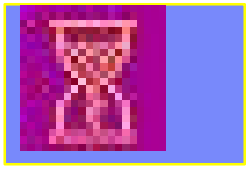
EDA Patterns

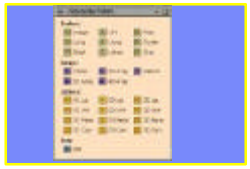
Event Representation

Structure Type

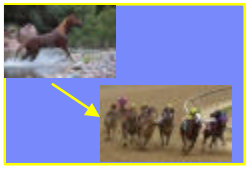
Chronon



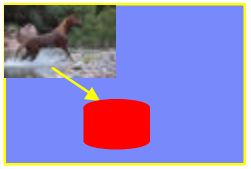
Attribute data type



Attribute entity reference

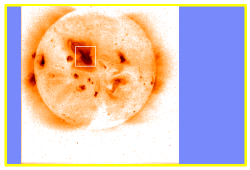


Attribute DB Reference

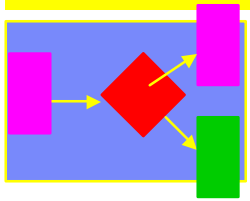


Event schema – properties and attributes

Complex Event



Conditional delegation



Temporal explicit causality

Event Relationships

Event Representation – Cont.

Event Identity

Event Source

Occurrence Time

Event Type

Con verse Events

Occurrence space

Detection Time

Event Common Attributes

Event Schema - type properties and attributes

- Structure type: the schema is one of the following structures: {flat, MRM, XML. RSS, Object}
- Chronon: Time granularity for "time point".
- Attribute data type: for each attribute, a data type for the attribute's representation
- Attribute entity reference: For an attribute, an entity that is referenced by this attribute
- Database reference: For an attribute a database record (tuple) that is referenced by this attribute

Event schema – event relationships

- Complex events is an event that may consist of other events.
- Conditional delegation: if the predicate is satisfied,

Event processing context

- The players in the decision if an event is required to be processed by a certain mediator
- In the most general sense:
  - Time interval window +
  - Space coordinates window +
  - Partition of the universe of events (e.g. there is a context for every customer).

