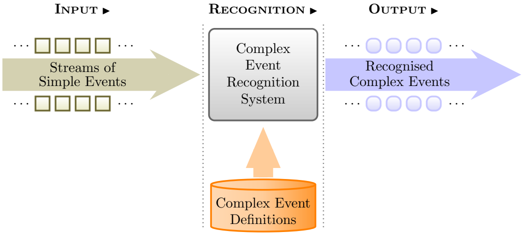
# Event Calculus for Run-Time Reasoning

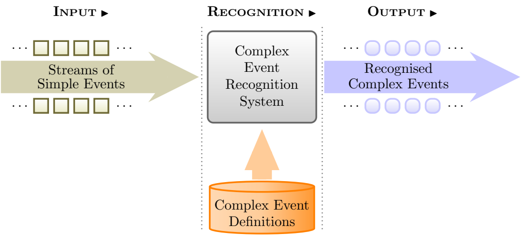
Periklis Mantenoglou[1,2] Alexander Artikis[1,3]

[1]National Kapodistrian University of Athens, Greece
[2]National Research Centre 'Demokritos', Athens, Greece
[3]University of Piraeus, Athens, Greece

# Stream Reasoning



# Stream Reasoning

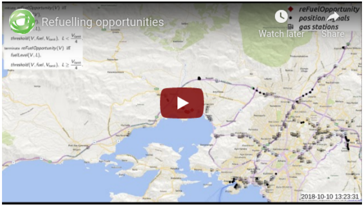




https://cer.iit.demokritos.gr (maritime)

# Stream Reasoning







https://cer.iit.demokritos.gr

# Stream Reasoning

Problem:

- Continuous pattern matching over data streams
- Reasoning over complex temporal specifications
- Reporting complex event instances with minimal latency

# Stream Reasoning

Problem:

- Continuous pattern matching over data streams
- Reasoning over complex temporal specifications
- Reporting complex event instances with minimal latency

Approach:

- Complex temporal specifications $\Longrightarrow$ Event Calculus
- Stream reasoning $\Longrightarrow$ Run-Time Event Calculus (RTEC)

# 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.

---

Robert A. Kowalski, Marek J. Sergot: A Logic-based Calculus of Events. New Gener. Comput. 4(1): 67-95, 1986.

# 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.

Robert A. Kowalski, Marek J. Sergot: A Logic-based Calculus of Events. New Gener. Comput. 4(1): 67-95, 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 A., Sergot M. and Paliouras G., An Event Calculus for Event Recognition. In IEEE Transactions on Knowledge and Data Engineering (TKDE), 27(4), 895–908, 2015.

# 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 A., Sergot M. and Paliouras G., An Event Calculus for Event Recognition. In IEEE Transactions on Knowledge and Data Engineering (TKDE), 27(4), 895–908, 2015.

# Simple Fluent Specification

Definition:

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

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

where

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

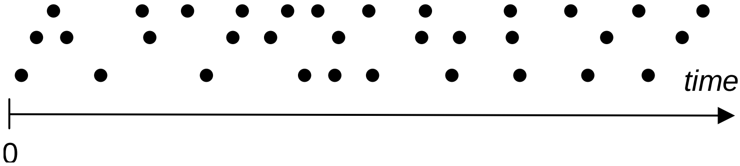
# Simple Fluent Computation

Definition:

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

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

Reasoning:



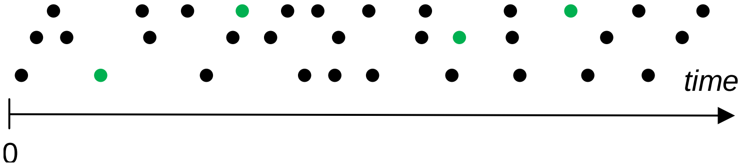
# Simple Fluent Computation

Definition:

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

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

Reasoning:



# Simple Fluent Computation

Definition:

$$\textbf{initiatedAt}(F = V,\ T) \leftarrow \\ \quad \textbf{happensAt}(E_{In_1},\ T), \\ \quad [\text{conditions}]$$

...

