

On Defining and Capturing Events in Business Communication Systems*

Alan S. Abrahams

David M. Evers

Steven O. Kimbrough

asa28 (à) wharton.upenn.edu

David.Evers (à) cl.cam.ac.uk

kimbrough (à) wharton.upenn.edu

21 April 2006

*File: flbc-implementation-foils.tex/pdf.

Abstract

File available at: <http://opim-sky.wharton.upenn.edu/~sok/sokpapers/2007/flbc-implementation-foils-20060422.pdf> Firms have in large part discovered the need for maintaining detailed records of inventory. Knowing what was (is/will be) where and when has proved to be of great value for managing an enterprise. The work we report on in this talk is motivated by a similar presumption with regard to computerized business communication (e.g., EDI messages, messages for administration and control, etc.). We sketch a unified account of messaging in terms of the theory of event semantics, which we integrate with the event calculus for purposes of reasoning about events. Event semantics and the event calculus had entirely different origins (one came from philosophy and linguistics, the other from computer science; they were developed for different purposes). We discuss how each can be generalized to fit into a coherent and, we submit, very useful whole. In support of this claim we present and discuss a working prototype that embodies these ideas.

What are we talking about?

- FLBC research project.
- Formal Languages for Business Communication
- Aim: To develop and ready for fielding a fully formal, broadly logical theory for representing business messaging and for reasoning about business messages.

Interested both in underlying theory (interesting and quite challenging) and in implementation and use. This talk: focus on current development and implementation efforts, with many side comments on the larger principles involved.

Business messaging?

- Some examples:

<http://opim-sky.wharton.upenn.edu/~sok/flbc/messagetypes/>

Just the As: Accept(ance), Acknowledge(ment), Agree(ment), Announcement, Apology, Application, Assignment, Authorization

- Especially: purchase orders, shipping notices, RFQs, invoices, payments, bills of lading, ...all the application-level messages needed to conduct business.
- Effective automation promises to save time, money, improve quality, etc. Effective automation also, we believe, requires formalization and an adequate general theory.

Hasn't this already been done? EDI, XML and all that.

No.

Although, these solutions are in use (in the case of EDI, have long been in use) and are under active development, it is clear that the technology employed has severe limitations and shortcomings. There certainly is no underlying theory.

This finding can be defended in detail and at length, but it is not the subject of this talk.

But if you are proposing something that is radically different, how do you hope to get it accepted and used?

Could we please get on with the presentation? Points arising:

1. SMEs, nationally and internationally are largely orphaned by current systems. First trade problem. We can compete with nothing.
2. As we hope will become apparent, there should be considerable advantages to the framework or approach we are advancing. Also, it is entirely possible to map between the FLBC framework and an edi-style representation, so that representation could be retained for transport purposes, but with loss of functionality.

Is there an overarching theme or idea that undergirds your approach and that would help us get a feel for what you are doing?

We're very glad you asked that question. Two such ideas or themes.

1. The distinction between communication (of meaning) by reference and by articulation.
2. The turn to events.

Reference versus articulation

- Joke club joke
- At one extreme (of communication by reference) every message has a unique ID and we simply send that ID. Works well if there are only a few different messages and they rarely change. Works poorly if there subtle shades of meaning need to be expressed or if the message types need to change very much.

Present-day edi protocols (X12, EDIFACT, ebXML, etc.) are essentially parameterized reference devices, parameterized in rather simple ways.

Reference versus articulation

- A message is articulated if it is regularly composed from atomic (or subatomic) elements whose meaning can be determined directly.

Example: The Bob and Carol principle:

If you want to say “Bob and Carol”, use a representation that conjoins “Bob” and “Carol”. Specifically, don’t create a new expression, e.g., “BobCarol”, to refer to the conjoining of Bob and Carol. Articulate the composed meaning by composing it from its primitive elements.

FLBC aims to articulate as much as possible (and practicable).