Event Derivation

Context

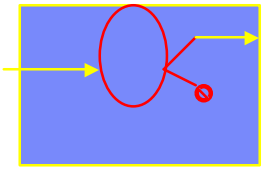
Relevant time

Relevant space

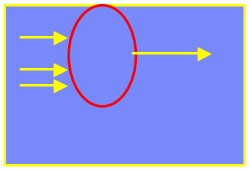
Partitioning entities

Event Pattern

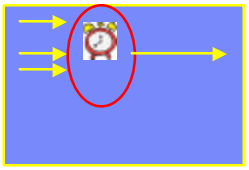
Filtering



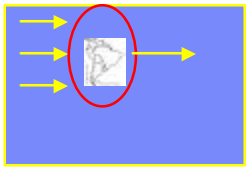
Pattern matching



Temporal oriented Pattern matching

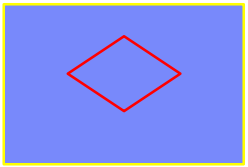


Spatial oriented Pattern matching

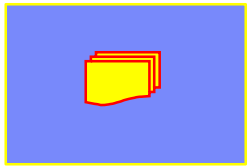


Predicate

Predicate

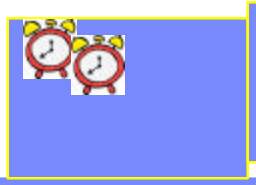


Set Condition

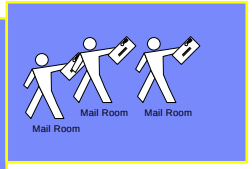


Policies

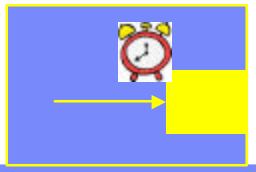
Overlapping contexts



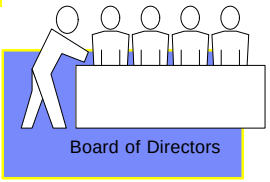
Handling synonyms



Processing Invocation



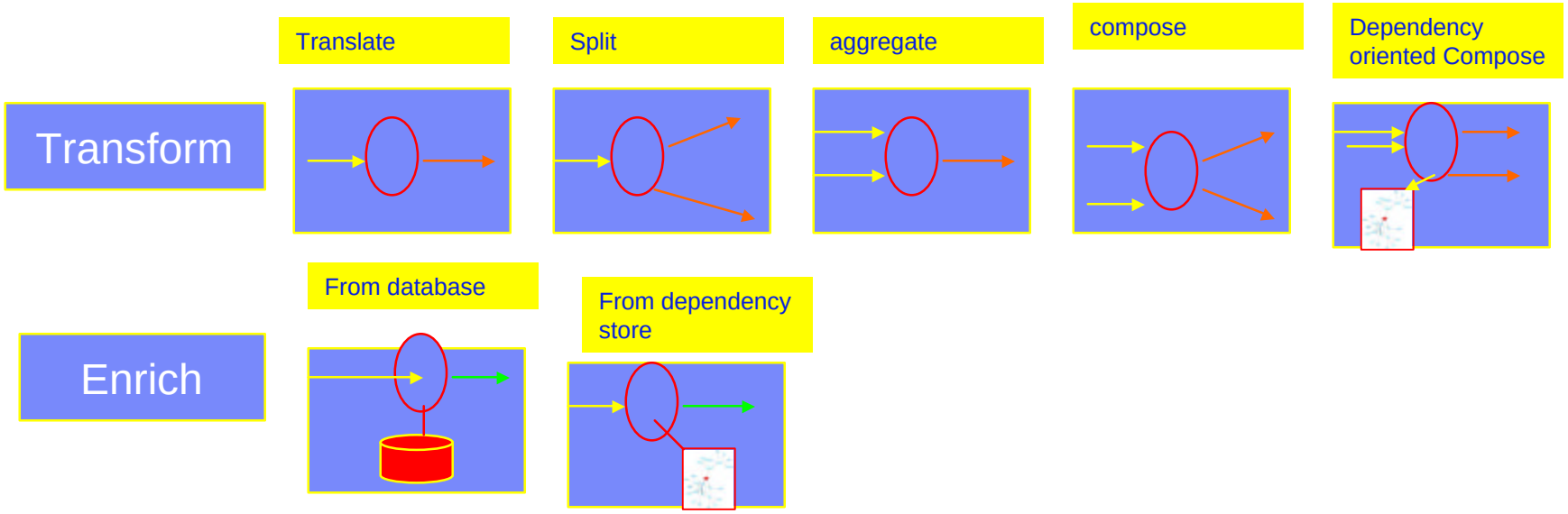
Directives



Event Derivation

- A pattern is a combination of all four:
- Context : Determine the boundaries of a single selection
  - Relevant time: events that occur in this time interval are assessed for the same selection.
  - Relevant space: events that occur in this space coordinates are assessed for the same selection
  - Partition entities: events that refer to this entity or to these combination of entities are assessed for the same selection
- Event Pattern: The pattern on history of events
  - Filtering: stateless single event decisions
  - Pattern matching: logical operator on events set
  - Temporal oriented matching: temporal operators on events set
  - Spatial oriented matching: spatial operators on events set
- Predicates
  - Logical connection predicates
  - Set oriented predicates
- Policies
  - Overlapping contexts: what to do when there are overlapping instances of a single relevance interval
  - Handling synonyms: what to do when there are multiple instances of the same event type
  - Processing Invocation: when to invoke the processing component
  - Directives: tuning of design decisions.