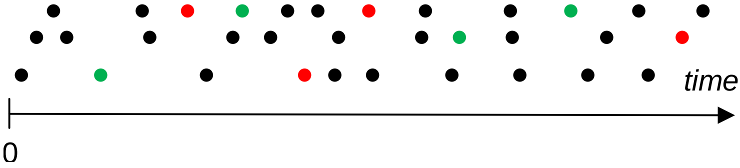
$$\textbf{initiatedAt}(F = V,\ T) \leftarrow \\ \quad \textbf{happensAt}(E_{In_i},\ T), \\ \quad [\text{conditions}]$$

$$\textbf{terminatedAt}(F = V,\ T) \leftarrow \\ \quad \textbf{happensAt}(E_{T_1},\ T), \\ \quad [\text{conditions}]$$

...

$$\textbf{terminatedAt}(F = V,\ T) \leftarrow \\ \quad \textbf{happensAt}(E_{T_j},\ T), \\ \quad [\text{conditions}]$$

Reasoning:



# Simple Fluent Computation

Definition:

$\textbf{initiatedAt}(F = V,\ T) \leftarrow$
$\quad \textbf{happensAt}(E_{In_1},\ T),$
$\quad [\text{conditions}]$

$\ldots$

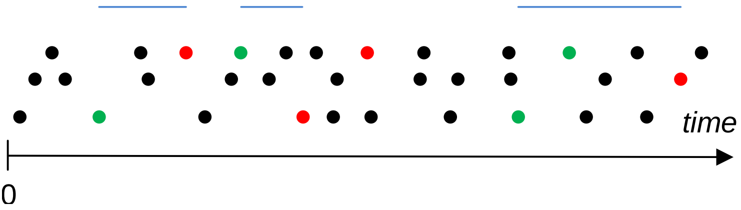
$\textbf{initiatedAt}(F = V,\ T) \leftarrow$
$\quad \textbf{happensAt}(E_{In_i},\ T),$
$\quad [\text{conditions}]$

$\textbf{terminatedAt}(F = V,\ T) \leftarrow$
$\quad \textbf{happensAt}(E_{T_1},\ T),$
$\quad [\text{conditions}]$

$\ldots$

$\textbf{terminatedAt}(F = V,\ T) \leftarrow$
$\quad \textbf{happensAt}(E_{T_j},\ T),$
$\quad [\text{conditions}]$

Reasoning: $\textbf{holdsFor}(F = V,\ I)$



# High Speed Near Coast

CE definition:

$$\begin{aligned}
&\textbf{initiatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \mathrm{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}),\ T), \\
&\quad \textbf{holdsAt}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \mathrm{true},\ T), \\
&\quad \mathit{threshold}(v_{hs}, V_{hs}), \\
&\quad \mathit{Speed} > V_{hs}.
\end{aligned}$$

# High Speed Near Coast

CE definition:

$$\begin{aligned}
&\textbf{initiatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}),\ T), \\
&\quad \textbf{holdsAt}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true},\ T), \\
&\quad \mathit{threshold}(v_{hs}, V_{hs}), \\
&\quad \mathit{Speed} > V_{hs}.
\end{aligned}$$

$$\begin{aligned}
&\textbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}),\ T), \\
&\quad \mathit{threshold}(v_{hs}, V_{hs}), \\
&\quad \mathit{Speed} \leq V_{hs}.
\end{aligned}$$

# High Speed Near Coast

CE definition:

$$
\begin{aligned}
&\textbf{initiatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}),\ T), \\
&\quad \textbf{holdsAt}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true},\ T), \\
&\quad \mathit{threshold}(v_{hs}, V_{hs}), \\
&\quad \mathit{Speed} > V_{hs}.
\end{aligned}
$$

$$
\begin{aligned}
&\textbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}),\ T), \\
&\quad \mathit{threshold}(v_{hs}, V_{hs}), \\
&\quad \mathit{Speed} \leq V_{hs}.
\end{aligned}
$$

$$
\begin{aligned}
&\textbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\text{end}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true}),\ T).
\end{aligned}
$$

# High Speed Near Coast

CE definition:

$$\begin{aligned}
&\textbf{initiatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}),\ T), \\
&\quad \textbf{holdsAt}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true},\ T), \\
&\quad \mathit{threshold}(v_{hs}, V_{hs}), \\
&\quad \mathit{Speed} > V_{hs}.
\end{aligned}$$

$$\begin{aligned}
&\textbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\mathit{velocity}(\mathit{Vessel}, \mathit{Speed}),\ T), \\
&\quad \mathit{threshold}(v_{hs}, V_{hs}), \\
&\quad \mathit{Speed} \leq V_{hs}.
\end{aligned}$$

$$\begin{aligned}
&\textbf{terminatedAt}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true},\ T) \leftarrow \\
&\quad \textbf{happensAt}(\text{end}(\mathit{withinArea}(\mathit{Vessel}, \mathit{nearCoast}) = \text{true}),\ T).
\end{aligned}$$

