

Rule-Based Composite Event Queries: The Language XChange^{EQ} and its Semantics

François Bry, Michael Eckert
LMU München



LMU

Motivation: Composite Events

Generating and reacting to events on the Web



Composite Events

- Must be inferred from “atomic events” (messages)
- Multiple atomic events, relationship between them
- Need query language!

LMU

Int. Conf. on Web Reasoning and Rule Systems 2007

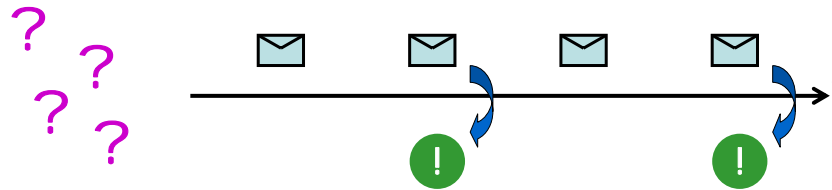
2

Queries against Event Streams

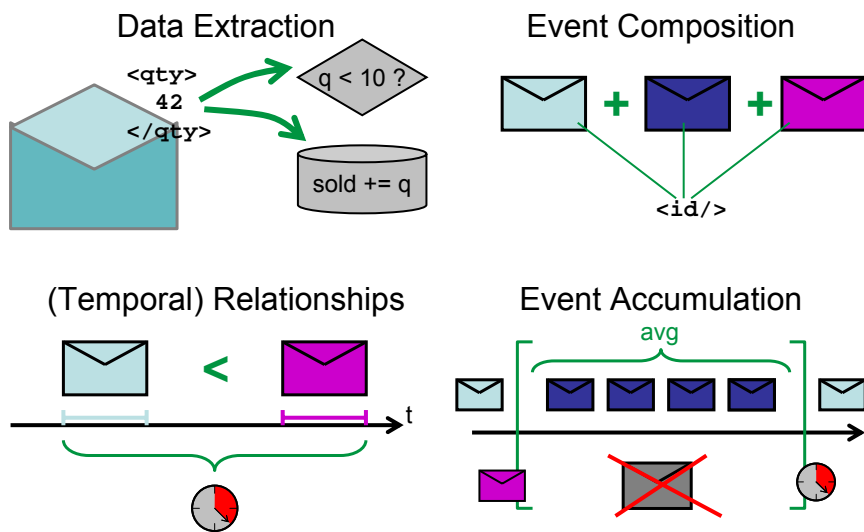
Database Queries, Web Queries:



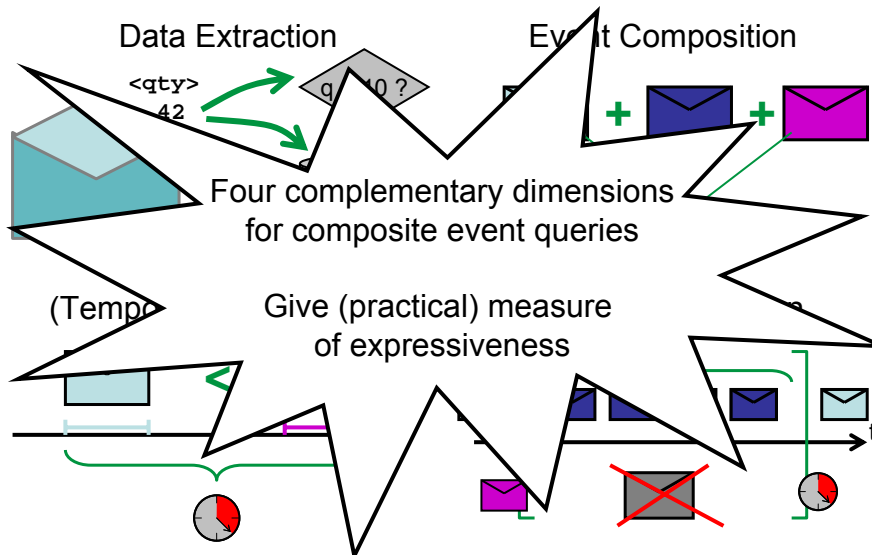
(Composite) Event Queries:



