On Lean Messaging with Wrapping and Unfolding for E-Commerce

Steven O. Kimbrough
University of Pennsylvania, 3620 Locust Walk, Suite 1300, Philadelphia, PA 19104-6366
Kimbrough@Wharton.upenn.edu, (215) 898-5133, http://opim.wharton.upenn.edu/~sok/

Yao-Hua Tan
EURIDIS, Erasmus University Rotterdam, Burgemeester Oudlaan 50, 3062 PA Rotterdam, The Netherlands, ytan@fac.fbk.eur.nl, +31 10 408.2255, http://www.euridis.fbk.eur.nl/Euridis/

Abstract
EDI (electronic data interchange) messages are notoriously lean and difficult to interpret without additional information. While sympathizing with the many criticisms that have been made of the design of EDI protocols, we argue in this paper that there is something basically correct in the noted leaniness of EDI messages. We present a framework that describes how interpretation of EDI messages works, and indeed must work. Central elements of this framework are what we call wrapping and unfolding of messages. We demonstrate how to exploit these concepts in formalizations for electronic commerce. In particular, we show how Kimbrough's event semantics for speech acts and Tan's theory of directed obligation can be fit naturally and fruitfully into this framework, and to each other. Much work remains to be done, but the progress in formalization in evidence here should be generalizable.

1 Context

Most observers of, and indeed most participants in, EDI would agree that present technology and standards for computer-to-computer exchange of structured information are importantly, even fundamentally, deficient. Costs are too high, the first-trade problem is daunting and discourages SMEs (small- and medium-sized enterprises) from participating in EDI, too much manual intervention is needed even for well-established systems, EDI protocols are not suited to the Web, there is no semantics for existing protocols (or only incomplete and confused ones), and so on (cf., [12, 16, 30] among many others). All this and more is part of the received wisdom today, and we largely agree.

What to do? The usual move in Information Systems when faced with a puzzling application problem is to structure the application domain so that it can more effectively be computerized. A formal theory, including a formal semantics, would seem to be in order. Why?

(1) Computerization requires that the application domain be structured and formalized. This may occur explicitly, in the form of a theory, or implicitly, in the minds of the programmers. Clearly, the explicit route is preferable.

(2) Explicit structuring and formalization allows us in an important sense to know what we are doing. A formal semantics for a domain gives us insight and understanding, which we may use to revise and apply our theory. It allows us to specify and test requirements independently of a particular implementation. For example, a logic model of a purchase order can be examined and tested for the inferences it supports, whether or not there is any implementation. This provides ways of validating specific implementations.

(3) Formalization and structuring also facilitate extending our representations and designing rules for generating and processing messages. We shall illustrate this in the sequel. For now, we simply note that structuring and formalization ought to help with interpreting EDI messages but also with the rules and policies that do the interpreting.

Again, these are widely-agreed upon thoughts with which we agree. But how to proceed from here? Dif-
different people will have—do have—different approaches to structuring and formalizing the EDI domain. Some favor a logic-based approach (e.g., Kimbrough, Scott Moore, Michael Covington). Others, one based on XML (e.g., [17, 30]), Petri nets (Ronald Lee and his group at EURIDIS), or some other formalism. And within a given formal approach there will no doubt be different substantive approaches.

Besides the call for formalization and rigor, are there any other principles that can be invoked to guide these investigations? We think so and our main purpose in writing this paper is to present and argue for certain such principles, which we call lean messaging, wrapping, and unfolding. In brief, we believe that existing EDI systems embody these principles already and for good reasons. Even with improved messaging standards they should remain in force. In fact, a main reason for such improvements is to avail EDI systems of the benefits of these principles.

2 Lean Messaging in EDI

Consider in the abstract how EDI messaging is effected. See Figure 1, the basic EDI messaging schema. When two organizations, s and r, wish to do business with EDI they begin by forming an interchange agreement. This is a contract which, among other things, specifies rules of trade between s and r, including which EDI protocols and transaction sets are to be used, and how EDI messages are to be interpreted. Once this contract is in place the sender of a message can draw upon its own knowledge bases and the interchange agreement to formulate a specific message, u (utterance) in the figure. Similarly, the receiver of a message draws upon its own knowledge bases and the interchange agreement in order to interpret and act upon the incoming information. We now offer some reflections on this arrangement.

