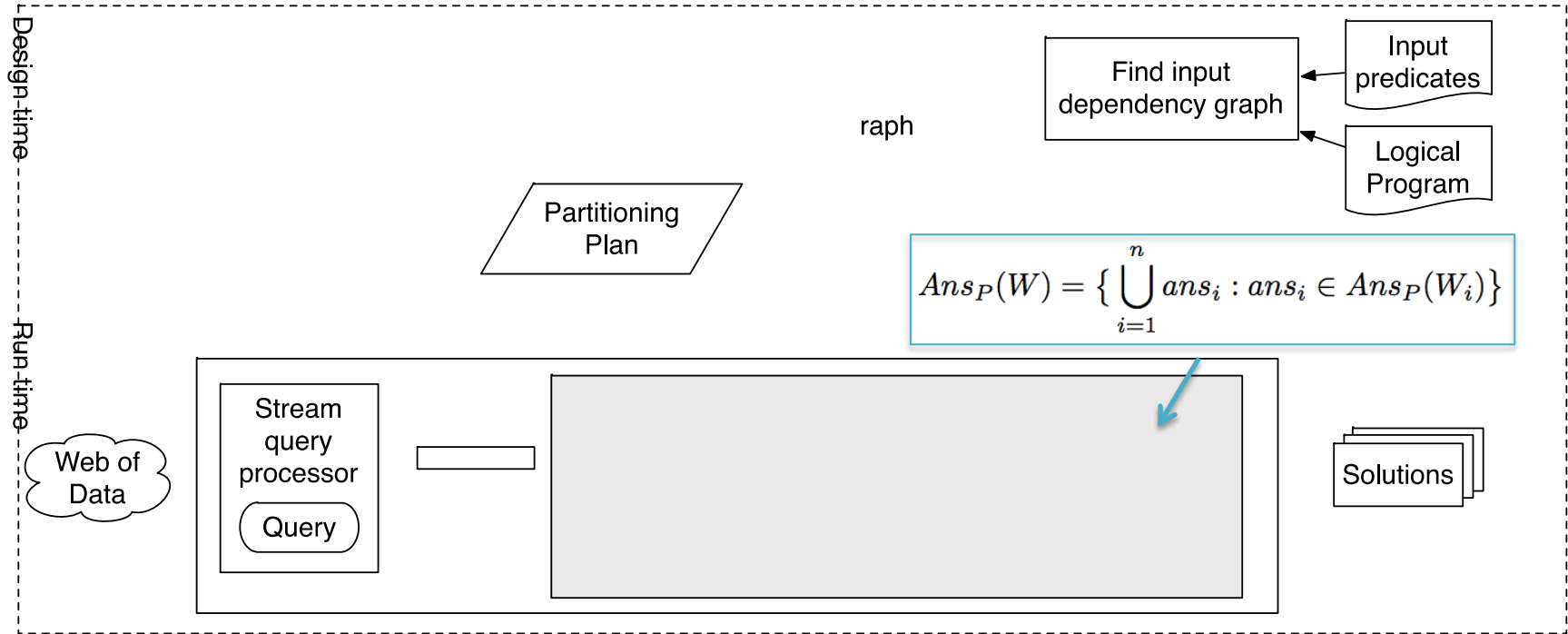
Extended Dependency Graph

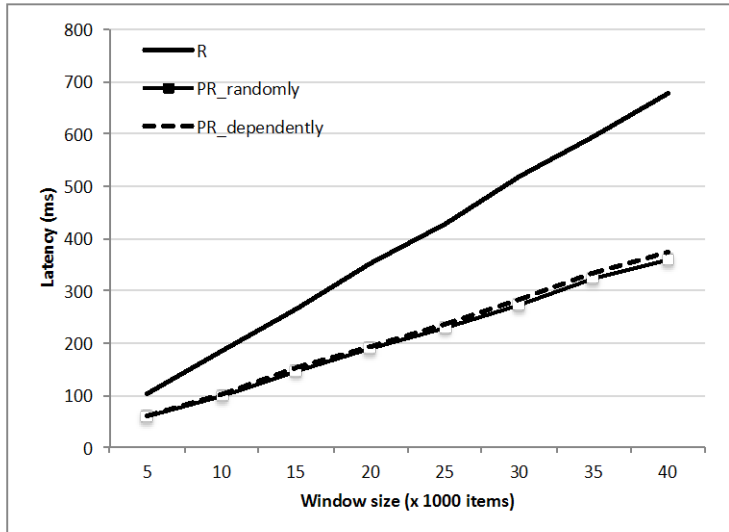


Duplication

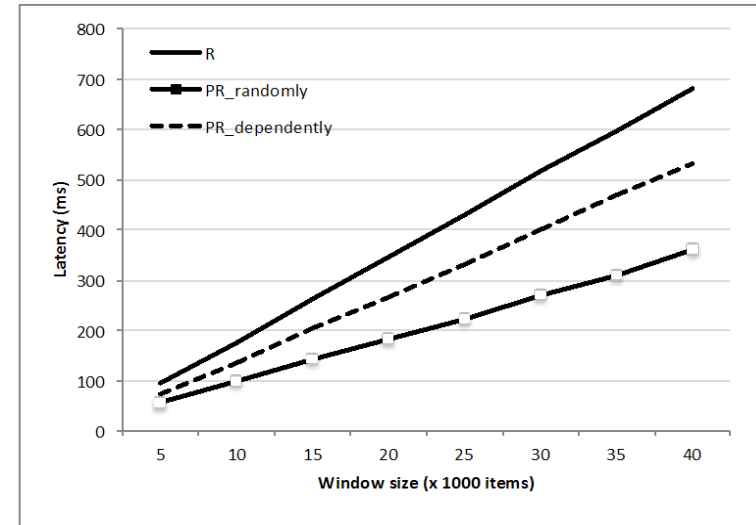


Input Dependency graph





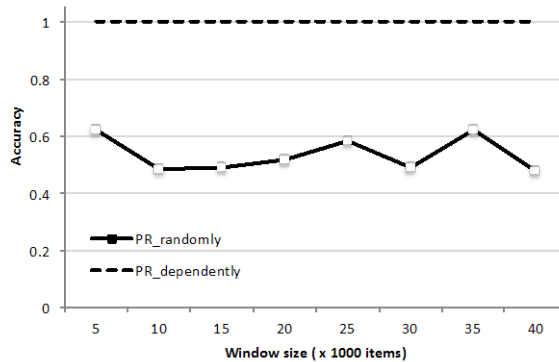
P (No duplication)



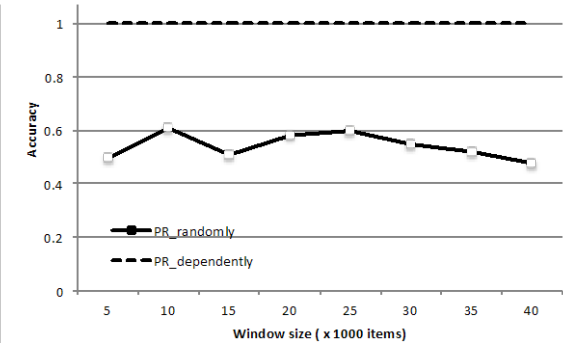
P' (Duplication)



$$\max \left(\frac{|ans_i \cap ans_j|}{|ans_j|} : ans_j \in Ans_P^R(W) \right)$$



P (No Duplication)



P' (Duplication)



◆ Conclusion

- (Extended/Input) Dependency Graph
- Duplication process
- Extended StreamRule
- working cases / not-working cases?

◆ what's next?

- benchmark
- proof