Language Requirements



Language Requirements



XChange^{EQ}: Rule-Based CEQs

- High-level, declarative query language for composite events, fully covers four dimensions
- Pattern-based queries on XML event messages: embeds Web query language Xcerpt
- Integrates into reactive rule language XChange
 - Perform automatic reactions, timing important
- Deductive (event) rules:
 - Define new, “virtual” events from received events
 - mediation, abstraction, reasoning (cf. database views)
 - Side-effect free; don't implement by reactive rules: optimization, (human) understanding

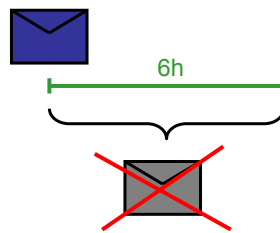
XChange^{EQ}: Example Rule

```

DETECT
  overdue {
    id { var ID }
    cust { var C } }
ON
  and {
    event o: order {{
      id { var ID },
      quantity { var Q },
      cust { var C } }}
    event w: extend[o, 6h],
    while w: not shipped {{
      id { var ID } }}
  } where { var Q < 10 }
END
    
```

```

DETECT
  order {
    id { var ID },
    quantity { var C },
    cust { var C } }
FROM
  ...
END
    
```



Semantics (1)

- Declarative Semantics for XChangeEQ: *model + fixpoint theories for stratified programs* (A standard approach for rule languages)

- (Tarski-style) model theory: Define $M \models F^t$ recursively

$I, S, r \models (\text{event } i : q)^t$ iff exists $e^i \in E$ with $r(i) = e^i, t^i = t_1$ and for all $e^j \in E, S(i)$ we have $e^j \neq e^i$.
 $I, S, r \models (\text{event } i : \text{extend}[j, d])^t$ iff exists e^i with $r(i) = e^i, r(j) = e^j, t^i = t^j = t'$ and $t' = t' + d$. (Definitions for other temporal events are similar and skipped).
 $M \models (p \wedge q)^t$ iff $M \models p^t$ and $M \models q^t$ and $t = t_1, t_2$.
 $M \models (p \vee q)^t$ iff $M \models p^t$ or $M \models q^t$.
 $I, S, r \models (Q \text{ where } C)^t$ iff $I, S, r \models Q^t$ and $M_{S, r}(C) = \text{true}$.
 $I, S, r \models (\text{while } j : \text{not } q)^t$ iff exists e^j with $r(j) = e^j, t^j = t_1$ and for all $e^i \in E$ we have $I, S, r \models q^t$.
 $I, S, r \models (\text{while } j : \text{collect } q)^t$ iff exists e^j with $r(j) = e^j, t^j = t_1$ and exist $n \geq 0, S_1, \dots, S_n, t_1 \leq t_2 \leq \dots \leq t_n$ with $S = \bigcup_{i=1..n} S_i$ and for all $i = 1..n$ we have $I, S_i, r \models q^{t_i}$.
 $I, S, r \models (e = Q)^t$ iff $(1) S^t(e) \subseteq Q$ for S^t maximal w.r.t. $\text{FreeVars}(Q)$ and r^t such that $I, S^t, r^t \models Q^t$, or (2) $I, S^t, r^t \not\models Q^t$ for all S^t, r^t .
 $M_{S, r}(i \text{ before } j) = \text{true}$ iff $\text{end}(r(i)) < \text{begin}(r(j))$.
 $M_{S, r}(i \text{ during } j) = \text{true}$ iff $\text{begin}(r(i)) < \text{begin}(r(j))$ and $\text{end}(r(i)) < \text{end}(r(j))$.
 $M_{S, r}(i \text{ overlaps } j) = \text{true}$ iff $\text{begin}(r(i)) < \text{begin}(r(j)) < \text{end}(r(i)) < \text{end}(r(j))$.

