Formal Models of Complex Event Recognition

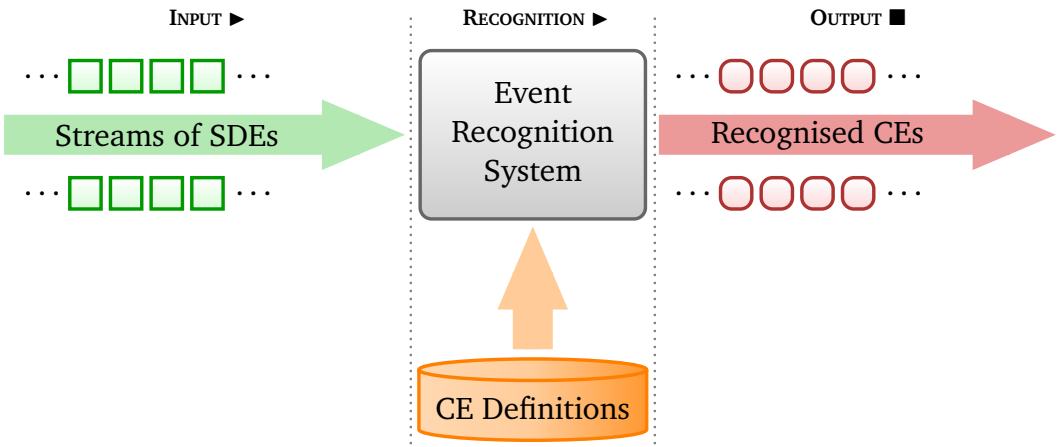
Alexander Artikis

NCSR Demokritos
University of Piraeus
Athens, Greece
a.artikis@unipi.gr

https://cer.iit.demokritos.gr

DEBS 2022

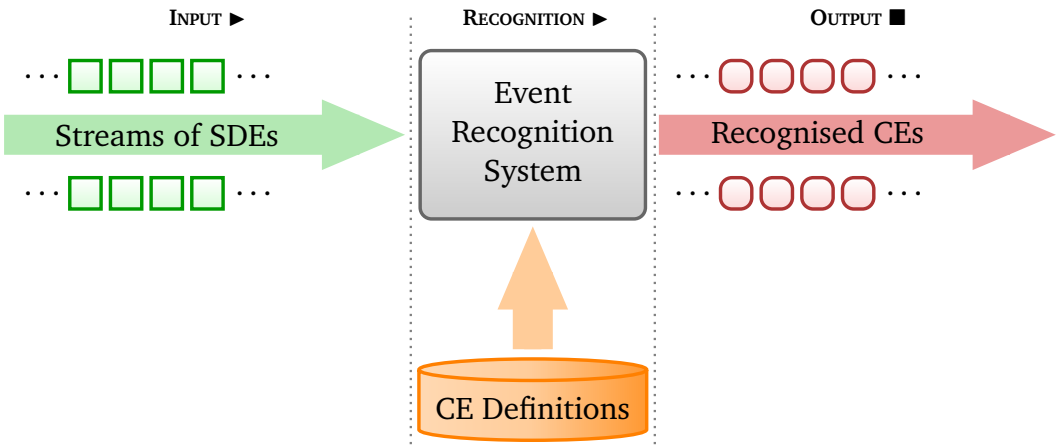
Complex Event Recognition (Event Pattern Matching)

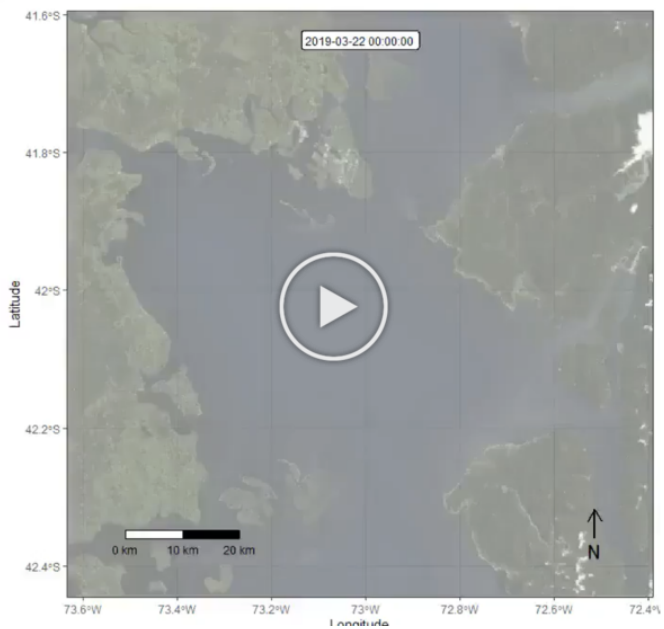


Giatrakos et al, Complex event recognition in the Big Data era: a survey, VLDB Journal, 2020.

Gugola and Margara, Processing flows of information: From data stream to complex event processing. ACM Computing Surveys, 2012.

Complex Event Recognition (Event Pattern Matching)



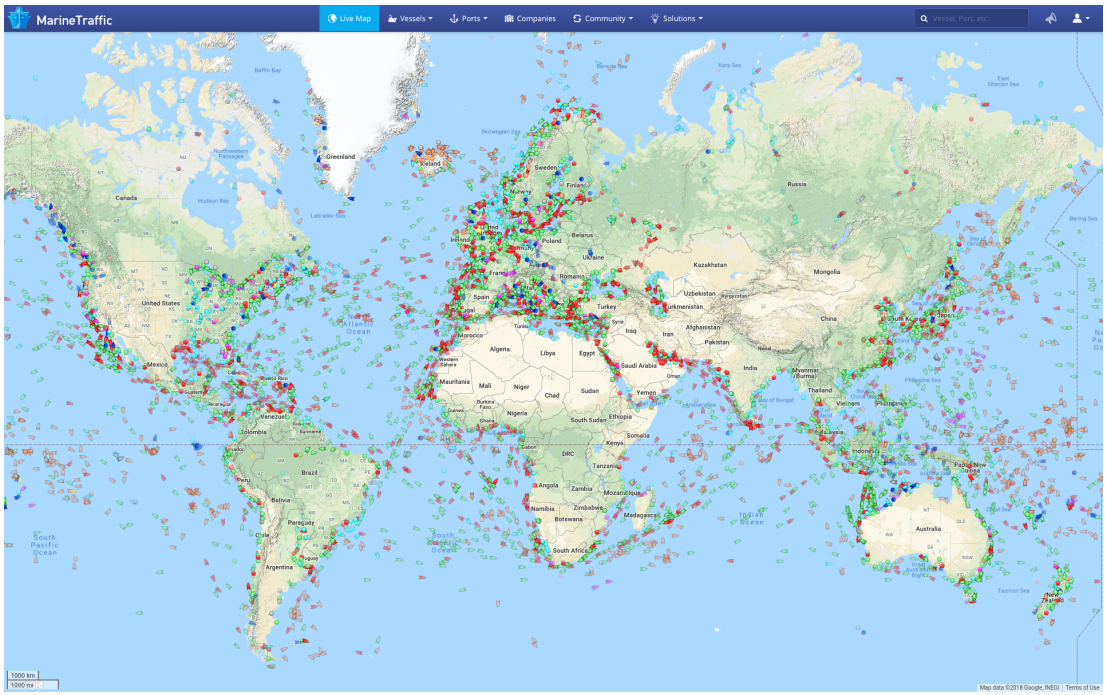


Giatrakos et al, Complex event recognition in the Big Data era: a survey, VLDB Journal, 2020.

Gugola and Margara, Processing flows of information: From data stream to complex event processing. ACM Computing Surveys, 2012.

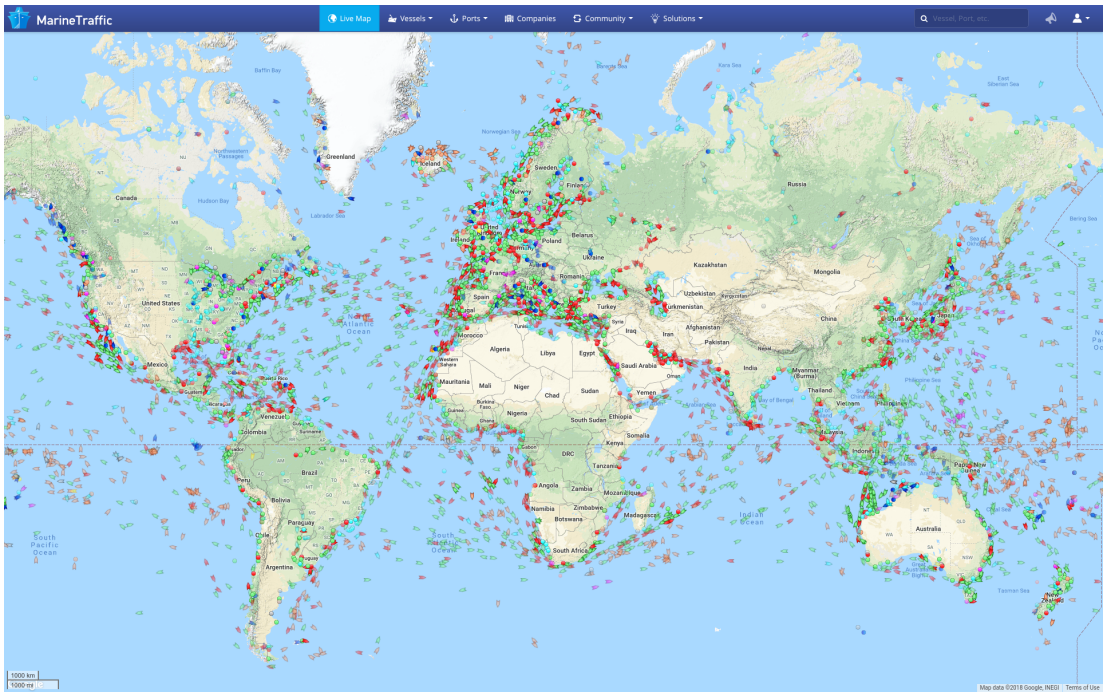
https://rdcu.be/cNkQE

Maritime Situational Awareness



http://www.marinetraffic.com

Maritime Situational Awareness



http://www.marinetraffic.com

https://cer.iit.demokritos.gr (maritime)

Data Challenges

- Velocity, Volume: 19,000,000 position signals/day at European scale.

Data Challenges

- Velocity, Volume: 19,000,000 position signals/day at European scale.
- Variety: Position signals need to be combined with other data streams
  - Weather forecasts, sea currents, etc.
- ... and static information
  - NATURA areas, shallow waters areas, coastlines, etc.

Data Challenges

- Velocity, Volume: 19,000,000 position signals/day at European scale.
- Variety: Position signals need to be combined with other data streams
  - Weather forecasts, sea currents, etc.
- ... and static information
  - NATURA areas, shallow waters areas, coastlines, etc.
- Lack of Veracity: GPS manipulation, vessels reporting false identity, communication gaps.

Data Challenges

- Velocity, Volume: 19,000,000 position signals/day at European scale.
- Variety: Position signals need to be combined with other data streams
  - Weather forecasts, sea currents, etc.
- ... and static information
  - NATURA areas, shallow waters areas, coastlines, etc.
- Lack of Veracity: GPS manipulation, vessels reporting false identity, communication gaps.
- Distribution: Vessels operating across the globe.

Many Other Applications

- Cardiac arrhythmia recognition.
- Financial fraud detection.
- Human activity recognition.
- Intrusion detection in computer networks.
- Traffic congestion recognition and forecasting in smart cities.

Requirements

- Expressive representation
  - to capture complex relationships between the events that stream into the system.

Requirements

- Expressive representation
  - to capture complex relationships between the events that stream into the system.
- Efficient reasoning
  - to support real-time decision-making in large-scale, (geographically) distributed applications.

Requirements

- Expressive representation
  - to capture complex relationships between the events that stream into the system.
- Efficient reasoning
  - to support real-time decision-making in large-scale, (geographically) distributed applications.
- Automated knowledge construction
  - to avoid the time-consuming, error-prone manual CE definition development.

Requirements

- Expressive representation
  - to capture complex relationships between the events that stream into the system.
- Efficient reasoning
  - to support real-time decision-making in large-scale, (geographically) distributed applications.
- Automated knowledge construction
  - to avoid the time-consuming, error-prone manual CE definition development.
- Reasoning under uncertainty
  - to deal with various types of noise.

Requirements

- Expressive representation
  - to capture complex relationships between the events that stream into the system.
- Efficient reasoning
  - to support real-time decision-making in large-scale, (geographically) distributed applications.
- Automated knowledge construction
  - to avoid the time-consuming, error-prone manual CE definition development.
- Reasoning under uncertainty
  - to deal with various types of noise.
- Complex event forecasting
  - to support proactive decision-making.

Issues

- The main research focus in CER has been on practical issues.
- The semantics of the CER operators is often defined indirectly, by means of examples, or by translation into evaluation models.

Issues

- The main research focus in CER has been on practical issues.
- The semantics of the CER operators is often defined indirectly, by means of examples, or by translation into evaluation models.
- Even when a formal semantics is given, this semantics is unsatisfactory because it has unintuitive behaviour (eg, sequencing is non-associative) or is restricted (eg, operators cannot be nested).
- As a result, it is not straightforward to understand and compare CER languages (and systems).

Grez et al, A Formal Framework for Complex Event Recognition. ACM Transactions on Database Systems, 2021.

Event Calculus

- A logic programming language for representing and reasoning about events and their effects.
- Key components:
  - event (typically instantaneous).
  - fluent: a property that may have different values at different points in time.

Event Calculus

- A logic programming language for representing and reasoning about events and their effects.
- Key components:
  - event (typically instantaneous).
  - fluent: a property that may have different values at different points in time.
- Built-in representation of inertia:
  - $F = V$ holds at a particular time-point if $F = V$ has been *initiated* by an event at some earlier time-point, and not *terminated* by another event in the meantime.

Kowalski and Sergot, A Logic-based Calculus of Events. New Generation Computing, 1986.