By virtue of this schema a certain leanliness attends EDI messaging. In fact, one of the characteristic features of Electronic Data Interchange (EDI) messages is their unintelligibility without substantial reference to context. EDI messages present data structures that are basically stripped down, with nothing explaining the relations among the data elements constituting the message. For example, in the Purchase Order EDI message only a code in the header indicates that it is a data structure for representing purchase order information. The types of information given in this data structure are names and addresses of buyer, seller, involved banks for payment of the goods, payment method, delivery terms, and so forth. But in the EDI message there is no explanation how these different roles relate to each other. For example, the information that the seller is the one who delivers goods to the buyer, and that the buyer is the one who pays for the goods is not (explicitly) represented in the EDI data structure.

There are several reasons for this minimal representation of information in EDI messages. The first reason is of course efficiency of data transfer. EDI messages are supposed to be transferred as electronic messages via electronic networks. The fewer bits such a message contains the faster it goes. A second consideration is that EDI messages are international standards. By only providing the bare data and omitting textual explanation about the relations between the data, the problem of translation from one language, or legal system, to another is reduced.

However, by omitting this information, the understanding and processing of EDI messages becomes a very elaborate exercise. EDI is not only about electronic transfer of messages, but also (or perhaps even

\[2\text{At least this is the case in both the X12 and EDIFACT standards. It is also true of the S.W.I.F.T. protocols.}\]
more!\) about computer mediated automated processing
of these messages. Automated processing of an EDI
purchase order by the computer of a vendor means that
the computer is able to store the name of the buyer
in its own data base as the buyer of the goods. Au-
tomated processing also means that the computer is
not only able to store the incoming information at the
proper place, but that it can also initiate further ac-
tion. A very simple example is that the computer sends
information to the production department that is rele-
vant for the production of the requested goods, sends
an advanced shipment notification to the carrier for ship-
ment of the goods, and sends automatically an invoice
to the buyer after the goods are shipped. Depending
on the terms of delivery the computer might even have
to arrange insurance of the goods for the transport.

Currently, this automated processing, if at all avail-
able, is hard-wired into the company’s information sys-
tem. So-called EDI translator software translates in-
coming messages for the companies information sys-
tem. This hard wired solution is not only very expen-
sive, but also very inflexible and inefficient [23, 25].
Usually, the way this is hard wired is idiosyncratic.
The more companies will process EDI automatically
the harder it is to find EDI translators that can translate at a higher
level of abstraction. A semantic model of an EDI
message gives information about the relations among the
data in an EDI message. This information can be used
for processing of the message, and also as intermedi-
ary for communication between two EDI translators
of different companies. An example of this semantic
model approach is eCoSystem [28]. The eCoSystem
is developed to enable interoperability of the different
software environments for electronic commerce. Each
of these environments contains a module for electronic
payments, but they are all based on different standards.
To make these payment systems interoperable an
underlying semantic model of electronic payment needs
to be defined, from which translations are defined to
every specific payment module.

An additional benefit of a semantic model of EDI
messages is that it can be used for developing on-line
support tools for human users. The model could be
used to explain content of EDI messages, which are
now unintelligible for the human reader, because only
the data and not the relations are represented. Gener-
ating a semantic model for a EDI message is what we
call meaning unfolding. In meaning unfolding relations
are added to the bare data of, e.g., an EDI message.
For example, in an EDI purchase order, data are given
about names and addresses of buyer and seller, and
about quantities and types of goods purchased, but it
is not explicitly stated that the buyer is responsible for
payment for the goods. This payment by the buyer for
the benefit of the seller is a relation that is added in
the meaning unfolding of a purchase order.