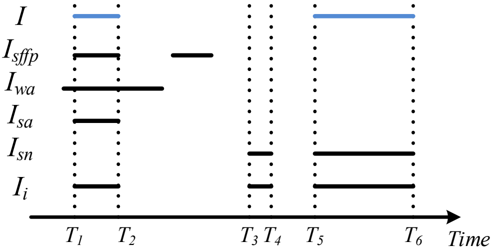
CE recognition: $\textbf{holdsFor}(\mathit{highSpeedNC}(\mathit{Vessel}) = \text{true},\ I)$

# Statically determined fluent: Anchored or Moored

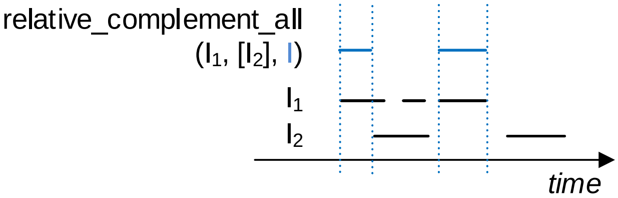
$$
\begin{aligned}
&\mathbf{holdsFor}(\mathit{anchoredOrMoored}(\mathit{Vessel}) = \mathrm{true},\ I) \leftarrow\\
&\quad \mathbf{holdsFor}(\mathit{stopped}(\mathit{Vessel}) = \mathit{farFromPorts},\ I_{sffp}),\\
&\quad \mathbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel}, \mathit{anchorage}) = \mathit{true},\ I_{wa}),\\
&\quad \mathbf{intersect\_all}([I_{sffp}, I_{wa}],\ I_{sa}),\\
&\quad \mathbf{holdsFor}(\mathit{stopped}(\mathit{Vessel}) = \mathit{nearPorts},\ I_{sn}),\\
&\quad \mathbf{union\_all}([I_{sa}, I_{sn}],\ I_i),\\
&\quad \mathit{threshold}(v_{aorm}, V_{aorm}),\\
&\quad \mathit{intDurGreater}(I_i, V_{aorm}, I).
\end{aligned}
$$

# Statically determined fluent: Anchored or Moored

$$
\begin{aligned}
&\textbf{holdsFor}(anchoredOrMoored(Vessel) = \text{true},\ I) \leftarrow \\
&\quad \textbf{holdsFor}(stopped(Vessel) = farFromPorts,\ I_{sffp}), \\
&\quad \textbf{holdsFor}(withinArea(Vessel, anchorage) = true,\ I_{wa}), \\
&\quad \textbf{intersect\_all}([I_{sffp}, I_{wa}],\ I_{sa}), \\
&\quad \textbf{holdsFor}(stopped(Vessel) = nearPorts,\ I_{sn}), \\
&\quad \textbf{union\_all}([I_{sa}, I_{sn}],\ I_i), \\
&\quad threshold(v_{aorm}, V_{aorm}), \\
&\quad intDurGreater(I_i, V_{aorm}, I).
\end{aligned}
$$



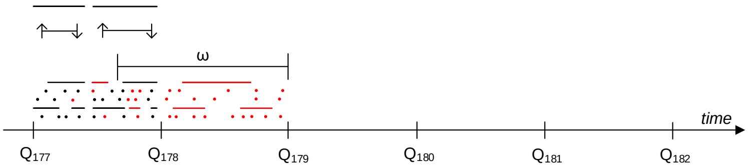
# Interval Manipulation: Relative Complement