Run-Time Event Calculus (RTEC)

Predicate	Meaning
happensAt(E, T)	Event E occurs at time T
initiatedAt$(F = V, T)$	At time T a period of time for which $F = V$ is initiated
terminatedAt$(F = V, T)$	At time T a period of time for which $F = V$ is terminated
holdsFor$(F = V, I)$	I is the list of the maximal intervals for which $F = V$ holds continuously
holdsAt$(F = V, T)$	The value of fluent F is V at time T
union_all$([J_1, \ldots, J_n],\ I)$	$I = (J_1 \cup \ldots \cup J_n)$
intersect_all$([J_1, \ldots, J_n],\ I)$	$I = (J_1 \cap \ldots \cap J_n)$
relative_complement_all $(I',\ [J_1, \ldots, J_n],\ I)$	$I = I' \setminus (J_1 \cup \ldots \cup J_n)$

Artikis et al, An Event Calculus for Event Recognition. IEEE TKDE, 2015. https://github.com/aartikis/RTEC

Run-Time Event Calculus (RTEC)

Predicate	Meaning
happensAt(E, T)	Event E occurs at time T
initiatedAt$(F = V, T)$	At time T a period of time for which $F = V$ is initiated
terminatedAt$(F = V, T)$	At time T a period of time for which $F = V$ is terminated
holdsFor$(F = V, I)$	I is the list of the maximal intervals for which $F = V$ holds continuously
holdsAt$(F = V, T)$	The value of fluent F is V at time T
union_all$([J_1, \ldots, J_n],\ I)$	$I = (J_1 \cup \ldots \cup J_n)$
intersect_all$([J_1, \ldots, J_n],\ I)$	$I = (J_1 \cap \ldots \cap J_n)$
relative_complement_all $(I',\ [J_1, \ldots, J_n],\ I)$	$I = I' \setminus (J_1 \cup \ldots \cup J_n)$

Artikis et al, An Event Calculus for Event Recognition. IEEE TKDE, 2015. https://github.com/aartikis/RTEC

CE Definitions in the Run-Time Event Calculus

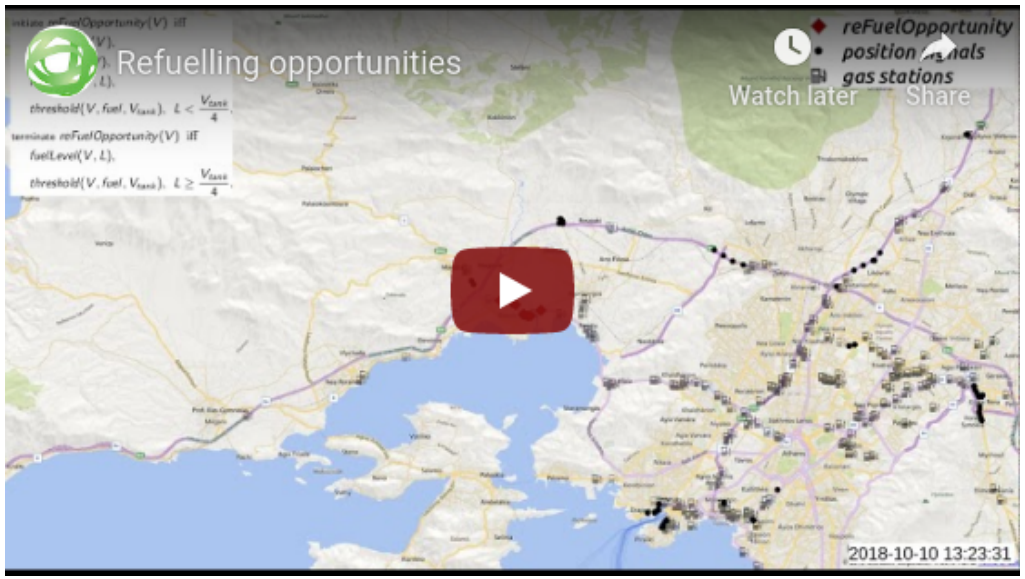
$$\textbf{initiatedAt}(CE,\ T) \leftarrow \textbf{happensAt}(E_{In_1},\ T), [\text{conditions}]$$
$$\ldots$$
$$\textbf{initiatedAt}(CE,\ T) \leftarrow \textbf{happensAt}(E_{In_i},\ T), [\text{conditions}]$$

$$\textbf{terminatedAt}(CE,\ T) \leftarrow \textbf{happensAt}(E_{T_1},\ T), [\text{conditions}]$$
$$\ldots$$
$$\textbf{terminatedAt}(CE,\ T) \leftarrow \textbf{happensAt}(E_{T_j},\ T), [\text{conditions}]$$

where

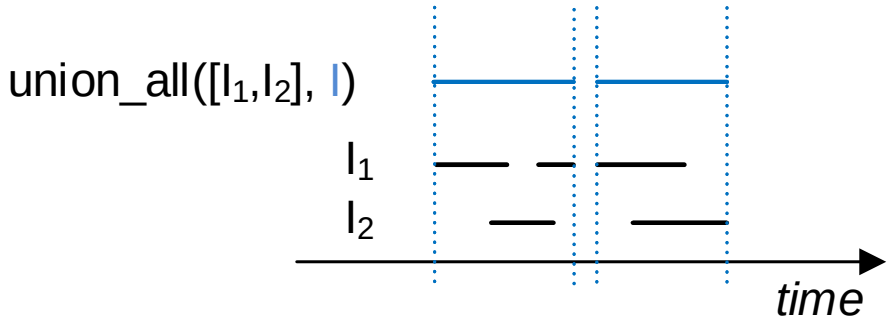
$$\text{conditions:} \quad {}^{0-K}\textbf{happensAt}(E_k,\ T), \ {}^{0-M}\textbf{holdsAt}(F_m,\ T), \ {}^{0-N}\text{atemporal-constraint}_n$$

Fleet Management

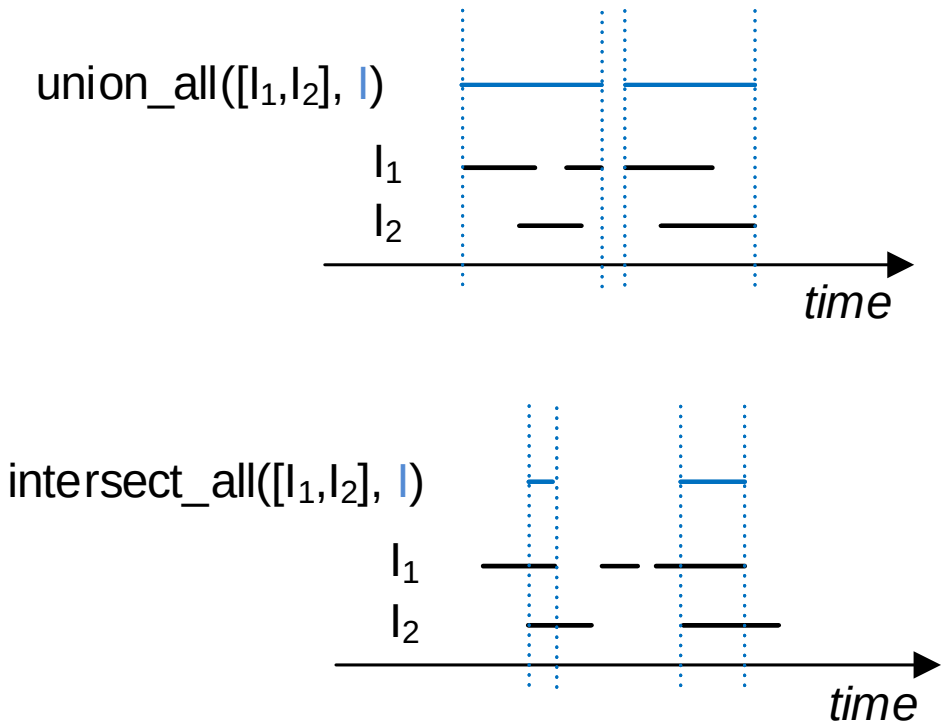


https://cer.iit.demokritos.gr (fleet management)

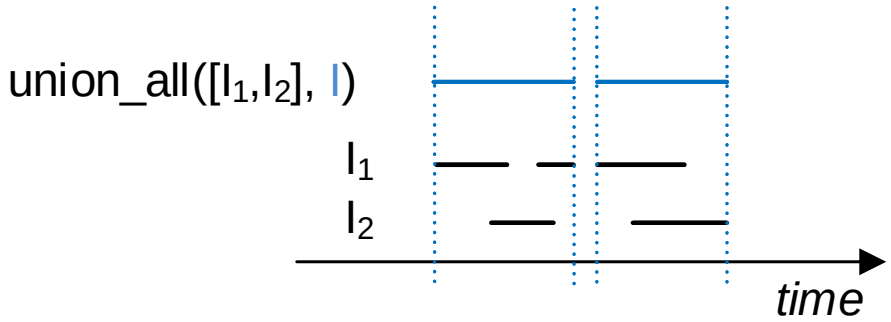
Interval Manipulation

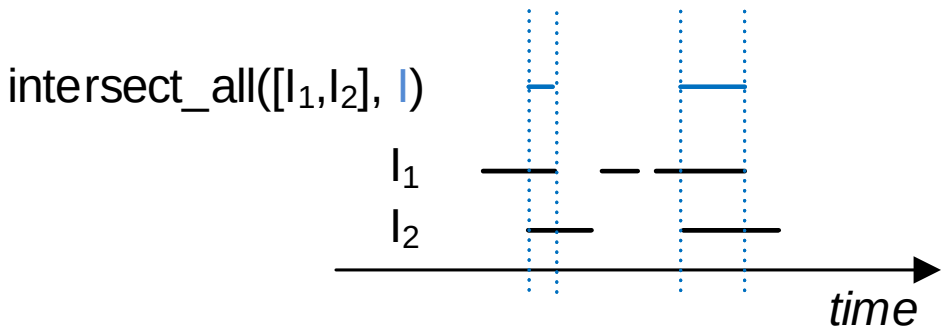


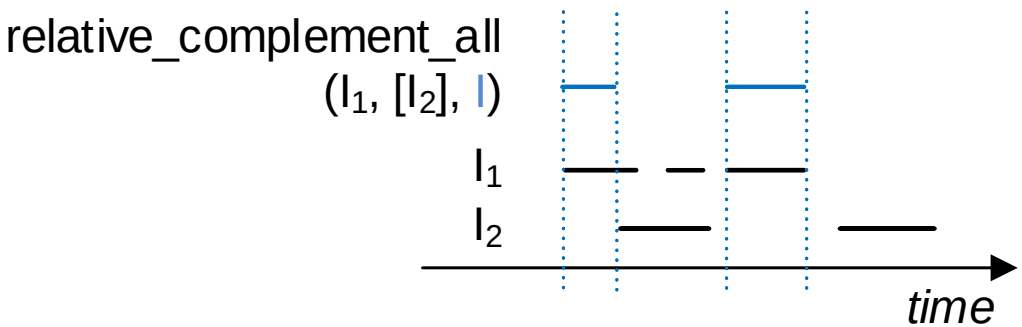
Interval Manipulation



Interval Manipulation







CE Definitions in the Run-Time Event Calculus

$$
\begin{aligned}
&\textbf{holdsFor}(\mathit{fighting}(P_1, P_2) = \text{true},\ I) \leftarrow \\
&\quad \textbf{holdsFor}(\mathit{abrupt}(P_1) = \text{true},\ I_1), \\
&\quad \textbf{holdsFor}(\mathit{abrupt}(P_2) = \text{true},\ I_2), \\
&\quad \textbf{union\_all}([I_1, I_2],\ I_3), \\
&\quad \textbf{holdsFor}(\mathit{close}(P_1, P_2) = \text{true},\ I_4), \\
&\quad \textbf{intersect\_all}([I_3, I_4],\ I_5), \\
&\quad \textbf{holdsFor}(\mathit{inactive}(P_1) = \text{true},\ I_6), \\
&\quad \textbf{holdsFor}(\mathit{inactive}(P_2) = \text{true},\ I_7), \\
&\quad \textbf{relative\_complement\_all}(I_5,\ [I_6, I_7],\ I)
\end{aligned}
$$

CE Definitions in the Run-Time Event Calculus

$$\begin{aligned}
&\textbf{holdsFor}(\mathit{fighting}(P_1, P_2) = \text{true},\ I) \leftarrow \\
&\quad \textbf{holdsFor}(\mathit{abrupt}(P_1) = \text{true},\ I_1), \\
&\quad \textbf{holdsFor}(\mathit{abrupt}(P_2) = \text{true},\ I_2), \\
&\quad \textbf{union_all}([I_1, I_2],\ I_3), \\
&\quad \textbf{holdsFor}(\mathit{close}(P_1, P_2) = \text{true},\ I_4), \\
&\quad \textbf{intersect_all}([I_3, I_4],\ I_5), \\
&\quad \textbf{holdsFor}(\mathit{inactive}(P_1) = \text{true},\ I_6), \\
&\quad \textbf{holdsFor}(\mathit{inactive}(P_2) = \text{true},\ I_7), \\
&\quad \textbf{relative_complement_all}(I_5,\ [I_6, I_7],\ I)
\end{aligned}$$

Shorthand:

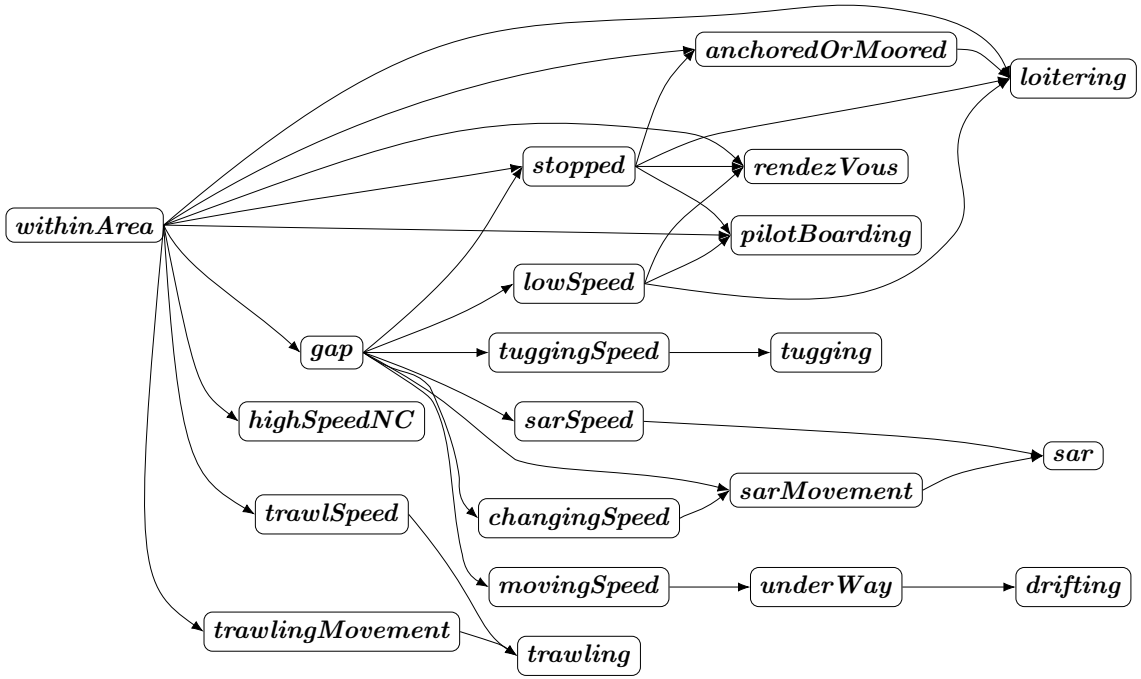
$$\begin{aligned}
&\mathit{fighting}(P_1, P_2) \text{ iff} \\
&\quad (\mathit{abrupt}(P_1) \text{ or } \mathit{abrupt}(P_2)), \\
&\quad \mathit{close}(P_1, P_2), \\
&\quad \text{not } (\mathit{inactive}(P_1) \text{ or } \mathit{inactive}(P_2))
\end{aligned}$$

Fighting

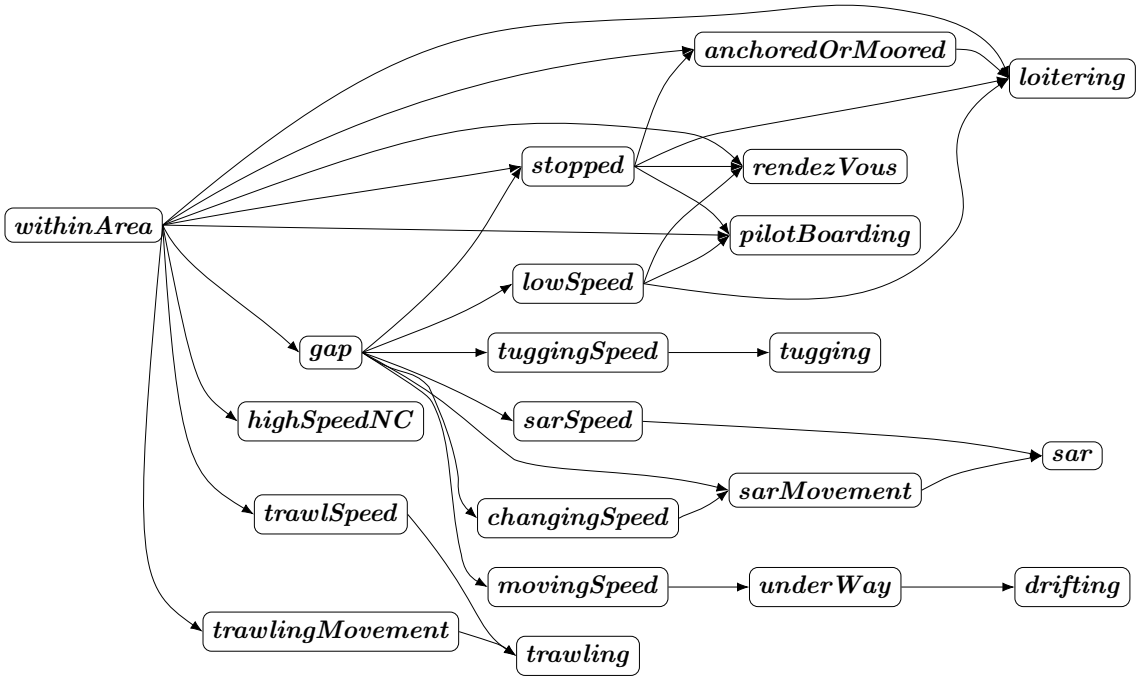
https://cer.iit.demokritos.gr (activity-recognition-i)