- Accommodates event identifiers (“event o:”)
- Events have occurrence times
- Temporal relations: fixed interpretation

Semantics (2)

- Restriction to stratified programs
 - w.r.t. negation, grouping, *relative temporal events*
- Fixpoint: model $M_{P,E}$
 - $T_P(I)$: all events derivable by rules in P
 - starting with incoming event stream E
 - compute fixpoints stratum by stratum
- Theorem:
 - P stratified program, E (incoming) event stream.
 - Then: $M_{P,E}$ is a minimal model of P under E and
 - Independent of the stratification of P

$$P = P_1 \uplus \dots \uplus P_n$$

$$T_P(I) = I \cup \{e' \mid \text{there exist a rule } e \leftarrow Q \in P,$$

a maximal substitution set $\Sigma,$
and a substitution τ s.t.
 $I, \Sigma, \tau \models Q'$ and $e \in \Sigma(e')$

$$T_P^* \text{; least fixpoint of } T_P$$

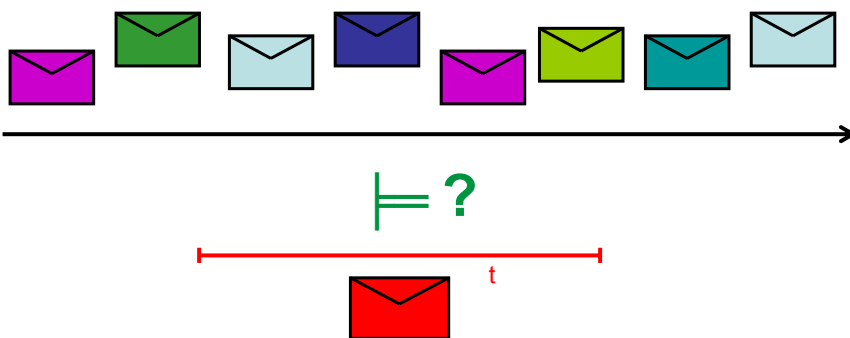
$$M_0 = E = T_P^*(E),$$

$$M_1 = T_{P_1}^*(M_0),$$

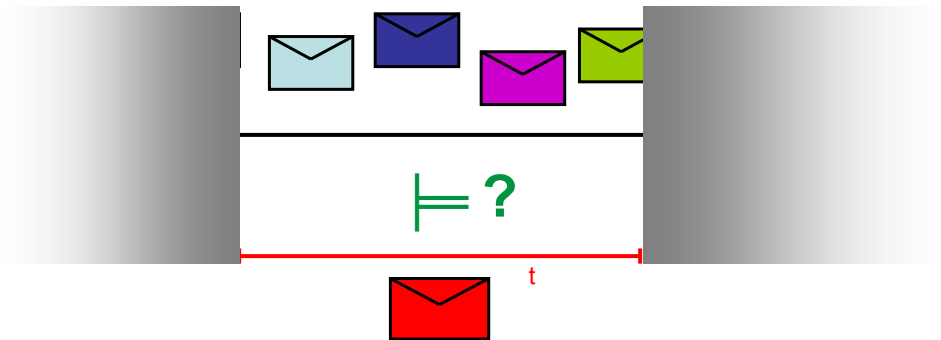
$$\dots,$$

$$M_n = T_{P_n}^*(M_{n-1}) =: M_{P,E}.$$

Unbounded Event Streams



Unbounded Event Streams



- XChange^{EQ}-Semantics are well-defined for *unbounded* (“infinite”) incoming event streams E

$$M_{P,E} \mid t = M_{P,E \mid t} \mid t$$

Summary and Outlook

- XChange^{EQ}:
 - High-level event query language
 - Full coverage of all four dimensions, XML support
 - Support for (deductive) event rules
- Declarative Semantics
 - Model and fixpoint theory for stratified programs
 - Well-defined on unbounded event streams
- Outlook
 - Incremental, data-driven evaluation
 - Optimizations based on temporal conditions