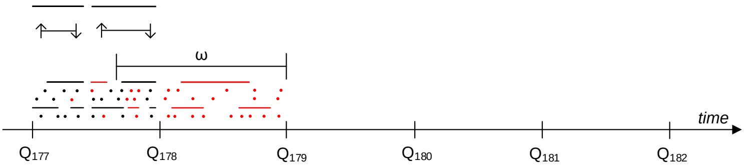
# Piloting

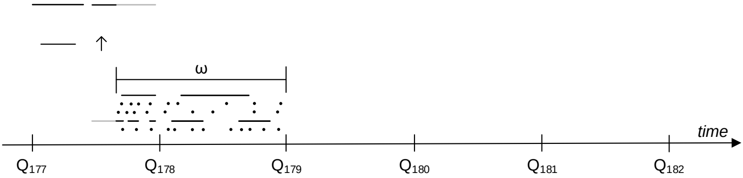
$$
\begin{aligned}
&\textbf{holdsFor}(\mathit{pilotBoarding}(\mathit{Vessel_1}, \mathit{Vessel_2}) = \text{true},\ I) \leftarrow \\
&\quad \mathit{oneIsPilot}(\mathit{Vessel_1}, \mathit{Vessel_2}), \\
&\quad \text{not } \mathit{oneIsTug}(\mathit{Vessel_1}, \mathit{Vessel_2}), \\
&\quad \textbf{holdsFor}(\mathit{lowSpeed}(\mathit{Vessel_1}) = \mathit{true},\ I_{l1}), \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel_1}) = \mathit{farFromPorts},\ I_{s1}), \\
&\quad \textbf{union\_all}([I_{l1}, I_{s1}],\ I_1), \\
&\quad \textbf{holdsFor}(\mathit{lowSpeed}(\mathit{Vessel_2}) = \mathit{true},\ I_{l2}), \\
&\quad \textbf{holdsFor}(\mathit{stopped}(\mathit{Vessel_2}) = \mathit{farFromPorts},\ I_{s2}), \\
&\quad \textbf{union\_all}([I_{l2}, I_{s2}],\ I_2), \\
&\quad \textbf{holdsFor}(\mathit{proximity}(\mathit{Vessel_1}, \mathit{Vessel_2}) = \text{true},\ I_p), \\
&\quad \textbf{intersect\_all}([I_1, I_2, I_p],\ I_f), \\
&\quad \textbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel_1}, \mathit{nearCoast}) = \mathit{true},\ I_{nc1}), \\
&\quad \textbf{holdsFor}(\mathit{withinArea}(\mathit{Vessel_2}, \mathit{nearCoast}) = \mathit{true},\ I_{nc2}), \\
&\quad \textbf{relative\_complement\_all}(I_f,\ [I_{nc1}, I_{nc2}],\ I_i), \\
&\quad \mathit{threshold}(v_{pil}, V_{pil}), \\
&\quad \mathit{intDurGreater}(I_i, V_{pil}, I).
\end{aligned}
$$

# Windowing

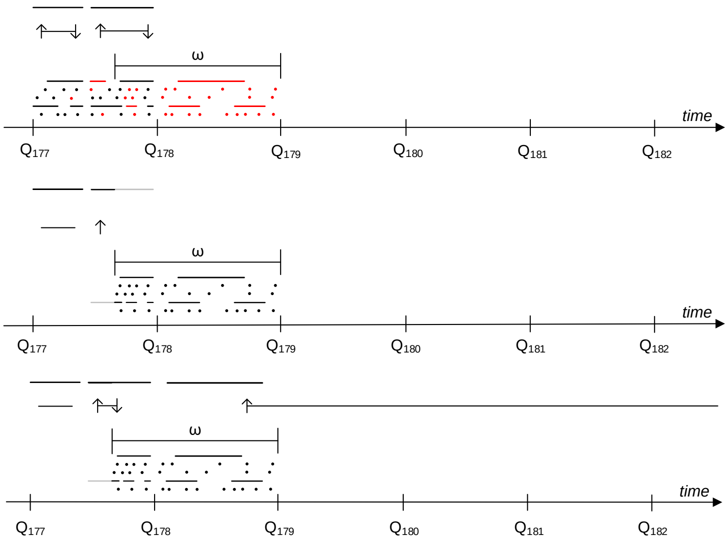


# Windowing


ω
time
Q177
Q178
Q179
Q180
Q181
Q182

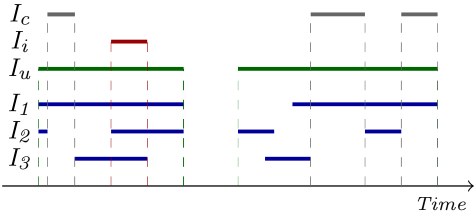

ω
time
Q177
Q178
Q179
Q180
Q181
Q182

# Windowing

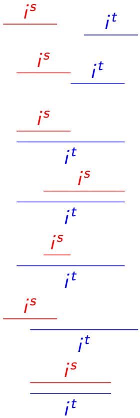

ω
time
$Q_{177}$
$Q_{178}$
$Q_{179}$
$Q_{180}$
$Q_{181}$
$Q_{182}$
ω
time
$Q_{177}$
$Q_{178}$
$Q_{179}$
$Q_{180}$
$Q_{181}$
$Q_{182}$
ω
time
$Q_{177}$
$Q_{178}$
$Q_{179}$
$Q_{180}$
$Q_{181}$
$Q_{182}$

# RTEC$_A$: RTEC with Allen Relations

**relative_complement_all**$(I_1, [I_2, I_3], I_c)$
**intersect_all**$([I_1, I_2, I_3], I_i)$
**union_all**$([I_1, I_2, I_3], I_u)$



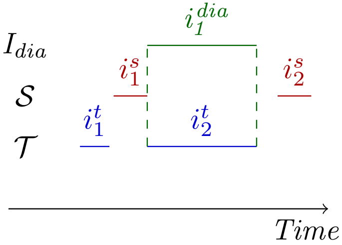

| Relation | Illustration |
|---|---|
| before$(i^s, i^t)$ | |
| meets$(i^s, i^t)$ | |
| starts$(i^s, i^t)$ | |
| finishes$(i^s, i^t)$ | |
| during$(i^s, i^t)$ | |
| overlaps$(i^s, i^t)$ | |
| equal$(i^s, i^t)$ | |




Periklis Mantenoglou, Dimitrios Kelesis, Alexander Artikis: Complex Event Recognition with Allen Relations. KR, 2023.