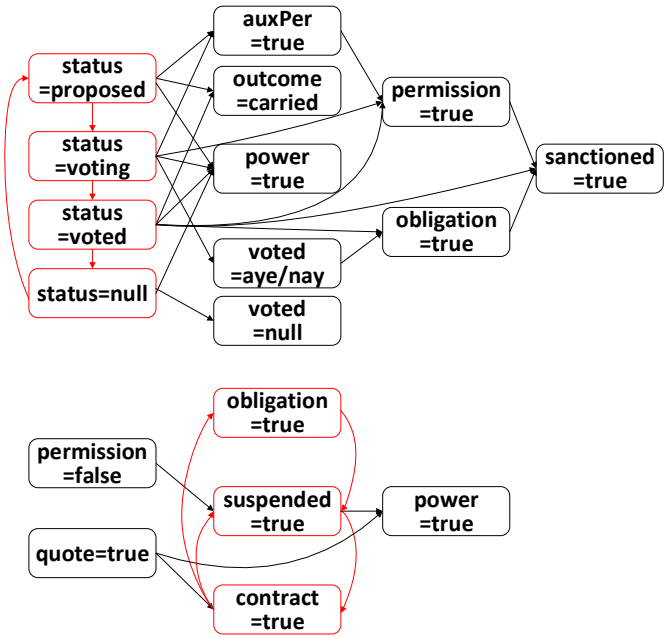
https://cer.iit.demokritos.gr (activity-recognition-ii)

Semantics

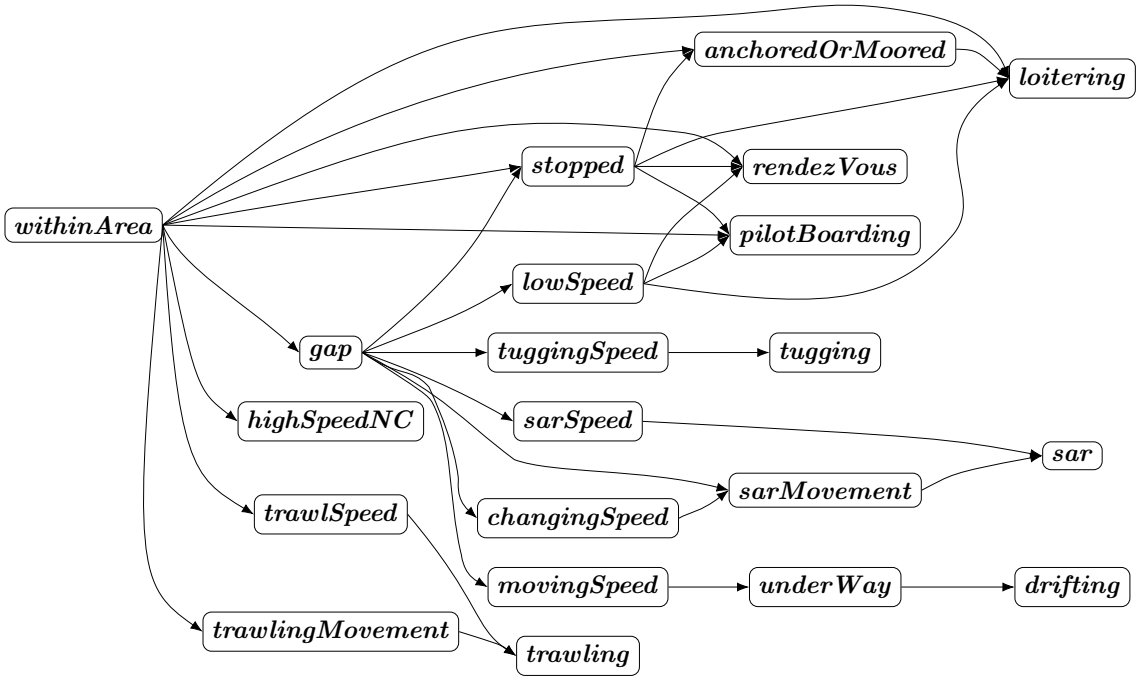

withinArea
anchoredOrMoored
loitering
stopped
rendezVous
pilotBoarding
lowSpeed
gap
tuggingSpeed
tugging
highSpeedNC
sarSpeed
sar
sarMovement
trawlSpeed
changingSpeed
movingSpeed
underWay
drifting
trawlingMovement
trawling

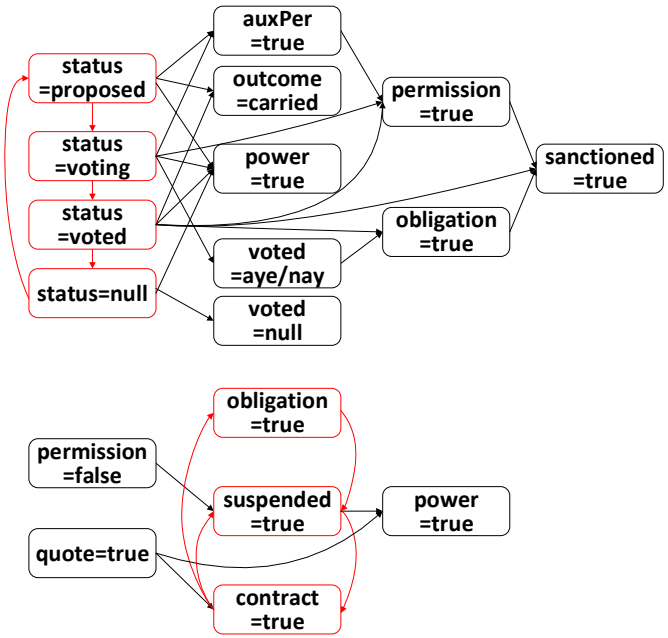
Semantics


withinArea
anchoredOrMoored
loitering
stopped
rendezVous
pilotBoarding
lowSpeed
gap
tuggingSpeed
tugging
highSpeedNC
sarSpeed
sar
sarMovement
trawlSpeed
changingSpeed
movingSpeed
underWay
drifting
trawlingMovement
trawling


status =proposed
status =voting
status =voted
status=null
auxPer =true
outcome =carried
power =true
voted =aye/nay
voted =null
permission =true
obligation =true
sanctioned =true
permission =false
quote=true
obligation =true
suspended =true
contract =true
power =true

Semantics



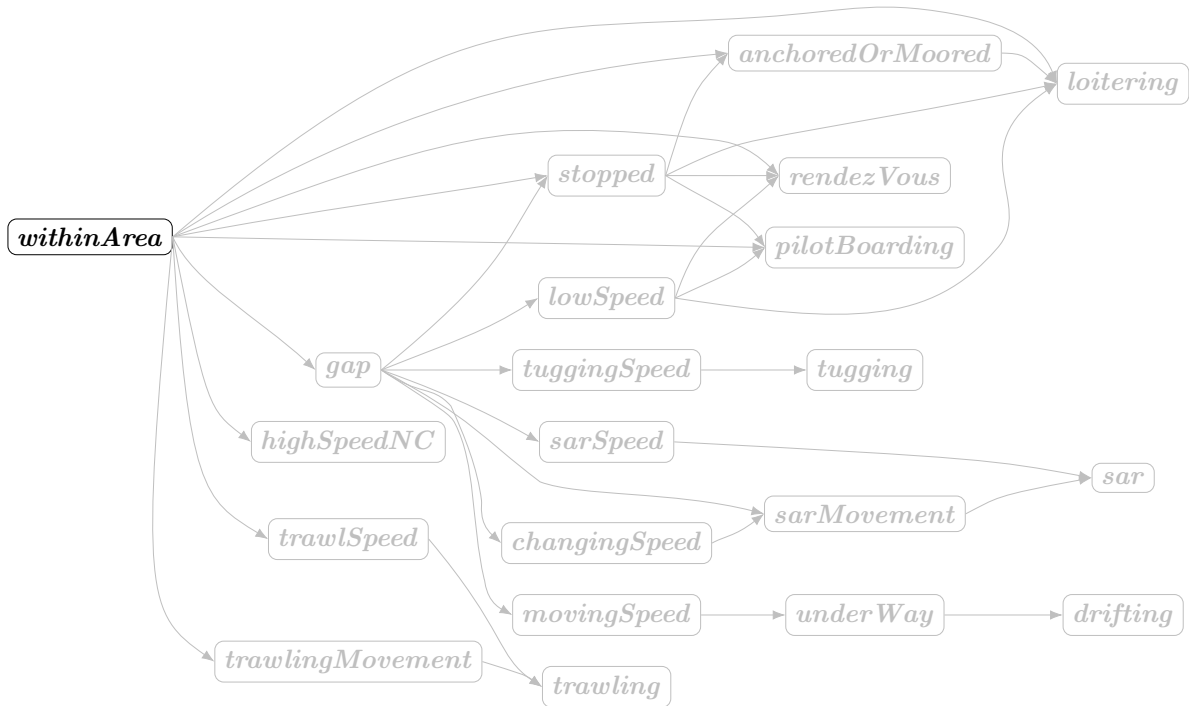


Proposition

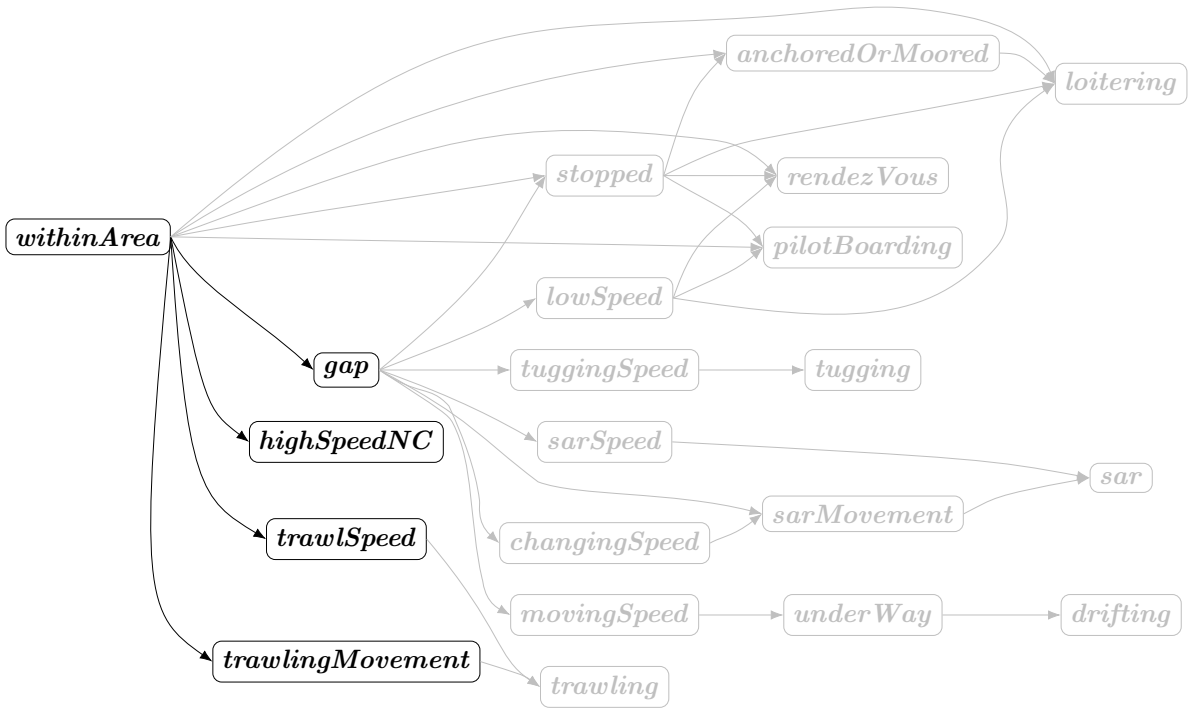
An event description in RTEC is a locally stratified logic program.

Mantenoglou et al, Stream Reasoning with Cycles. Knowledge Representation and Reasoning (KR), 2022.