In addition to this meaning problem of EDI messages
there is second problem, namely the legal inferencing
problem, i.e., the problem of inferring the legal impli-
cations of an EDI message. These legal implications
also include the obligations, permissions and rights that
hold for the parties mentioned in an EDI message. First
of all EDI messages are only legally binding between
partners that have signed a so-called interchange agree-
ment [25]. Without such an agreement the EDI mes-

sages have at best murky legal implications and hence
might not create obligations for agents mentioned in
the message. Secondly, in the interchange agreement it
has to be stipulated under which country’s jurisdiction
the agreement resorts. This also determines which legal
implications obtain for the message. Thirdly, the
legal implications of an EDI message might vary from
country to country. If a specific EDI message is legally
binding in country A, but not in country B, i.e., non-
liable, then only in country A does the promise
stated in the message imply an obligation. Fourthly,
one could even argue that inferring the legal impli-
cations of an EDI message is always defeasible even
within a country where the contract is valid, because
there might be exceptional circumstances that override
these implications. Examples of such exceptional cir-
cumstances are insanity or being under age of the act-
ing agents, or an export ban of the government that
legally forbids the execution of the purchase order. All
these examples clearly show that it is far from obvious
whether legal implications can be inferred from a particu-
lar EDI message, and hence which obligations hold for
the parties mentioned in an EDI message.

The meaning problem and the legal inferencing prob-
lem can be analysed in terms of speech act theory. In
this we agree with the received view in agent commu-
nication theory [13, 22, 24]. The basic idea is to illumi-
nate the job of the receiver, \( r \), of a message (see Figure
1). Suppose, to take a simple case (but one that gen-
eralizes), the utterance, \( u \), is a promise. We need to
distinguish three aspects of this case of promising:
(1) The temporal event of making a promise.
(2) The semantic meaning of promise.
(3) The fact that under specific circumstances the mak-
ing of a promise implies an obligation for the promiser to keep his promise.

A promise \( \{1\} \) is an event occurring in time and thus must be represented by a referring (or naming) expression. The (core) meaning of a promise \( \{2\} \) is a statement, made true by the (felicitous) event of making the promise. This statement is the semantics for the message sent in making the promise, \( u \), and is arrived at by a process we call unfolding (or translating or unpacking) the message \( u \). The (extended) meaning of a promise is what can be inferred from the core meaning plus additional assumptions, which we call the wrapping. In Figure 1 the wrapping for the message recipient, \( r \), consists of the interchange agreement plus \( r \)'s knowledge bases. In short, unfolding is generating a semantic model of the EDI message. Inevitably this requires going to a more detailed level of explanation. It is the wrapping that requires (and rewards) going to a more abstract level, allowing one message to get different interpretations depending on the legal system by which it is processed, for example. The difference between meaning unfolding and wrapping can be illustrated as follows. Meaning unfolding of an EDI purchase order is adding, for example, the information that the buyer is responsible for payment of the goods. However, this responsibility only implies an legal obligation for the buyer if the purchase order is legally binding, which depends on the interchange agreement. And this legal inferencing has its basis in the wrapping of the EDI message. The basic idea is that the obligations do not automatically follow from the meaning of an EDI message. The need for meaning unfolding and wrapping of EDI messages also explains why electronic commerce is much more than just exchanging EDI messages.

To sum up, we should think of the interchange agreement as a sort of (static) wrapper for each EDI message governed by it. Communication is effected by sending a particular message in the context of the wrapper (i.e., interchange agreement), which is itself communicated only once. The message is lean in the sense that only a minimal, or core meaning, is expressed. The larger, or extended, meaning of the message can only be determined in light of the governing interchange agreement. Thus, to interpret a message, a recipient will unpack (or unfold) its core meaning then use the interchange agreement (and other knowledge as appropriate) to make action-relevant inferences. We mean this as a descriptively accurate (albeit abstract) account of how things are now done (cf., Figure 1). The prescriptive element in our story concerns the need for a good semantic theory, corresponding to \( \{2\} \), above, but covering all EDI messages. Again, the received view (with which we agree) holds that having a proper semantics for EDI messages would be a very good thing. Our account helps us see why: a good semantics would provide the basis for enriching and validating wrappers and unfolding.