# Disappeared In Area

$$\textbf{holdsFor}(\textit{disappearedInArea}(\textit{Vl}, \textit{AreaType}) = \text{true}, I_{\textit{dia}}) \leftarrow$$
$$\textbf{holdsFor}(\textit{withinArea}(\textit{Vl}, \textit{AreaType}) = \text{true}, \mathcal{S}),$$
$$\textbf{holdsFor}(\textit{gap}(\textit{Vl}) = \textit{farFromPorts}, \mathcal{T}),$$
$$\textbf{allen}(\text{meets}, \mathcal{S}, \mathcal{T}, \text{target}, I_{\textit{dia}}).$$




Periklis Mantenoglou, Dimitrios Kelesis, Alexander Artikis: Complex Event Recognition with Allen Relations. KR, 2023.

# $\text{RTEC}_\text{A}$: Correctness & Complexity

**Correctness of $\text{RTEC}_\text{A}$**

$\text{RTEC}_\text{A}$ computes all maximal intervals of a fluent defined in terms of an Allen relation, and no other interval.

Periklis Mantenoglou, Dimitrios Kelesis, Alexander Artikis: Complex Event Recognition with Allen Relations. KR, 2023.

# $\text{RTEC}_\text{A}$: Correctness & Complexity

## Correctness of $\text{RTEC}_\text{A}$

$\text{RTEC}_\text{A}$ computes all maximal intervals of a fluent defined in terms of an Allen relation, and no other interval.

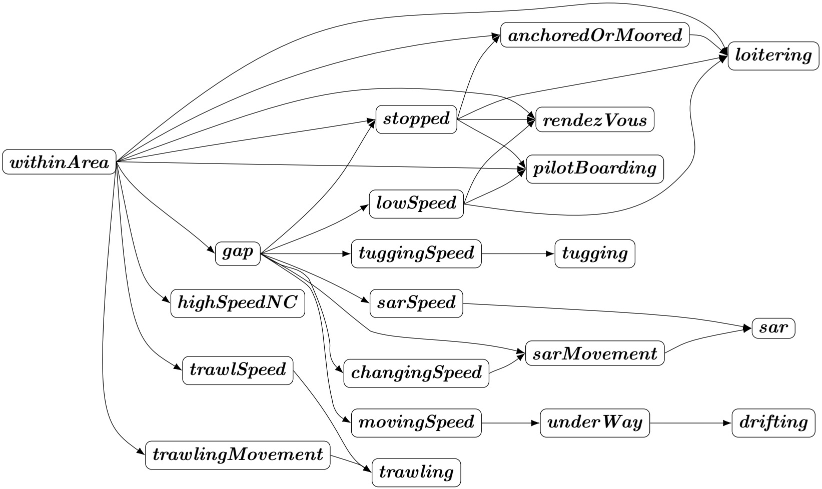
## Complexity of $\text{RTEC}_\text{A}$

The cost of computing the maximal intervals of a fluent defined in terms of an Allen relation is $\mathcal{O}(n)$, where $n$ is the number of input intervals.

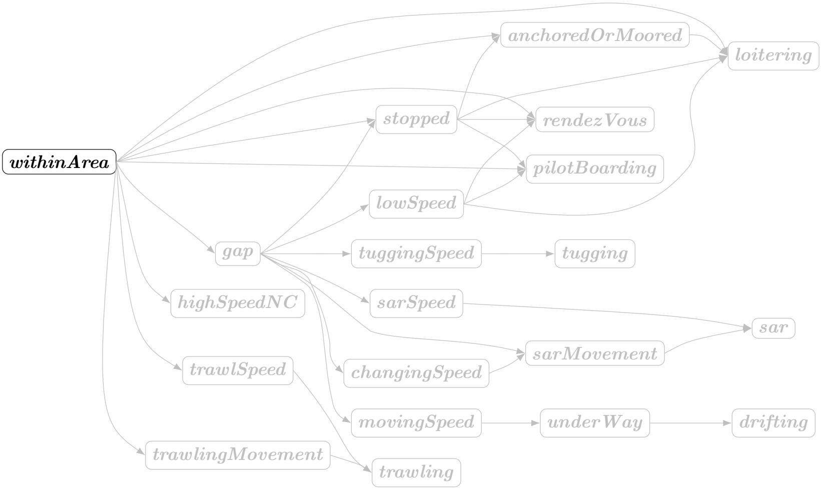
---

Periklis Mantenoglou, Dimitrios Kelesis, Alexander Artikis: Complex Event Recognition with Allen Relations. KR, 2023.