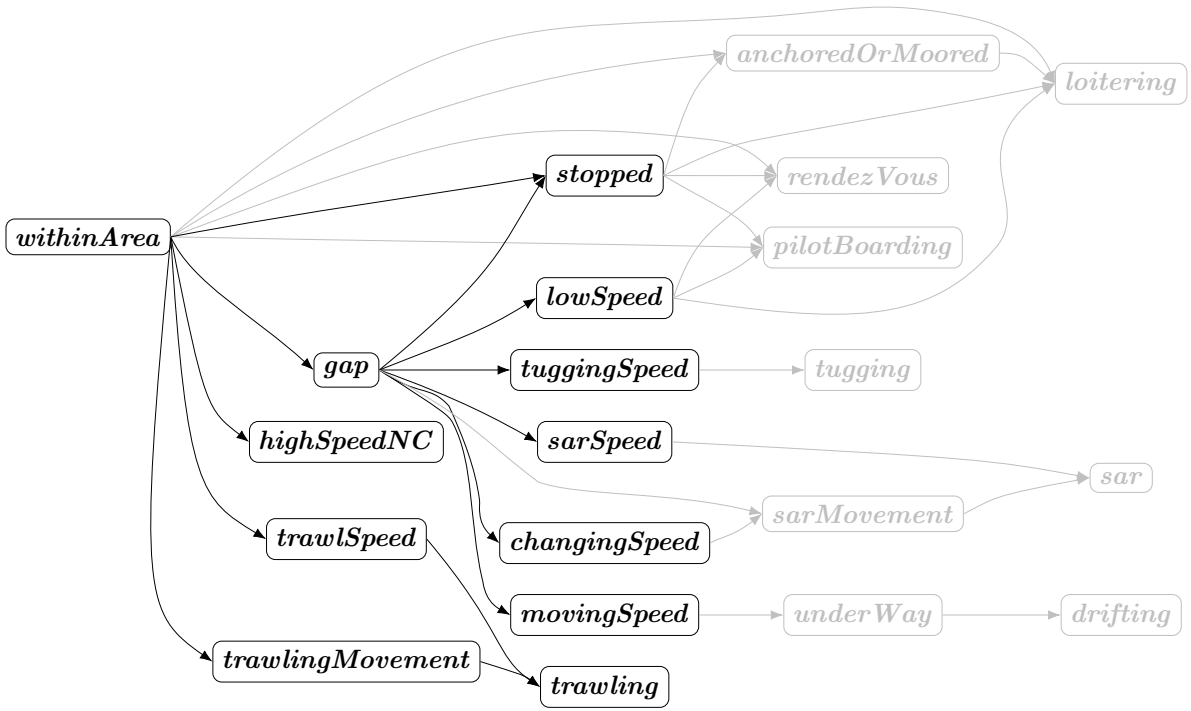
Stratification & Reasoning



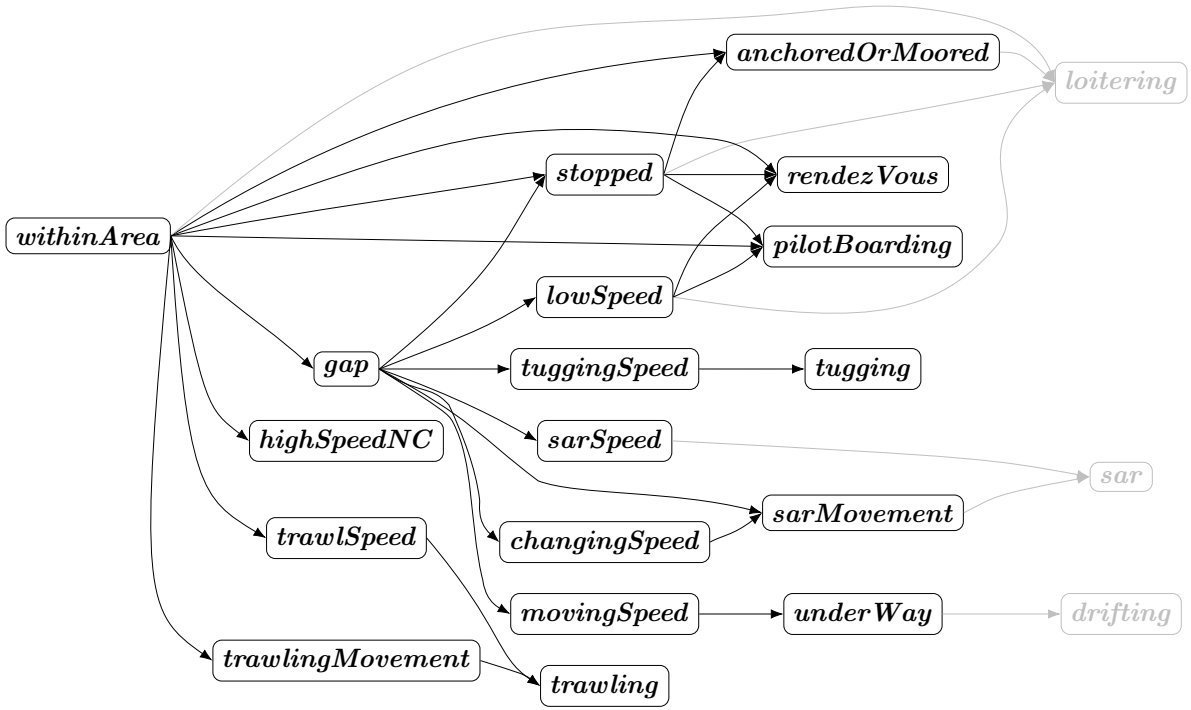
Stratification & Reasoning



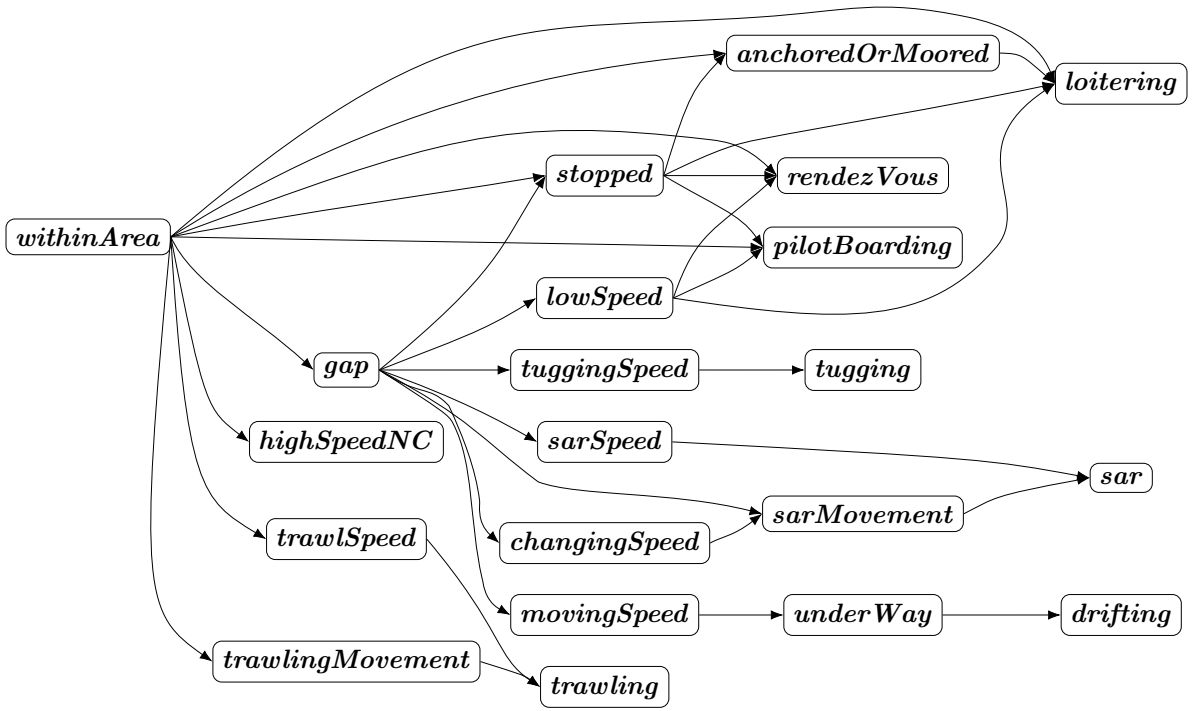
Stratification & Reasoning



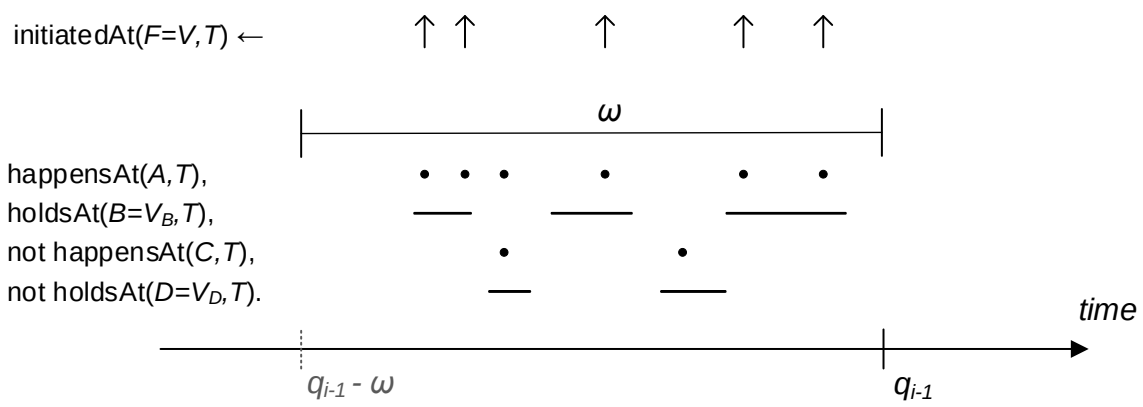
Stratification & Reasoning



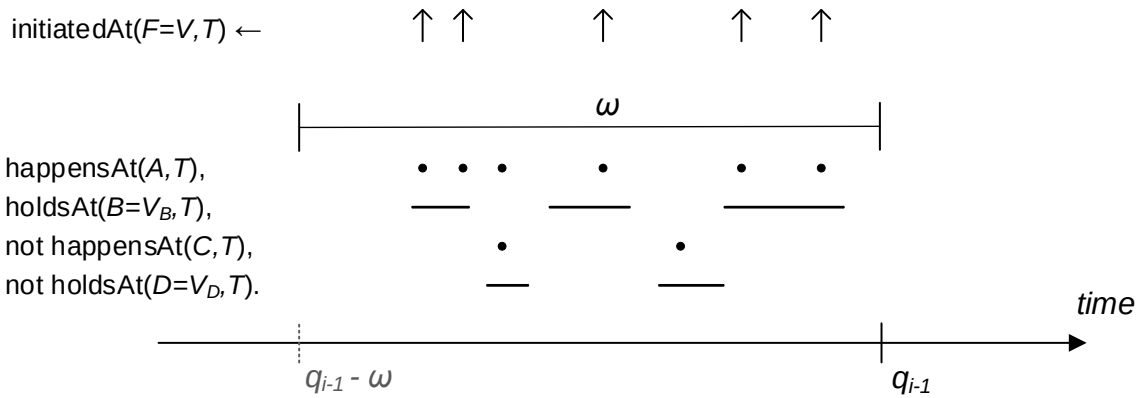
Stratification & Reasoning

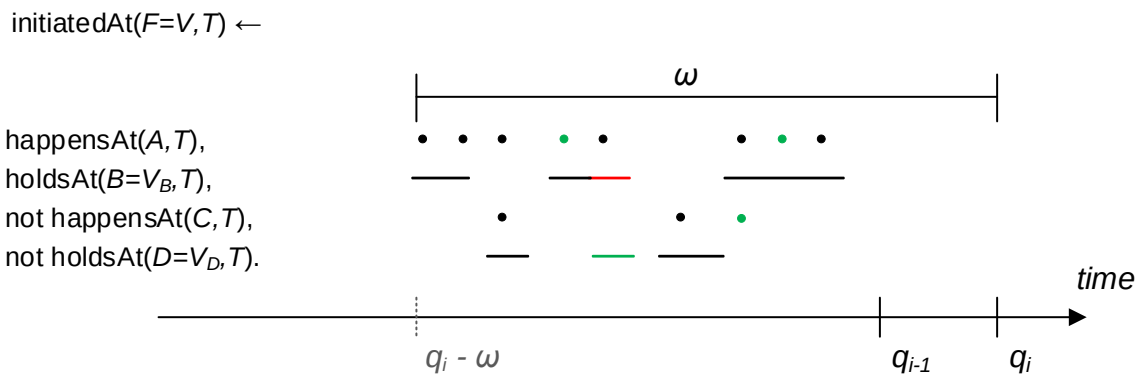


Windowing

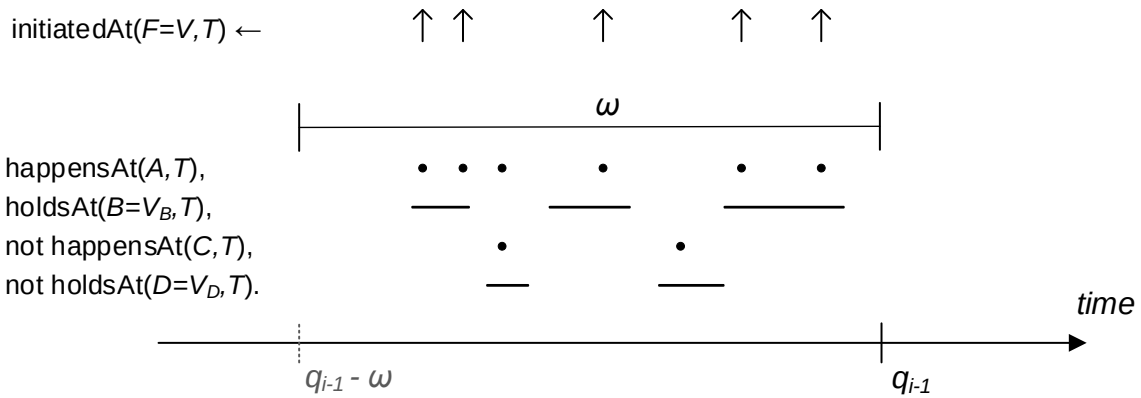

initiatedAt(F=V,T) ←
ω
happensAt(A,T),
holdsAt(B=V_B,T),
not happensAt(C,T),
not holdsAt(D=V_D,T).
time
q_{i-1} - ω
q_{i-1}

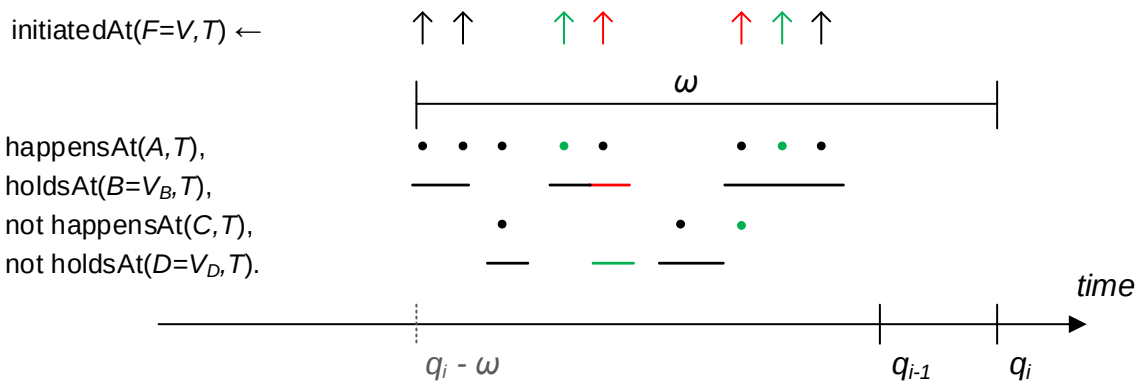
Windowing


initiatedAt(F=V,T) ←
ω
happensAt(A,T),
holdsAt(B=V_B,T),
not happensAt(C,T),
not holdsAt(D=V_D,T).
time
q_{i-1} - ω
q_{i-1}

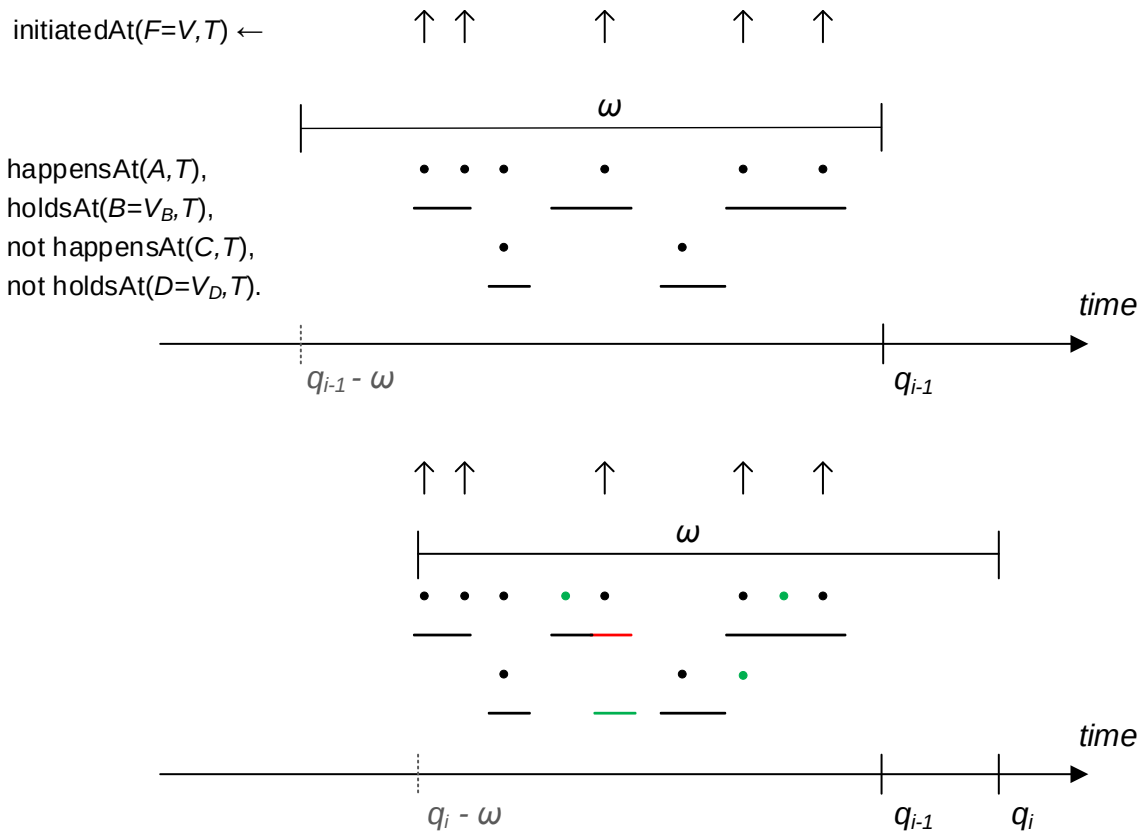

initiatedAt(F=V,T) ←
ω
happensAt(A,T),
holdsAt(B=V_B,T),
not happensAt(C,T),
not holdsAt(D=V_D,T).
time
q_i - ω
q_{i-1}
q_i

Windowing


initiatedAt(F=V,T) ←
ω
happensAt(A,T),
holdsAt(B=V_B,T),
not happensAt(C,T),
not holdsAt(D=V_D,T).
time
q_{i-1} - ω
q_{i-1}

