Web services, choreography and the Event Calculus

A short introduction
History

- Invented by Kowalski & Sergot (1986)
- Many variants
- Shanahan’s tutorial paper
  - [http://casbah.ee.ic.ac.uk/~mpsha/pubs.html](http://casbah.ee.ic.ac.uk/~mpsha/pubs.html)
- He uses it for robotics, reasoning about change
- Explored by Sloman for use in expressing policies
Event Calculus concepts

- Events
  - Something happened
- Fluents
  - Something is true of the world
- Time
  - Partially ordered set of time points
Event

- Occurs at a point in time
- May initiate the truth of a fluent
- May terminate a fluent
- May ‘release’ a fluent

(The truth of a released fluent is no longer known)
A logical formula whose truth (or falsity) has an extent in time

- Fluents are given concrete names
  - I.e., are first class entities
- Fluents are initiated, terminated by events
- Fluents may hold at a particular time
- Fluents can be clipped, released
EC Logic

- We can reason about what is true
- Uses standard predicate logic with a few extras - closed world assumption/ circumscription
An order

Event times may be partially ordered
Simple axioms of EC

HoldsAt(f,t) <- Initially(f), ¬Clipped(0,f,t).

HoldsAt(f,t) <- Happens(a,t1), Initiates(a,f), t>t1, ¬Clipped(t1,f,t).

Clipped(t1,f,t2) <- Happens(a,t), t1<t<t2, Terminates(a,f)
Simple Order logic

- Happens(sendOrder,T0).
- Happens(acceptOrder,T1).
- Happens(sendCash,T4)
- Initiates(acceptOrder,orderAccepted).
- Initiates(sendCash,orderCompleted).
Role in choreography

- Complements pi-calculus view
- not particularly sensitive to process abstraction
- External state modeled as a set of fluents
- Transitions/messages are marked by events
- Permits expression of verification conditions
- Permits expression of policies