All of this is, we think, unexceptionable and even inevitable. The criticisms that have been leveled at EDI do not pertain to the basic EDI schema, but to how it has been implemented. The challenges lie in making the creation of interchange agreements cheaper and faster, and in providing better support for unfolding the messages that are sent. What we offer here is directed at the latter problem, but the two challenges are related. The more that interchange agreements are structured and formalized, the better will the two challenges be met.

Before we present and discuss how exactly we propose to treat message unfolding, and the semantics of EDI messages, we need to say more about our notions of unfolding and wrapping.

3 Unfolding

We begin with two distinctions. The logical theory we offer will be sensitive to both.

First, we distinguish two sorts of meaning: core meaning and extended meaning. This distinction—which we alluded to above—is similar to the distinction made in speech act theory between the locutionary act and the illocutionary act. There, a distinction has been accepted between the sentence meaning (locutionary act) and the speaker's meaning (illocutionary act), although the two meanings may coincide. For example, the sentence meaning of "Do you know what time it is?" (an asking of the addressee whether the addressee is aware of the hour) is different from the usual speaker meaning (a request to be told the time). We recognize such a distinction in ordinary language, when we say things like "What she said was X but what she meant was Y."

As we have already noted, EDI messages have certain core meanings (in virtue of the standards to which they conform) as well as additional meanings that are un-
packed in light of the governing interchange agreement (the wrapper).

Second, we distinguish two types of unfolding: event unfolding and meaning unfolding. These represent two essential aspects of unfolding in general, as we conceive it. The following two subsections discuss each in turn.

3.1 Meaning Unfolding

Meaning unfolding is the articulation of the basic semantics for an expression. Using the example of promising, we may think of meaning unfolding as follows. First, consider the following expression, which should be thought of as belonging to an arbitrary application language—such as an EDI language—in which we can say that a promise is being made. (We will use typewriter font, along with Greek letters, for expressions in an application language. We will use standard logical notation for expressions in a logical language, which we use to present the semantics for an application language expression.)

\[ \text{promise}(e, \Phi) \]  

Expression (1) represents that there is an event, \( e \), of making a promise and that what is promised is \( \Phi \). If we want to know what the real content is of expression (1) and what it fundamentally means to make a promise, then we need to add more information. In other words, we need a theory to help us unfold the meaning of the expression, e.g., (1). In Kimbrough's theory (cf. [8, 9, 11]), based on ESΘ semantics, this is done in the following way.

\[ \text{promise}(e) \land (K(e) \land \exists e'(\Phi(e'))) \]  

where \( \text{promise}(e) \) means that \( e \) is a promising event, and \( K(e) \) means that the promise—the event \( e \) which is a promise—is kept. The expression \( \exists e'(\Phi(e')) \) says what the keeping of this particular promise means, i.e., it expresses the specific content of the promise being made. For example, if the promising event is a promise made in a purchase order to pay for goods after delivery (\( \Phi \)), then this promise is kept if payment is actually arranged after delivery of the goods. Figure 2 presents a more elaborate example, in the context of electronic commerce. The underlying idea is quite simple: A promise is kept if and only if what is promised is (neglecting tense) true.\textsuperscript{5} Thus, our suggestion is that

\text{we should think of \text{Promise}(e, \Phi) as standing for a possible EDI expression, and the content explanation—the semantics—is expressed by \( \exists e'(\Phi(e')) \)). Stating this more carefully, the idea is to find a \textit{translation function}, \( f \), that unfolds the EDI expression by mapping it to a formula that expresses the semantics, or core meaning, of the EDI expression. Symbolically:

\[ \text{Promise}(e, \Phi) \leftrightarrow \text{promise}(e) \land (K(e) \land \exists e'(\Phi(e'))) \]  

and \( f \), here unspecified, does the mapping. (Again, this is a schema, an abstraction only meant to communicate the essentials of the idea.)