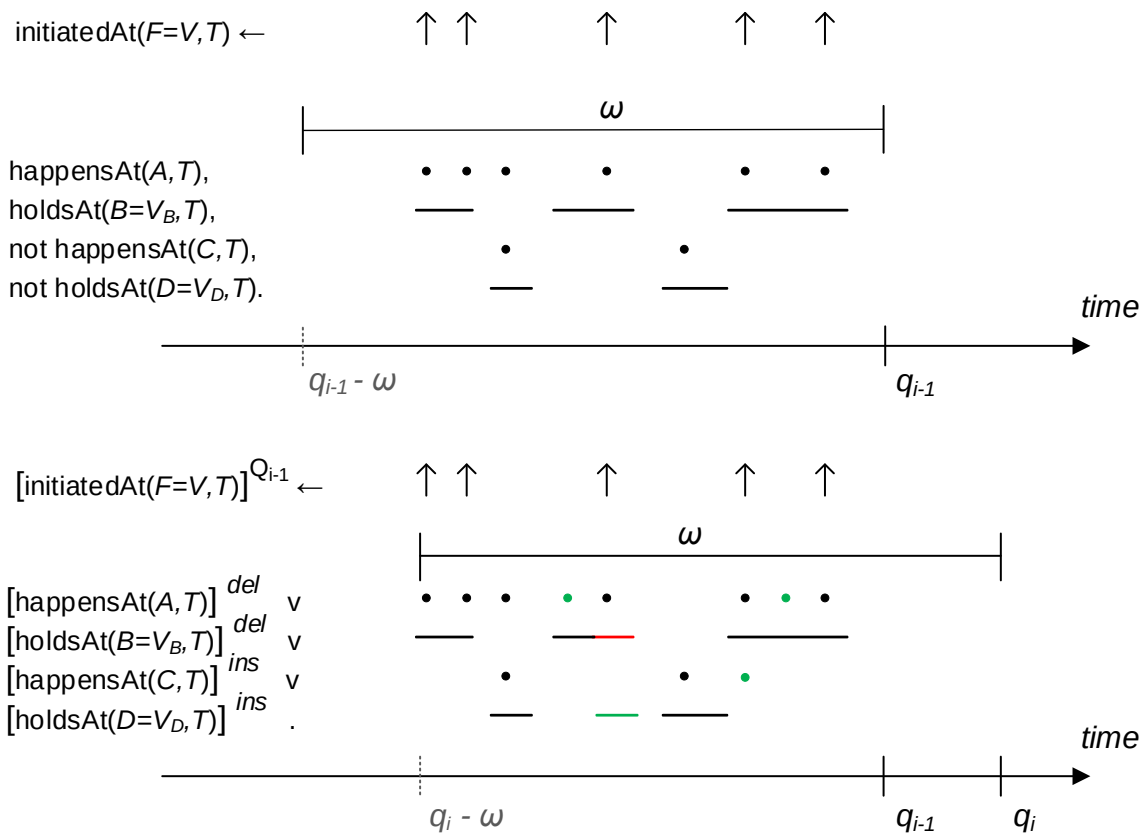

initiatedAt(F=V,T) ←
ω
happensAt(A,T),
holdsAt(B=V_B,T),
not happensAt(C,T),
not holdsAt(D=V_D,T).
time
q_i - ω
q_{i-1}
q_i

Incremental Reasoning: Deletion Phase



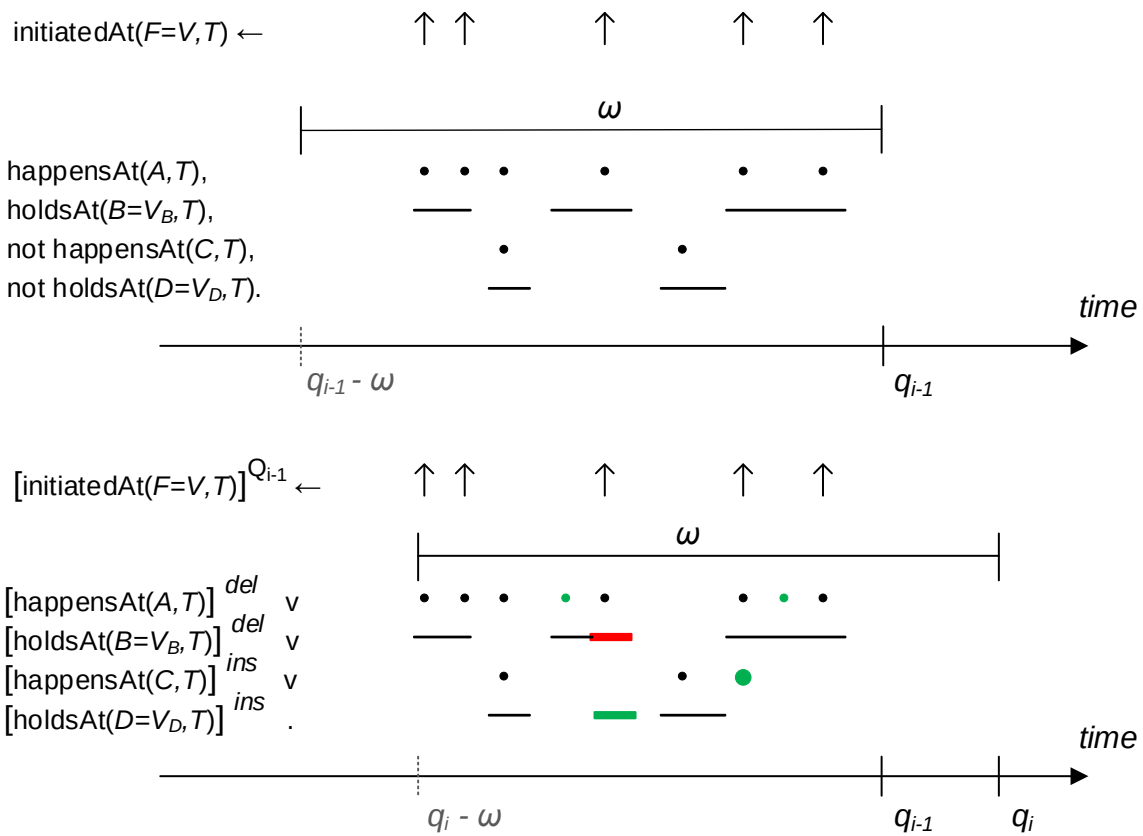
Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

Incremental Reasoning: Deletion Phase



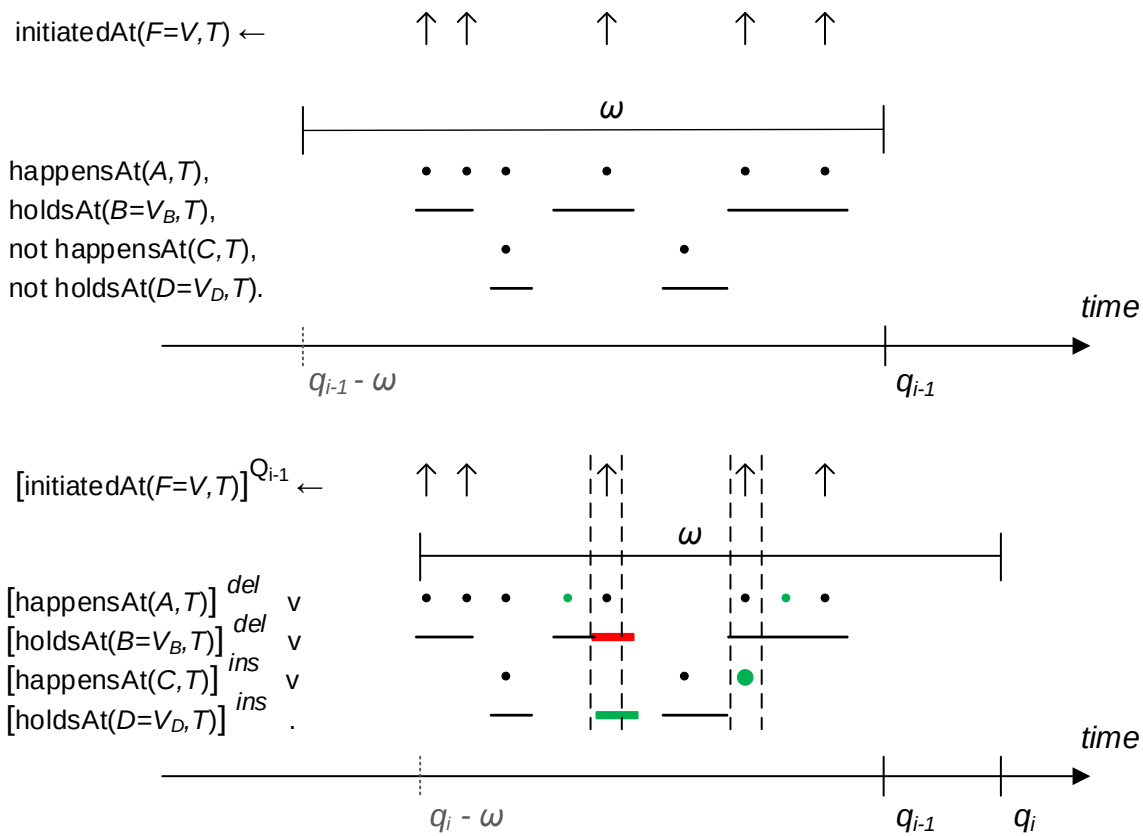
Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

Incremental Reasoning: Deletion Phase



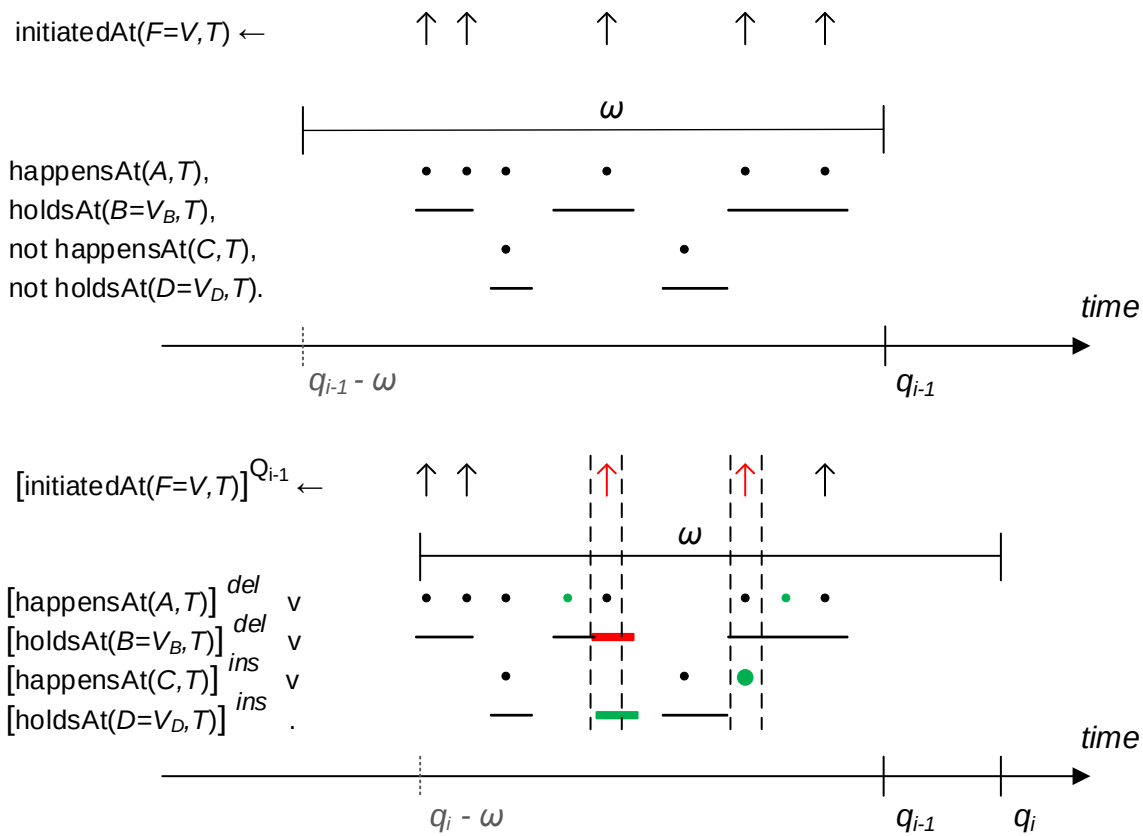
Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

Incremental Reasoning: Deletion Phase



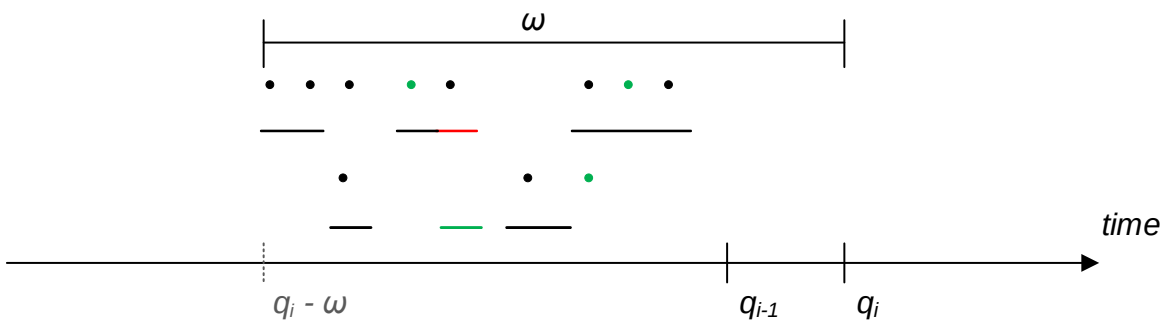
Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

Incremental Reasoning: Deletion Phase



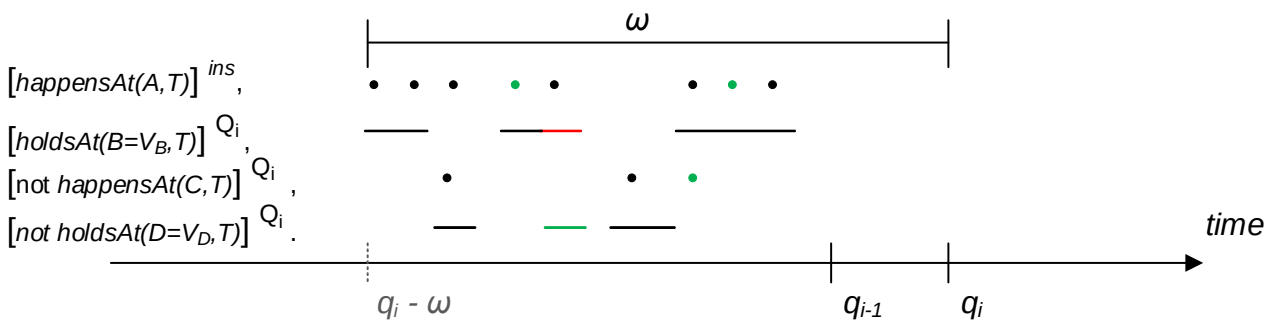
Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

Incremental Reasoning: Addition Phase



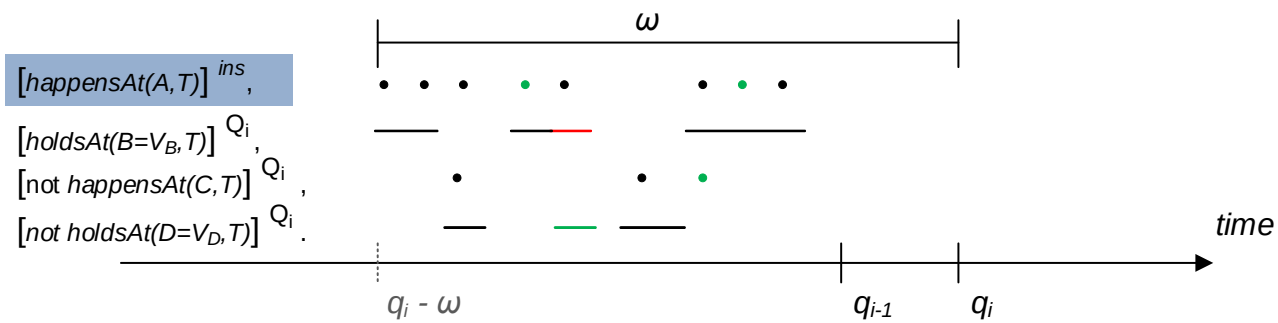
Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

Incremental Reasoning: Addition Phase



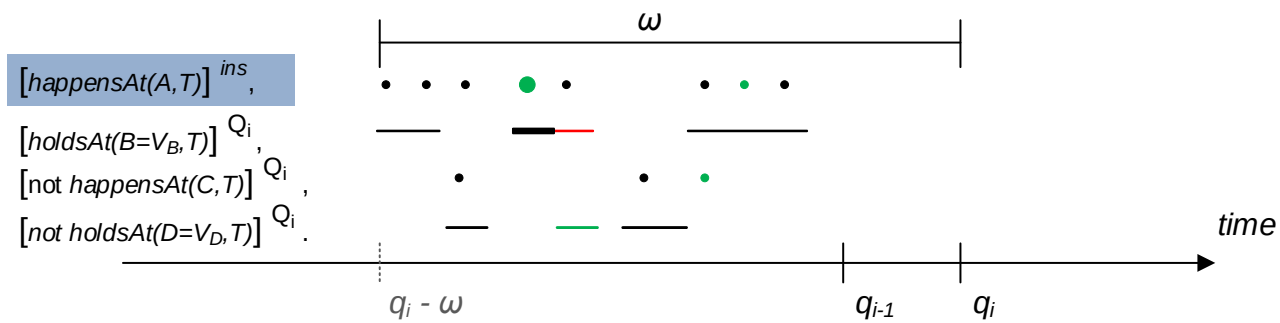
Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