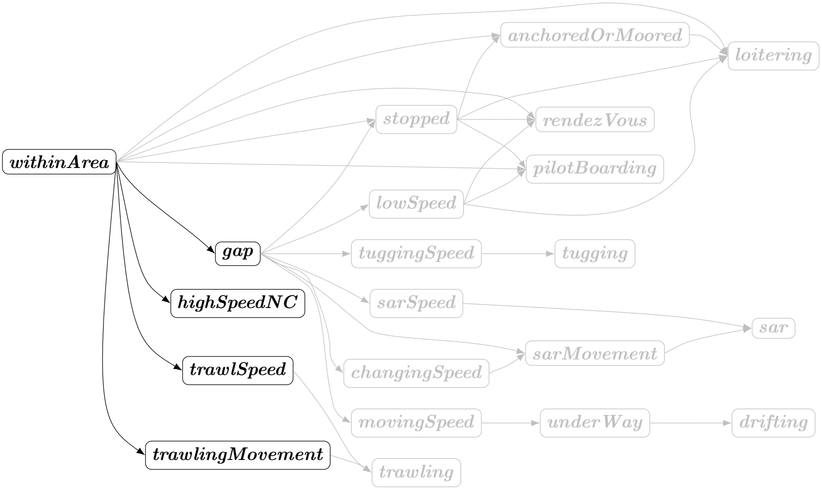
# Hierarchical Knowledge Bases: Maritime Situational Awareness



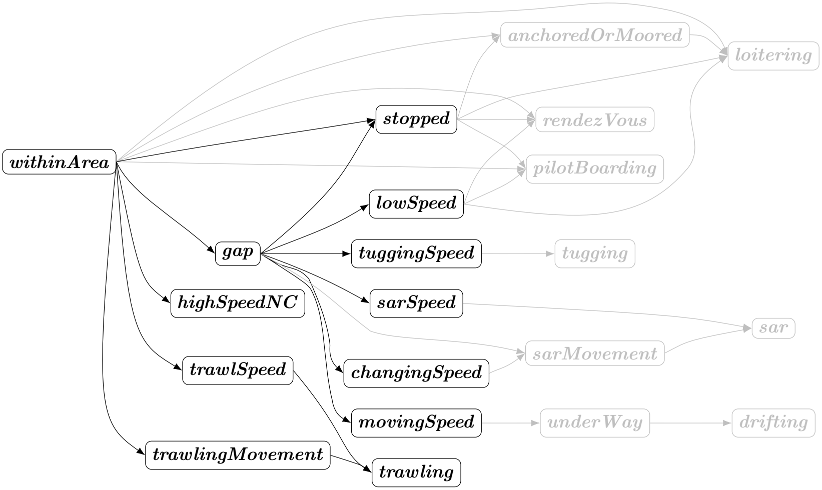
# Hierarchical Knowledge Bases: Maritime Situational Awareness



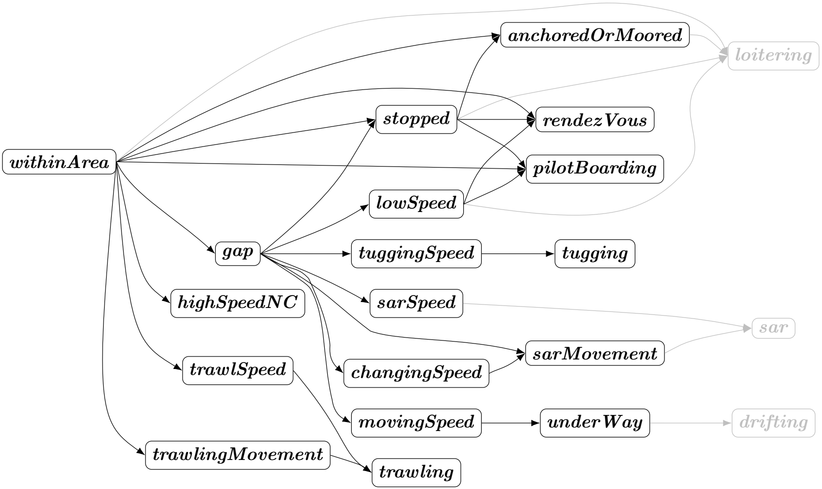
# Hierarchical Knowledge Bases: Maritime Situational Awareness