Some perspective may be helpful in understanding these remarks. For statements, or messages, we should consider three levels of representation:

1. Informal, typically in natural language.

Example:

"All politicians are liars." (4)

2. Syntactically formal, but not fully logical (e.g., without formal rules of inference, or a formal semantics).

Example (for representing "All politicians are liars," i.e., (4) \( \iff \) (5)):

\[ \text{politician and liar}(x) \]  

3. Semantically formal, i.e., in a logic and with a formal semantics. Example (for representing "All politicians are liars," i.e., (4) \( \iff \) (6) and (5) \( \iff \) (6)):

\[ \forall x(P(x) \rightarrow L(x)) \]  

Now, if machines are to process messages effectively, the messages cannot be expressed informally, as in Expression (4). Instead, formal expressions, using an appropriate and convenient syntax, as for example in Expression (5), are required. Important for the success of a formal messaging syntax, as in Expression (5), is a theory of semantics for what the messaging syntax means. This is provided (in outline) by Expression (6) for Expression (5). Similarly, we mean Expression (2) to have the same relationship to Expression (1) as Expression (6) has for Expression (5). Further, notice that Expression (6) is really an unfolding of the meaning of Expression (5), as is (2) of Expression (1). Our

\text{What we have here is a schema, an abstraction we are using to communicate the sense of the underlying approach. For details see [8, 9, 10, 11].}

\text{5Of course, this is not exactly what (2) says, and of course much more needs to be said and articulated about promises.}
point is not that formal logic expressions, such as (6) and (2), should be used instead of syntactically convenient expressions, such as (5) and (1). Instead, our point is that a theory of unfolding—with a semantics, as in (6) and (2)—is valuable for understanding how to design effective automation for electronic commerce. For example, something a proper logic gives us is an automatable way of exploring the inferences that can or cannot be made from a collection of statements. Especially for the sort of complex and subtle reasoning we need to support in electronic commerce, e.g., reasoning about commitments and obligations, this should be valuable both for specifying and validating software systems (working, e.g., with syntactically formal EDI expressions) and for facilitating the construction (dare we hope automatically?) of interchange agreements and other wrapping elements.

Comments: (a) There is nothing special about promising. The move in evidence here generalizes to the other illocutionary forces [8, 9, 10, 11]. (b) Although further details are required, the analysis is remarkably simple. Even so, it is able to capture the core logical behavior of various illocutionary forces [8, 9, 10, 11]. (c) Notice that the meaning, as analysed, implies nothing about whether certain actions are obliged by a promise! Promising is one thing, keeping a promise is another. We believe that this in fact accords well with the leanness of EDI messages, noted and discussed above. For example, it standardly belongs to the meaning of a purchase order that the seller delivers good to the buyer and the buyer pays the seller, but this is only obligatory for these parties if additional assumptions are made. Making these assumptions is normally part of what the interchange agreement (the wrapper) is about. In this respect our analysis deviates from standard view, or central dogma, of speech acts, exemplified in the work of Searle and Vanderveken [19, 20, 21, 29] among others. Searle and Vanderveken assume that a speech act of promising automatically implies the obligation to fulfill the promise. However, in our view this implication critically depends on the context. For some contexts it might hold and not for others. This inference is performed by using the wrapper, which we discuss in §4. After that, we will say more about the central dogma of speech acts.

3.2 Event Unfolding

As a schema for illustrating essential ideas, (2) is helpful but hardly complete. What we call the governance problem remains to be addressed if we are to have a basically adequate account of unfolding.

Consider for example the expression “The buyer promises the seller to send a request for quotation”. Intuitively this refers to three events, namely promise, send, and RFQ. Further, there is some ordering relation among the events. The example expression is about a promise to send ..., rather than, say, a sending of a promise .... We might be tempted say that when an illocutionary operator is to the left of another one, it governs this other operator. For example, in promise ... send ... RFQ ..., sends governs RFQ and these two are governed by promise. But this doesn’t work. When we have the conjunction

\[ \text{promise} \land \text{send} \land \text{RFQ} \]  

it is not clear which predicate governs which. Note also that governance cannot be defined at the predicate level. It is not always the case that the send predicate is governed by the promise one. It might be the reverse in another situation, or the same predicate might even occur at different positions, as in the expression “the buyer promises the seller to send a promise to send a request for quotation”. However, if we index our predicates with events, e.g., \( \text{promise}(e1) \land \text{send}(e2) \land \text{RFQ}(e3) \), this governance relation could be implemented by introducing a two-place predicate, Govern, on events. For example, Govern\((e1,e2)\) and Govern\((e2,e3)\). Assuming this predicate is transitive, we can infer from these Govern\((e1,e3)\). It is similar to a set containment predicate in a many-sorted first-order representation of set theory. It can also be viewed as introducing a hierarchical structure on events.