Incremental Reasoning: Addition Phase



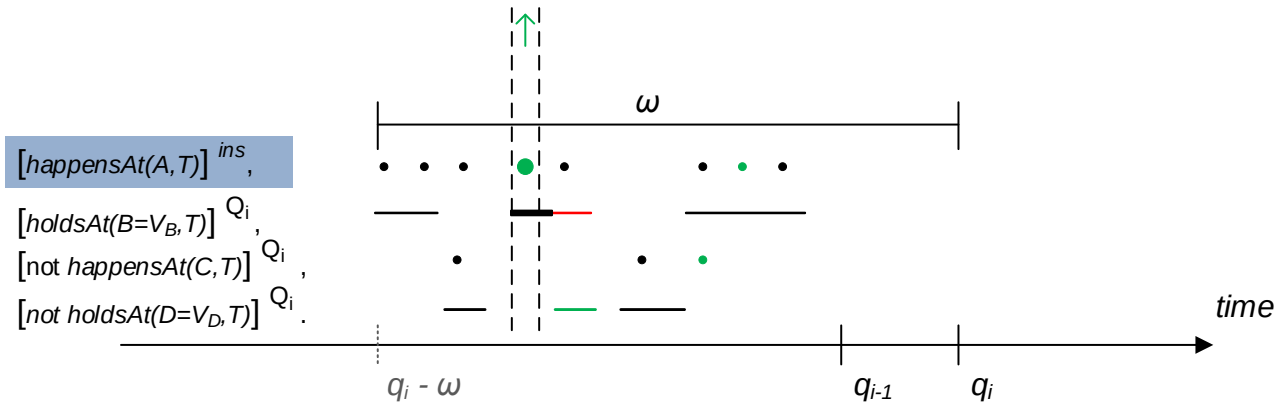
Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

Incremental Reasoning: Addition Phase



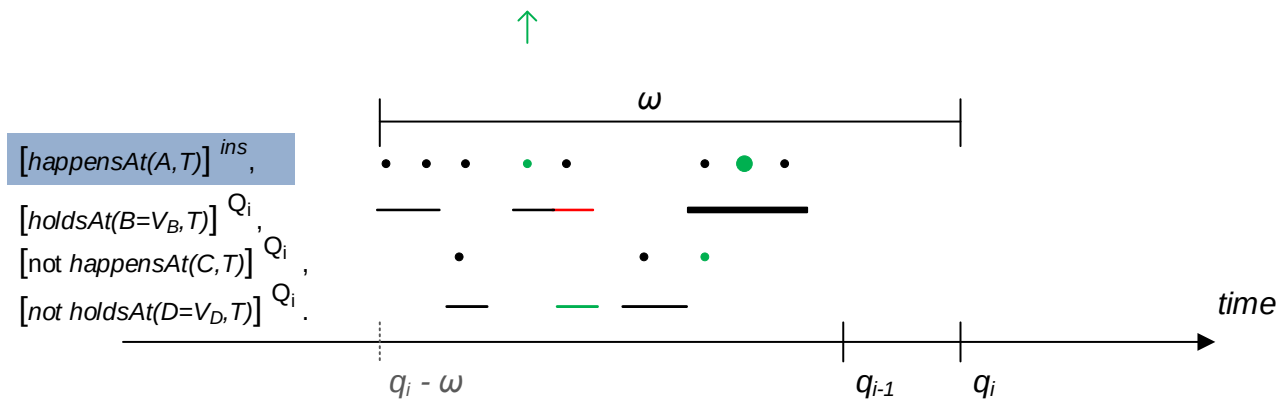
Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

Incremental Reasoning: Addition Phase



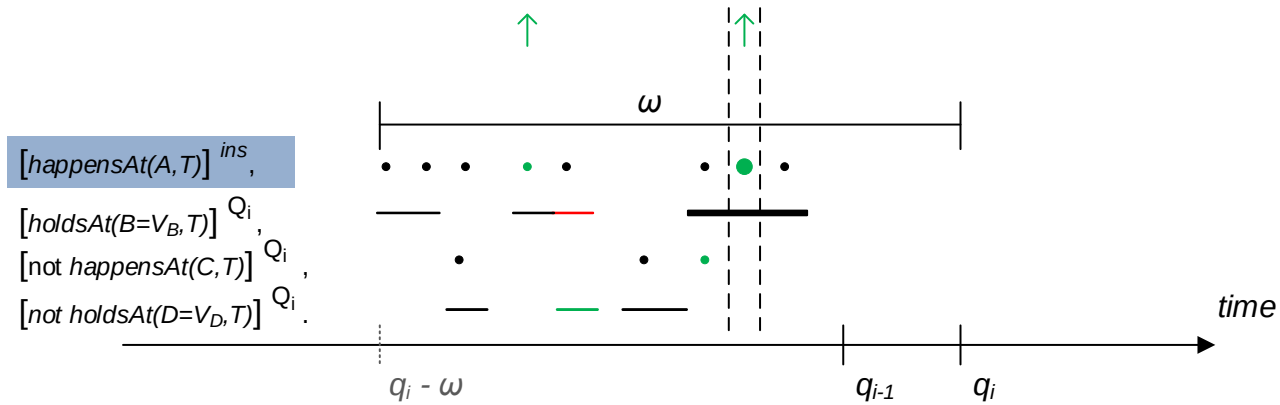
Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

Incremental Reasoning: Addition Phase



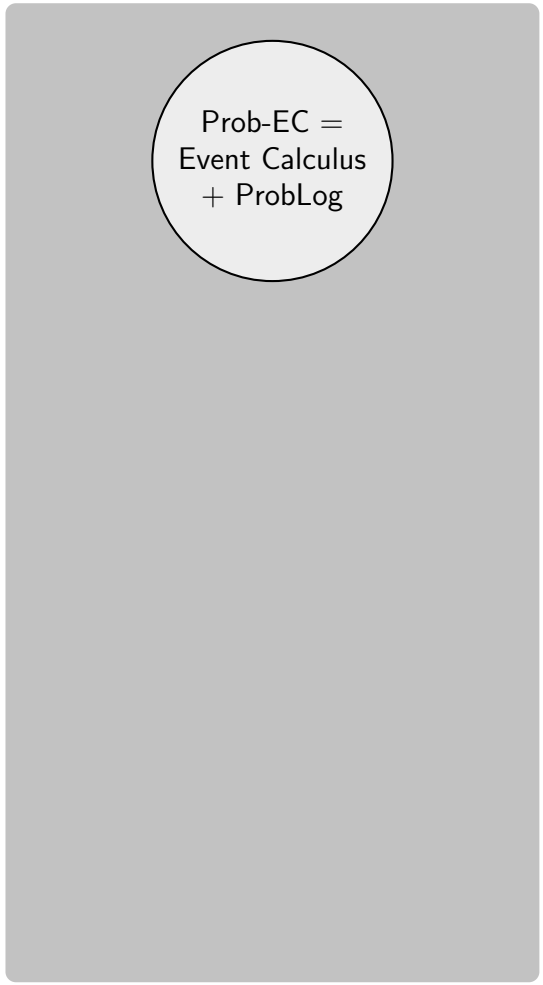
Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

Incremental Reasoning: Addition Phase

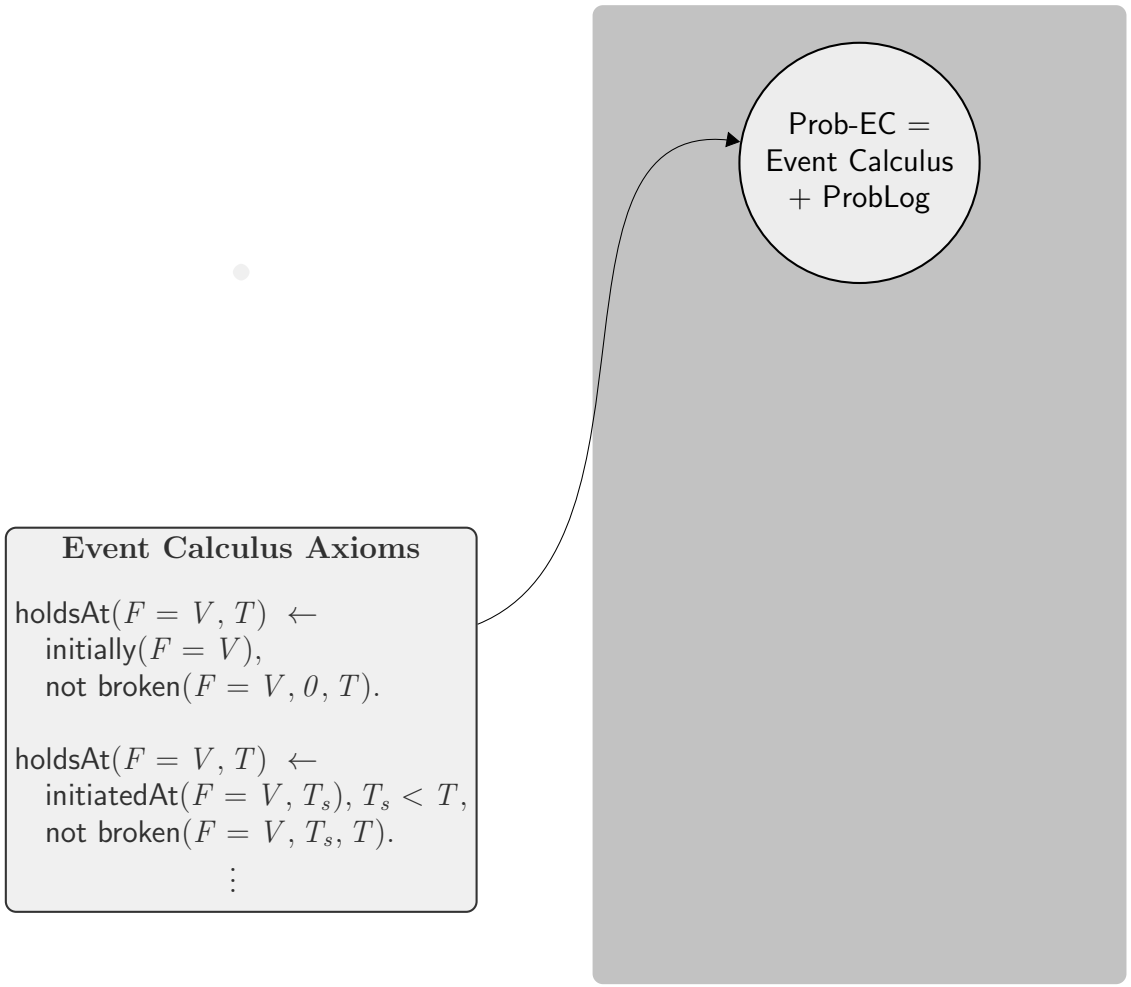


Tsilionis et al, Incremental Event Calculus for Run-Time Reasoning. Journal of AI Research (JAIR), 2022.

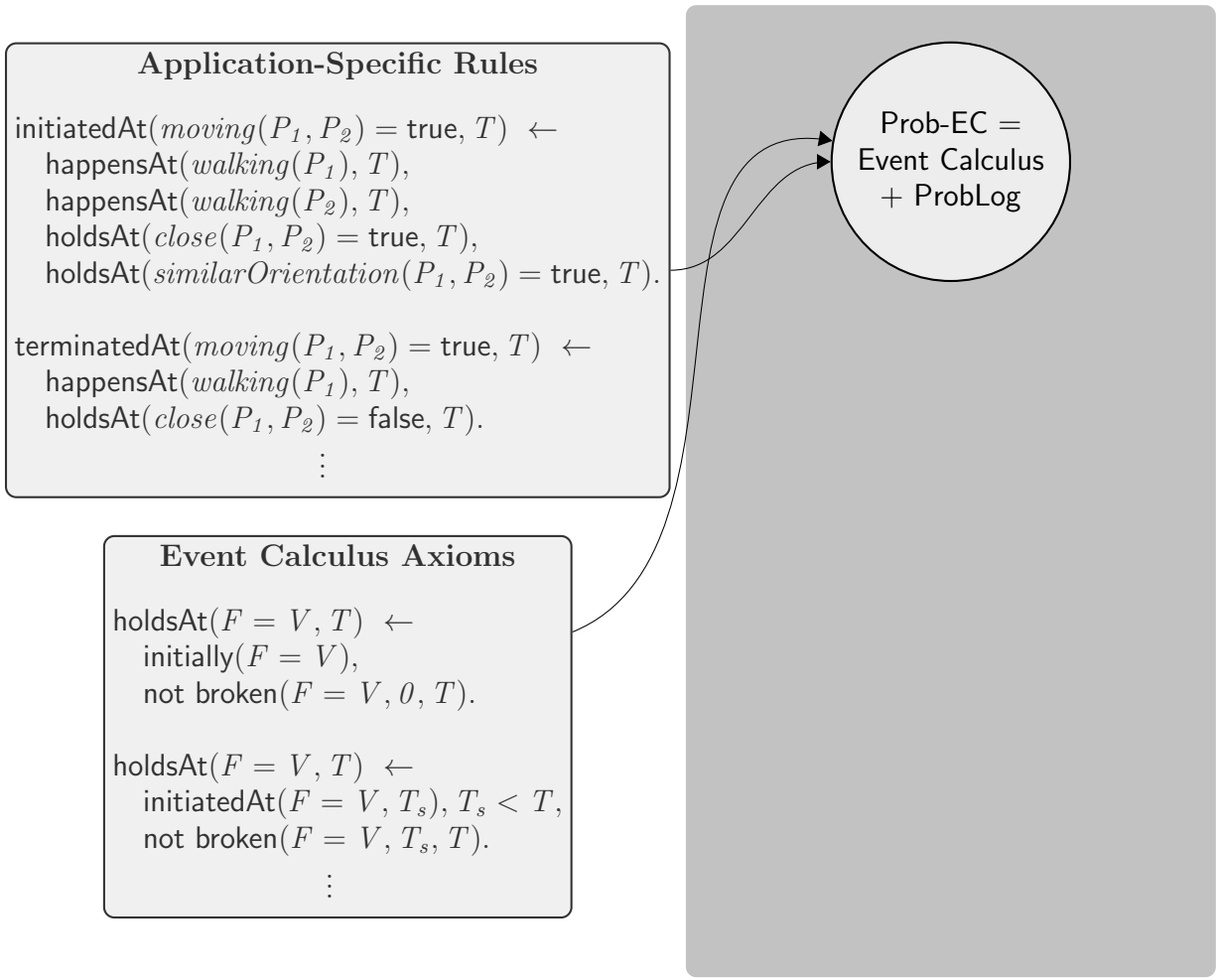
Noisy Data Streams: A Probabilistic Event Calculus