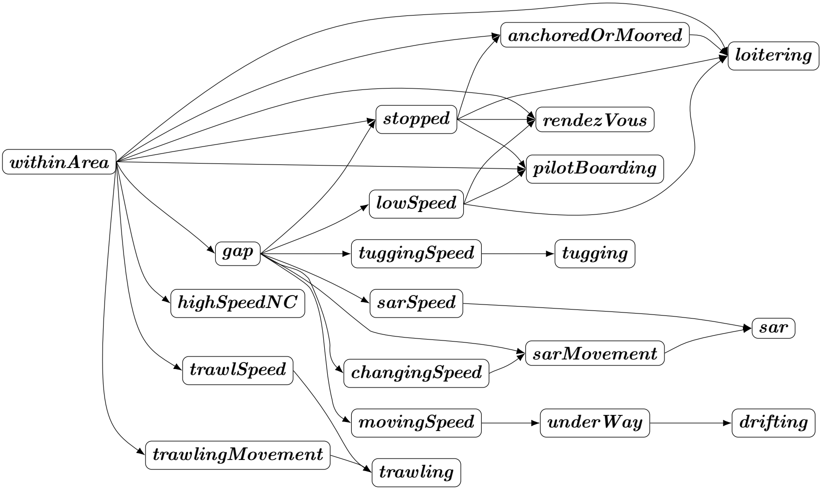
# Hierarchical Knowledge Bases: Maritime Situational Awareness



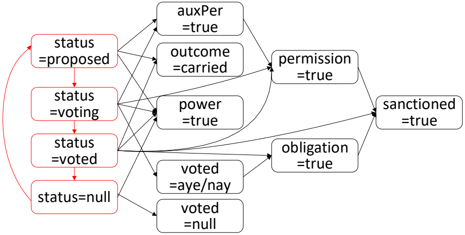
# Hierarchical Knowledge Bases: Maritime Situational Awareness



# Hierarchical Knowledge Bases: Maritime Situational Awareness



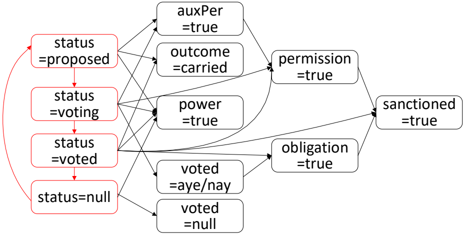
# Cycles in Knowledge Bases

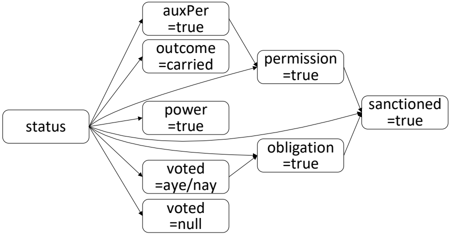



---

Periklis Mantenoglou, Manolis Pitsikalis, Alexander Artikis: Stream Reasoning with Cycles. KR, 2022.

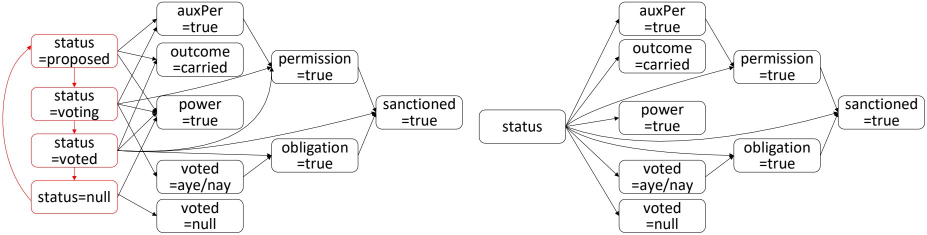
# Cycles in Knowledge Bases




Periklis Mantenoglou, Manolis Pitsikalis, Alexander Artikis: Stream Reasoning with Cycles. KR, 2022.

# Cycles in Knowledge Bases




## Semantics

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

Periklis Mantenoglou, Manolis Pitsikalis, Alexander Artikis: Stream Reasoning with Cycles. KR, 2022.

## Multi-Agent Systems: Voting & NetBill

- Compute normative positions of agents.