The turn to events

The term *event* appears in the information sciences often, in many different contexts, and with several rather distinct senses. In particular:

1. Event logging. Used to manage Web sites, computer systems, etc.
Closely related: journaling in database.
2. Event-based messaging/middleware, etc. as in, e.g., publish-subscribe systems.
3. Event calculus. A formalism developed by Kowalski and Sergot for reasoning about states that come and go with time and, generally, for presenting a principled response to the frame problem in AI.
4. Event semantics. A body of theory for explicating meaning in natural language. Used and extended in FLBC to model business messaging (sok, asa and others).

Our thesis or presumption

- These are each good ideas.
- Although the notion of event is historically different, it is possible to generalize and to organize the three ideas together in a common and coherent system.
- Not only is it possible, it is desirable, and that's what we're doing. Details to follow (very soon).

Intuition tickler: FLBC message with event semantics

UBS requests that Citibank credit the account of Bruce Weber in the amount of 2500 US dollars at today's rate of exchange.

request(ubs:1) ∧ Speaker(ubs:1, UBS) ∧ Addressee(ubs:1, CITI) ∧ Theme(ubs:1, ubs:2) ∧ Benefactive(ubs:1, 'Beat Schmid') ∧ address('Beat Schmid', '....') ∧ reason(ubs:1, 'HONORARIUM200608') ∧ Cul(ubs:1, 20060901) ∧ Content(ubs:1, [credit(ubs:2) ∧ Agent(ubs:2, CITI) ∧ Benefactive(ubs:2, '12-345-67') ∧ Theme(ubs:2, ubs:6) ∧ currency(ubs:6) ∧ unit(ubs:6, 'USD') ∧ quantity(ubs:6, 2500) ∧ name('12-345-67', 'Bruce Weber') ∧ address('12-345-67', ['WASHINGTON SQUARE', 'NEW YORK, NY 10000'])])

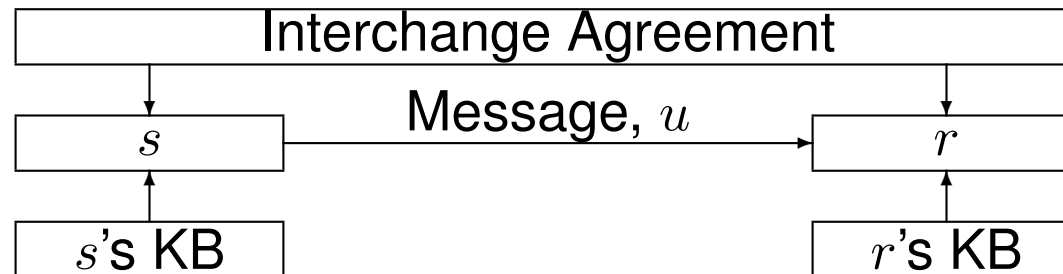
Intuition tickler: Event calculus

Axiomatization of inference rules on logs of historical events. For example, a state, or fluent, F , holds at, is true at, time T if at the start of things it held and since has not been terminated, or if it held at some time before T and was not terminated in the interim.

```
holdsAt (F=V, T) :-  
    initially (F=V), not broken (F=V, 0, T).  
holdsAt (F=V, T) :-  
    happens (Act, TT), TT < T, initiates (Act, F=V, TT),  
    not broken (F=V, TT, T).
```

This is valid Prolog.

Setup: Messaging framework



A message u from speaker s to addressee r . Sender: s . Addressee: r . knowledge base: KB.

Interchange agreement: Contract that specifies world–symbol relationship. Also, may (should) contain formalized rules, e.g., regarding promising, and other ‘terms of trade’ conventions or agreements.

Individual KBs: Include event calculus (and other) rules for reasoning about messages.

And now...

David will tell us about the implementation.

References

- [KT00] Steven O. Kimbrough and Yao-Hua Tan, *On lean messaging with unfolding and unwrapping for electronic commerce*, International Journal of Electronic Commerce **5** (2000), no. 1, 83–108.