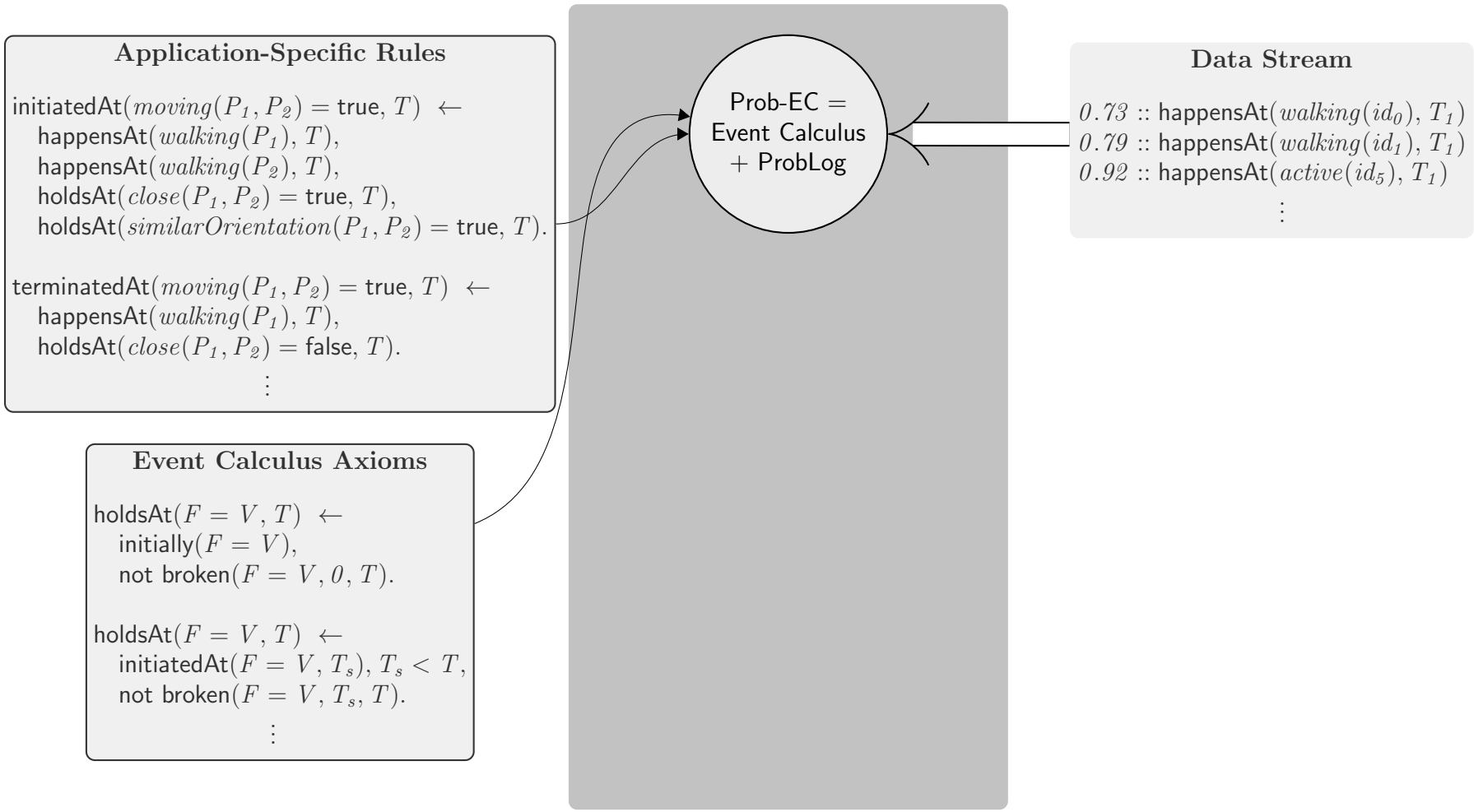
Noisy Data Streams: A Probabilistic Event Calculus



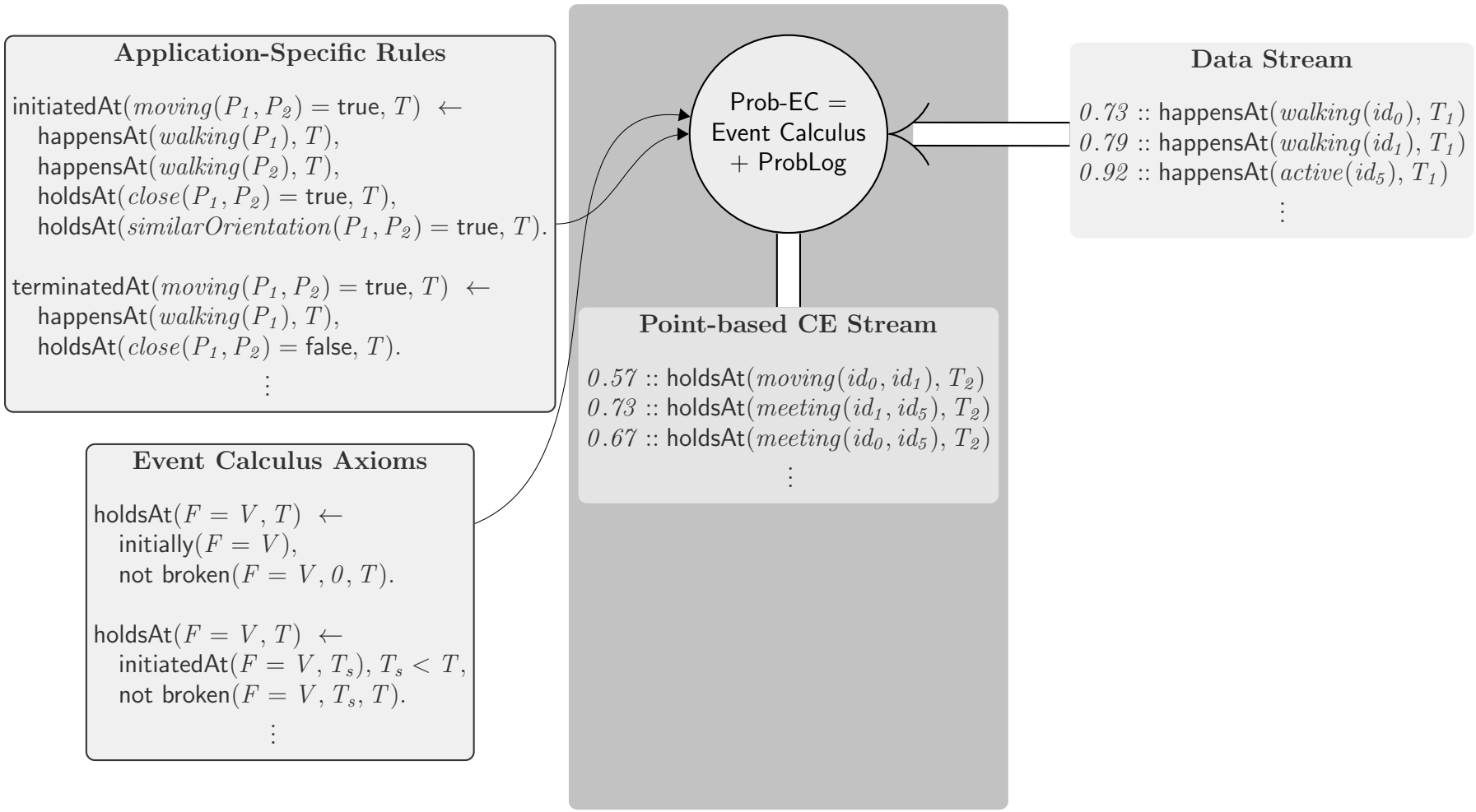
Noisy Data Streams: A Probabilistic Event Calculus



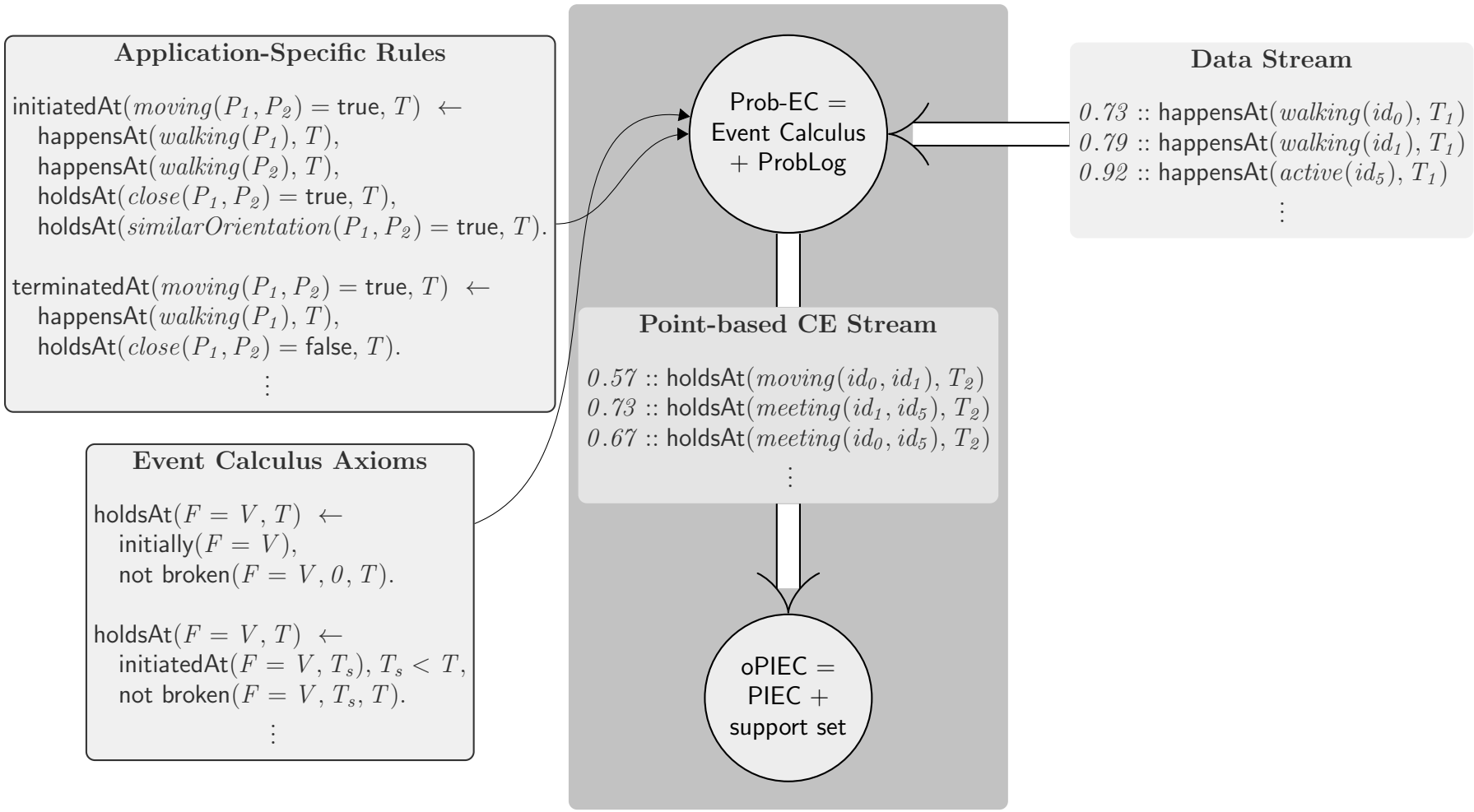
Noisy Data Streams: A Probabilistic Event Calculus



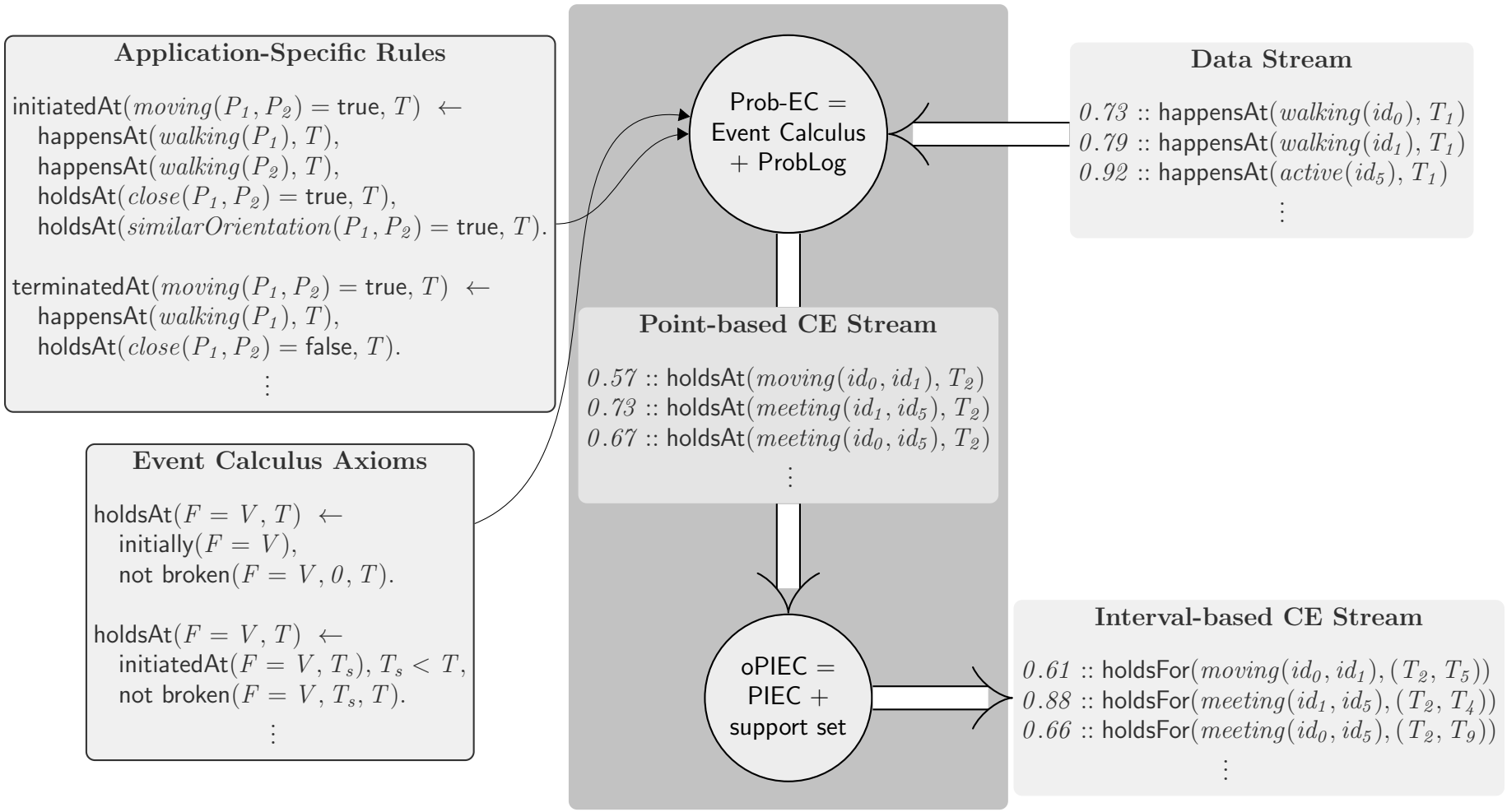
Noisy Data Streams: A Probabilistic Event Calculus



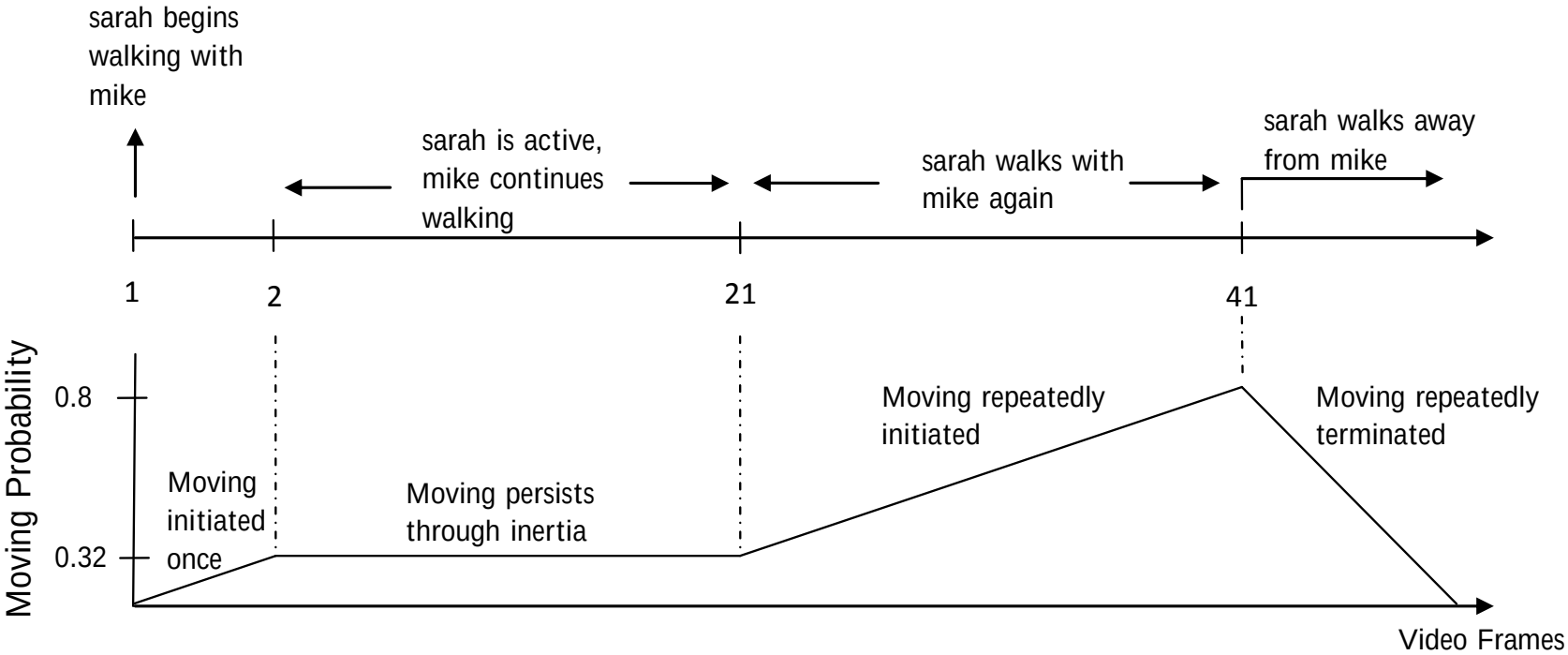
Noisy Data Streams: A Probabilistic Event Calculus