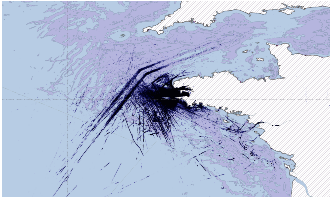
# Experimental Setup

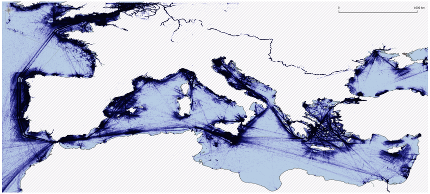
## Multi-Agent Systems: Voting & NetBill

- Compute normative positions of agents.

## Maritime Situational Awareness

- Recognise dangerous, illegal and suspicious vessel activity.





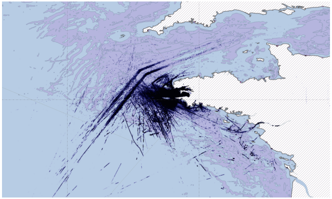
# Experimental Setup

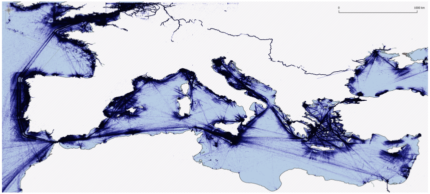
## Multi-Agent Systems: Voting & NetBill

- Compute normative positions of agents.

## Maritime Situational Awareness

- Recognise dangerous, illegal and suspicious vessel activity.



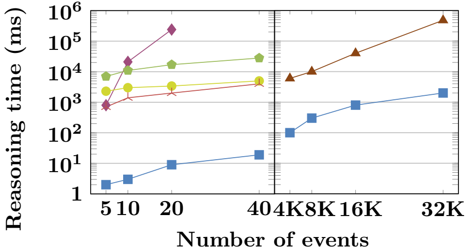


## Code, Data & Temporal Specifications

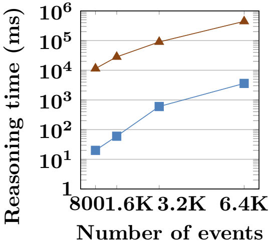
https://github.com/aartikis/RTEC

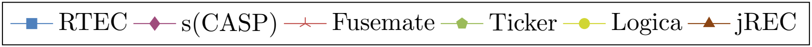
# Experimental Results
# Multi-Agent Systems

NetBill: no cycles

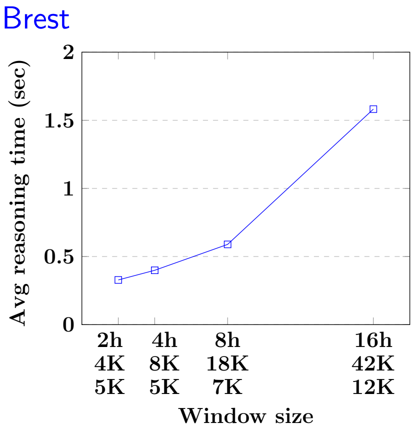


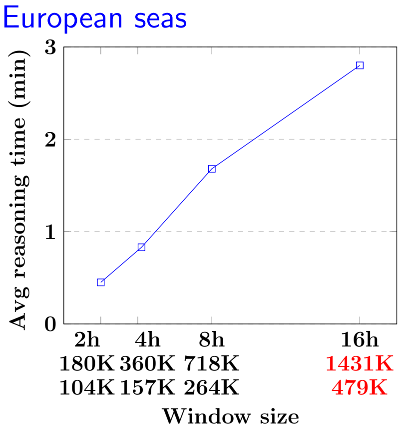

Voting: incl. cycles



# Experimental Results
# Maritime Situational Awareness



# Experimental Results
# CER with Allen relations

| Window size | | Reasoning Time (ms) | | Output Intervals | |
|---|---|---|---|---|---|
| Days | Input Intervals | $RTEC_A$ | $D^2IA$ | $RTEC_A$ | $D^2IA$ |
| 1 | 19K | **40** | 410 | 6K | 6K |
| 2 | 37K | **65** | 592 | 9K | 9K |
| 4 | 74K | **99** | 1.1K | 16K | 16K |
| 8 | 148K | **156** | 1.6K | 32K | 31K |
| 16 | 297K | **285** | 2.7K | 77K | 76K |

Code, Data & Temporal Specifications:

- https://github.com/aartikis/RTEC/tree/allen

# Summary & Further Work

## Summary

- Event pattern specification with the Event Calculus.
- Detection of the maximal intervals of composite events.
- Bottom-up computation, following the dependency graph.
- Support for Allen relations and cyclic dependencies.

## Further Work

- Support for events with delayed effects.
- Comparison with related frameworks w.r.t expressive power.
- Neuro-symbolic reaoning.

## Resources

https://github.com/aartikis/RTEC

https://cer.iit.demokritos.gr