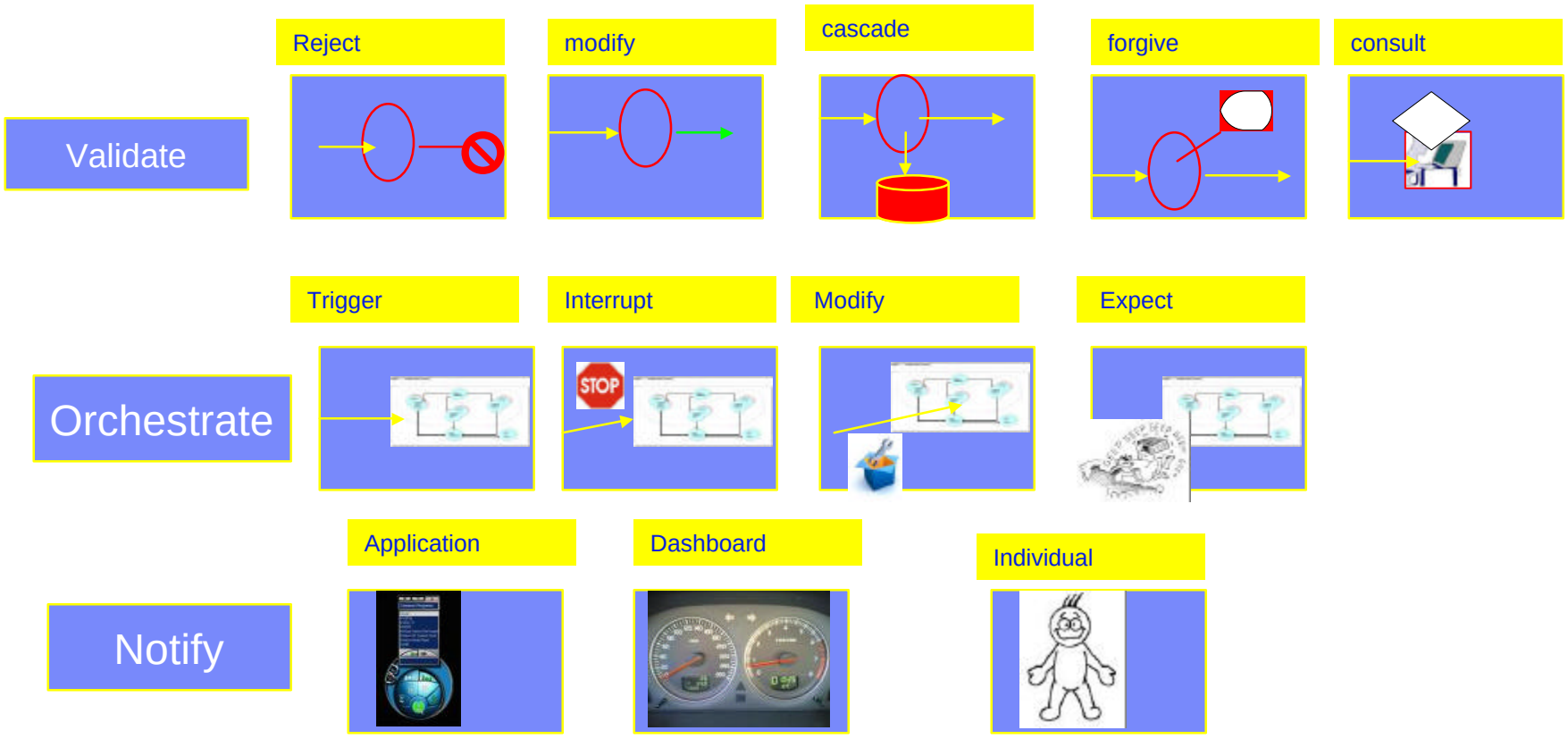
Event Transformation



Transformation

- Transform
  - Translate : 1 input event to 1 output event
  - Split: 1 input event to N output events
  - Aggregate: M input events to 1 output event
  - Compose: M input events to N output events
- Enrich: Create new event based on old event + additional attributes
  - From Database
  - From dependency store

Event-Driven Activities



Event-driven activities

- Validate: the selected events participate in constraint violation. Validate include stabilizing policies that can (conditionally) mitigate this violation
  - Reject: the causing event is filtered out, no output event from this processor
  - Modify: the causing event is being modified.
  - Cascade: correction of past event or modification of a database to eliminate the constraint violation
  - Forgive: pass the causing event, record the violation
  - Consult: consult either a human agent or an automatic decision system
- Orchestration:
  - Trigger: new workflow instance
  - Interrupt: existing workflow instance
  - Modify: existing workflow instance
  - Expect: Inject external event into control flow