Noisy Data Streams: A Probabilistic Event Calculus



Human Activity Recognition

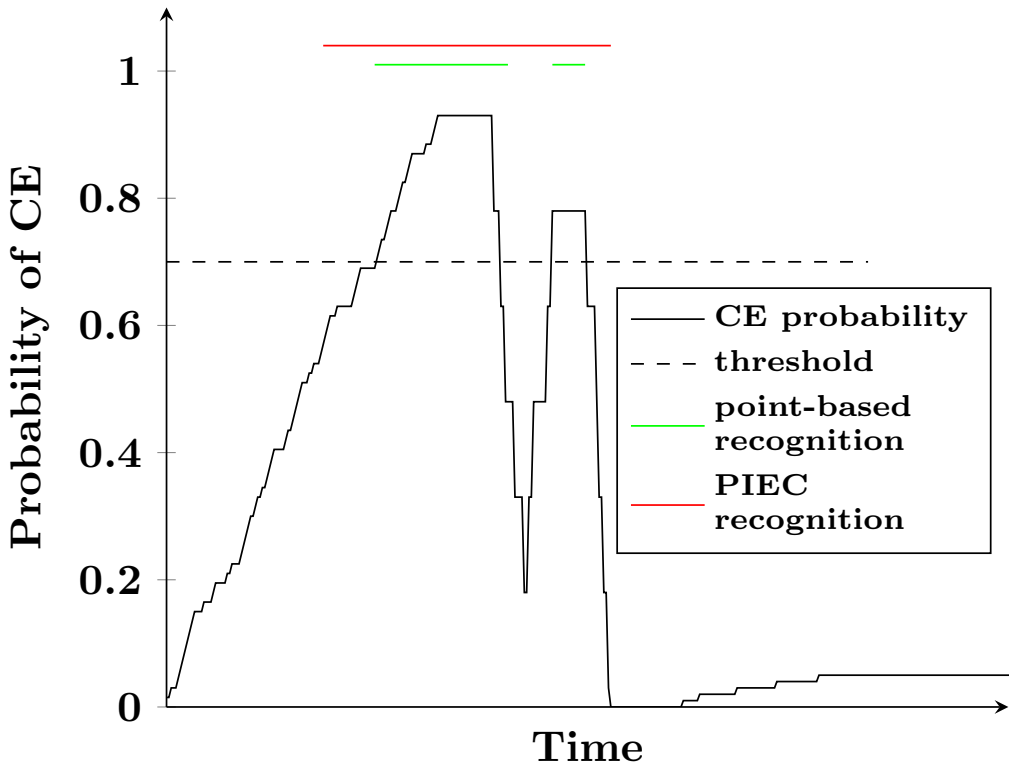


$\text{initiatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$
$\quad \text{happensAt}(\mathit{walking}(P_1),\ T),$
$\quad \text{happensAt}(\mathit{walking}(P_2),\ T),$
$\quad \text{holdsAt}(\mathit{close}(P_1, P_2) = \text{true},\ T),$
$\quad \text{holdsAt}(\mathit{similarOrientation}(P_1, P_2) = \text{true},\ T).$

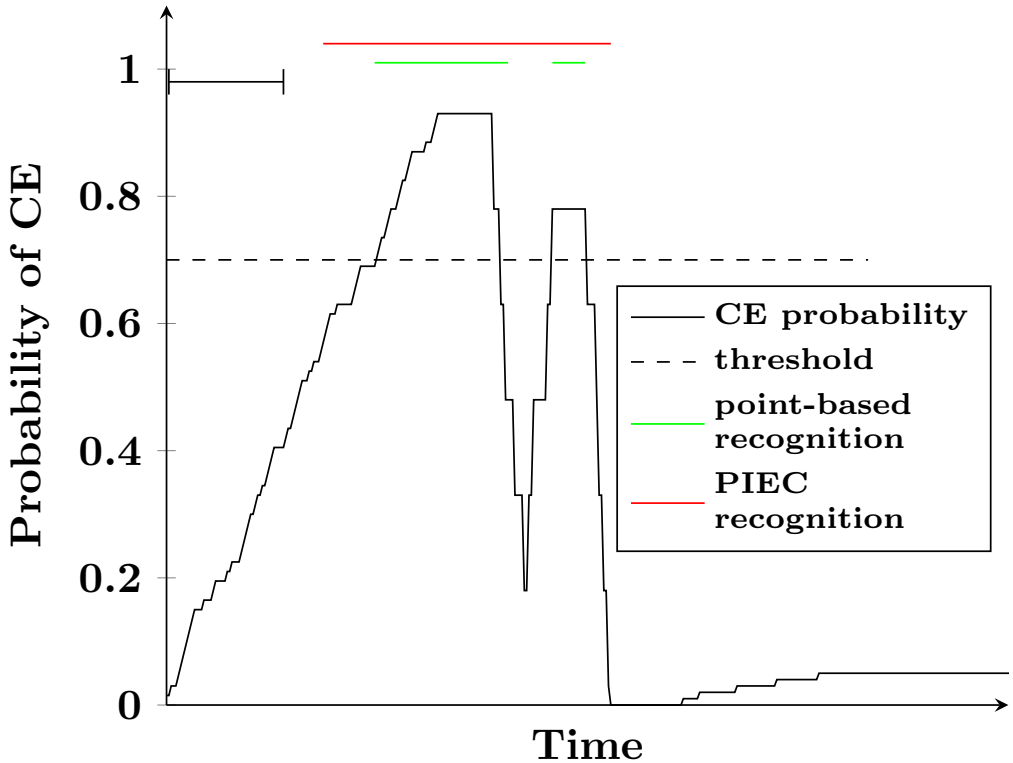
$\text{terminatedAt}(\mathit{moving}(P_1, P_2) = \text{true},\ T) \leftarrow$
$\quad \text{happensAt}(\mathit{walking}(P_1),\ T),$
$\quad \text{holdsAt}(\mathit{close}(P_1, P_2) = \text{false},\ T).$

$0.70 :: \text{happensAt}(\mathit{walking}(\mathit{mike}),\ 1).$
$0.46 :: \text{happensAt}(\mathit{walking}(\mathit{sarah}),\ 1).$
$0.73 :: \text{happensAt}(\mathit{walking}(\mathit{mike}),\ 2).$
$0.55 :: \text{happensAt}(\mathit{active}(\mathit{sarah}),\ 2).$
$0.69 :: \text{happensAt}(\mathit{walking}(\mathit{mike}),\ 21).$
$0.58 :: \text{happensAt}(\mathit{walking}(\mathit{sarah}),\ 21).$
$0.18 :: \text{happensAt}(\mathit{inactive}(\mathit{mike}),\ 41).$
$0.32 :: \text{happensAt}(\mathit{walking}(\mathit{sarah}),\ 41).$

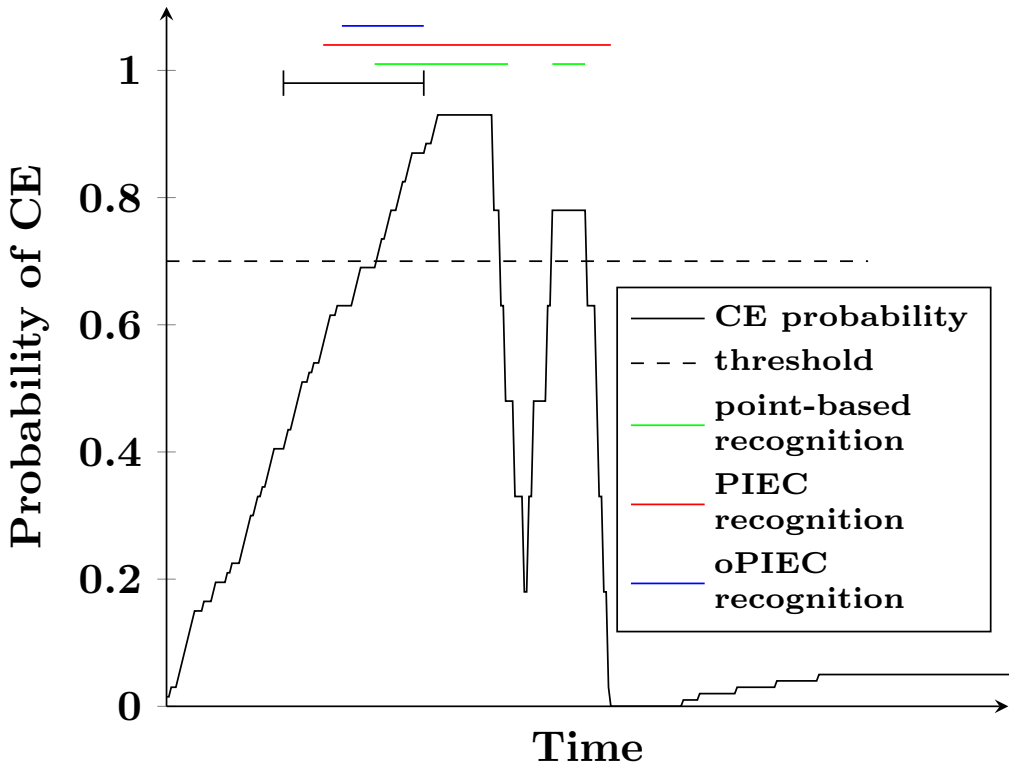
Instantaneous vs Interval-based Recognition



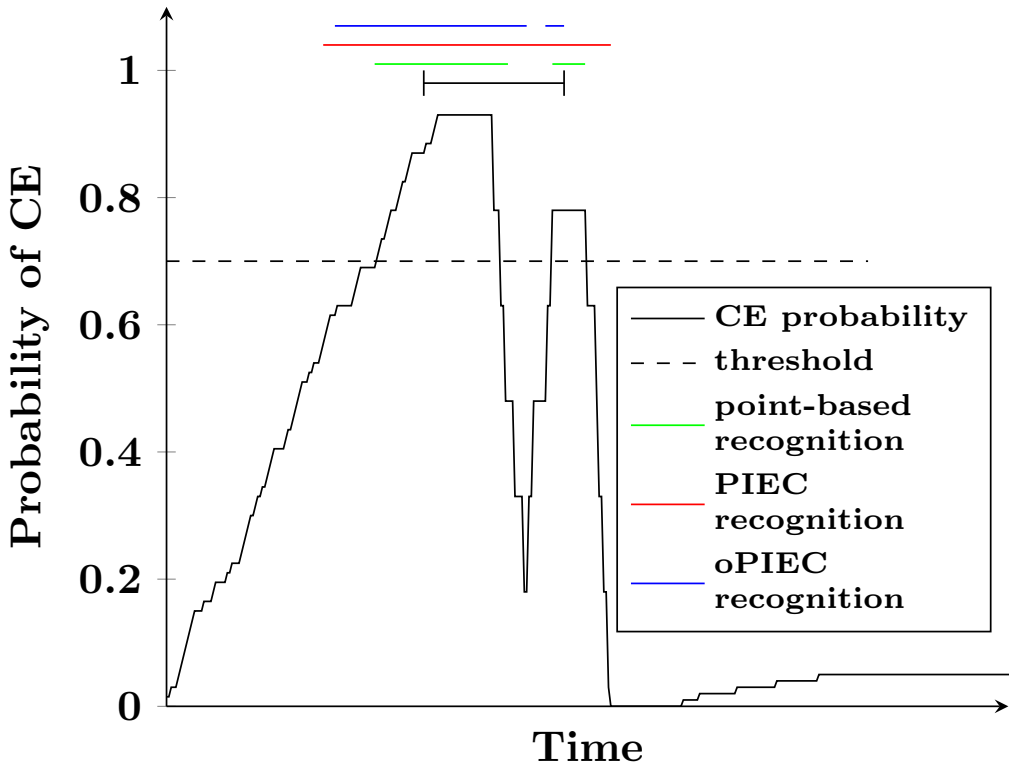
Online Interval-based Recognition



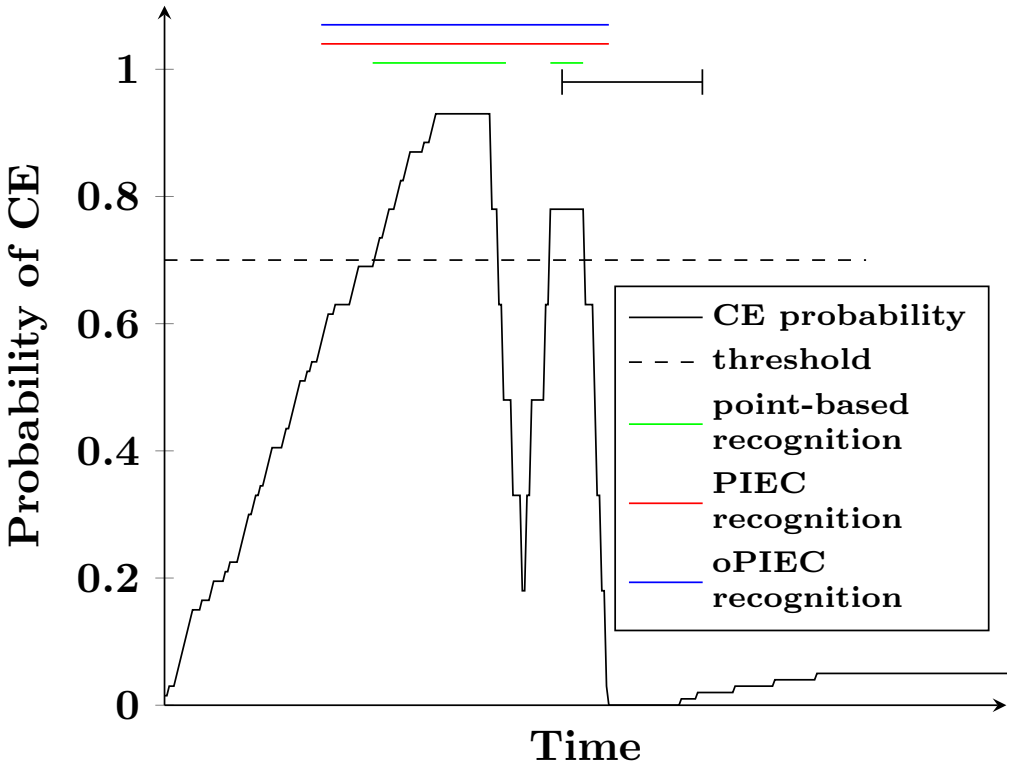
Online Interval-based Recognition



Online Interval-based Recognition



Online Interval-based Recognition



- Optimal stream history compression.
- Comparable accuracy to batch processing with very small memory.

Mantenoglou et al, Online Probabilistic Interval-based Event Calculus. European Conference on AI (ECAI), 2020.

https://github.com/Periklismant/oPIEC

Topics not covered

- Uncertainty in the event patterns*.
- Automated construction of event patterns†.
  - Semi-supervised ML‡.
- Neuro-symbolic reasoning for end-to-end CER◇.
- **Other approaches on formal CER▽**.
- Complex event forecasting♠.

* *Alevizos et al, Probabilistic Complex Event Recognition: A Survey. ACM Computing Surveys, 2017.*

† *Katzouris et al, Online Learning Probabilistic Event Calculus Theories in Answer Set Programming. Theory and Practice of Logic Programming, 2022.* https://github.com/nkatzz/ORL

‡ *Michelioudakis et al, Semi-Supervised Online Structure Learning for Composite Event Recognition. Machine Learning, 2019.* https://github.com/anskarl/LoMRF

◇ *Manhaeve et al, Neural probabilistic logic programming in DeepProbLog. Artificial Intelligence, 2021.* https://github.com/ML-KULeuven/deepproblog

▽ *Artikis et al, Dagstuhl Seminar on the Foundations of Composite Event Recognition. SIGMOD Record, 2020.*

♠ *Alevizos et al, Complex Event Forecasting with Prediction Suffix Trees. VLDB Journal, 2022.* https://github.com/ElAlev/Wayeb