A similar, but more general, technique is used in ESO (event semantics with thematic rôles) theory [9, 10, 11, 15, 18]. Verbs are indexed with events, e.g., promise\((e1)\) reads as “\(e_1\) is a promise event”. Instead of a single predicate, Govern, to link events, we have a short list of thematic rôles that serve to perform linking between events, as well as to add semantic richness with a small, controlled vocabulary. Table 1 provides a brief list of standard thematic rôles.

With this apparatus, we can go far by way of expressing the semantics of EDI messages. Suppose that \(s\) sends a purchase order to \(r\) at time \(t\). The subject of \(s\)'s purchase order is \(g\) (some goods), which \(s\) wishes to purchase in quantity \(q_s\) and units \(u_g\), for delivery on or before time \(t_e\). For this, \(s\) is prepared to pay \(r\) an amount of money, \(q_r\), in units \(u_r\) on or before time \(t_e\). We thus have a situation analogous to, but more complex than, (1). Instead of unfolding a promise, we
need to unfold a purchase order:

$$\text{po}(e, \Phi) \leftarrow \Psi(\exists \epsilon (\Phi(\epsilon)))$$  \hspace{1cm} (8)

where $\Phi$ has arguments including $(s, r, t, g, qg, ug, te1, q5, us, te2)$. Figure 2 provides the analysis, using ES6 theory and the lean analysis of speech acts discussed above. Our claim is that the expression in Figure 2 is essentially correct. This is not something that can be proved, but it is a claim open to counter-examples. Given a normal purchase order, what does and does not follow from it? Given the expression in Figure 2, what does and does not follow from it? To the extent that the answers to these questions coincide, the expression in Figure 2 is a good model of a normal purchase order. More positively, if you translate the expression in Figure 2 back to English the result would seem to capture the essence of a normal purchase order. Here it goes. "$\epsilon$ [think of it as the ID on the PO] is a purchase order, from $s$ to $r$, dated $t$. A delivery [$\epsilon'$] of goods $g$ is requested for the sake of this purchase order. The delivery should be to $s$ by $r$ (or $r$'s agent), and should occur on or before $t_e$. Measured in units $u_g$, the amount of goods delivered should be $q_g$. On condition that the delivery occurs as just described, $s$ promises to pay [$\epsilon''$] $r$ for the delivery. Payment will be in the form of money. Measured in units $u_b$, payment will be in quantity $q_b$, and will occur on or before $t_p$."

Comments: (a) The expression in Figure 2 is entirely in first-order logic. This is moderately surprising and, from a practical perspective, entirely welcome. (b) The expression in Figure 2 uses only predicates from a smallish controlled vocabulary. Everything is either a thematic role or a verb. This promises enormous economy of expression if anything like it holds in general for EDI. (c) The analysis is very lean. There is nothing here about the beliefs or intentions of anyone involved in the transaction, nor is there anything about the consequences of not delivering or not paying. Perhaps the analysis is too lean. Has something essential been left out? We think not. Certainly, we require apparatus for inferring additional consequences of a PO message, but that belongs, we think, in the wrapper for the message. To that now.

### 4 Wrapping

For reasons mentioned above, illocutionary, deontic, and other aspects of an EDI message typically have to be inferred. These inferences might even be defeasible (think of DX, Lee & Ryu’s system for defeasible deontic reasoning, Donald Nute’s d-prolog). Think of modeling the legal implications of a contract. Basically this is not represented in the EDI message, but is part of a background theory from which it can be inferred. A typical example is to infer whether a promise really implies an obligation, or in more legal terms to infer what the legal status is of an EDI message. All of this might vary from country to country. The wrapper is country dependent. These are large and important issues, and they cannot be resolved here in a short paper. But we can make a start.

Our suggestion is that the necessary inferences attending to EDI messages are presently achieved through a system of lean messaging with wrapping and subsequent unfolding. Further, we suggest there are good reasons to mirror this strategy in new protocols and in the development of a formal semantics for EDI, and we have offered (a sketch of) an account of how to represent the lean semantics (the unfolding) of an EDI message. Our aim now is to indicate how to formalize part of the wrapping so that required additional inferences can be made automatically.

<table>
<thead>
<tr>
<th>Role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent</td>
<td>Volitional initiator of action</td>
</tr>
<tr>
<td>Patient</td>
<td>Object or individual undergoing action</td>
</tr>
<tr>
<td>Theme</td>
<td>Object or individual moved by action</td>
</tr>
<tr>
<td>Goal</td>
<td>Individual toward which action is directed</td>
</tr>
<tr>
<td>Source</td>
<td>Object or individual from which something is moved by the event, or from which the event originates</td>
</tr>
<tr>
<td>Experiencer</td>
<td>Individual experiencing some event</td>
</tr>
<tr>
<td>Beneficiary</td>
<td>Object that benefits from the event</td>
</tr>
<tr>
<td>Location</td>
<td>Place at which the event is situated</td>
</tr>
<tr>
<td>Instrument</td>
<td>Secondary cause of event; the object or individual that causes some event that in turn causes the event to take place</td>
</tr>
<tr>
<td>Speaker</td>
<td>Volitional initiator of speech act (cf. Agent)</td>
</tr>
<tr>
<td>Addressee</td>
<td>To whom a speech act is directed</td>
</tr>
<tr>
<td>Unit</td>
<td>The unit of measurement for a given item</td>
</tr>
<tr>
<td>Quantity</td>
<td>The quantity of a given item (in the units)</td>
</tr>
<tr>
<td>Cul</td>
<td>Culmination time for an event</td>
</tr>
<tr>
<td>Sake</td>
<td>$x$ is for the sake of $y$</td>
</tr>
</tbody>
</table>

Table 1: Examples of thematic roles. (See also [11].)
particularly interesting and challenging case. Standard deontic logics are about obligations tout court. For business purposes, among others, we need a concept of, and hence a formalization of, directed obligation in which one agent has an obligation to another agent to do something. In the case of our purchase order, if \( r \) delivers, then \( s \) not only has to (is obliged to) pay for the goods, but has to pay \( r \) in particular. How to model this has only recently begun to be investigated. Initial results were not fully satisfactory (see [6]). Recently, however, Tan’s group has developed an analysis and a logic for directed obligation (see [27, 26]). The central formula

\[
\exists e' \exists e'' \exists t' \exists t'' ((pe(e) \land Speaker(e, s)) \land \neg Addressed(e, r) \land \text{Theme}(e, (e'k'')) \land \text{Cu}(e, t)) \land
\]

\[
H(e) \rightarrow
\]

\[
(delivering(e') \land \text{Agent}(e', r) \land \text{Goal}(e', s) \land \text{Theme}(e', g)) \land
\]

\[
\text{Sake}(e', e) \land \text{Unit}(e', g, u_g) \land \text{Quantity}(e', g, q_g) \land \text{Cu}(e', t') \land t' \leq t(e)) \land
\]

\[
(H(e) \rightarrow
\]

\[
\text{promising}(e) \land
\]

\[
(K(e) \rightarrow (paying(e'') \land \text{Agent}(e'', s) \land \text{Goal}(e'', r) \land \text{Theme}(e'', g)) \land
\]

\[
\text{Sake}(e'', e') \land \text{Unit}(e''), \$, u_\$) \land \text{Quantity}(e'', \$, q_\$) \land \text{Cu}(e'', t'') \land t'' \leq t(e)))
\]

Figure 2: Representation of a Simple Purchase Order (extensional approximation). (Note: Terms without quantifiers, e.g., \( e, \$, \) are names, or constants. \( (x|y) \) is the compound event consisting of events \( x \) and \( y \).)

\[\exists e' \exists e'' \exists t' \exists t'' ((pe(e) \land Speaker(e, s)) \land \neg Addressed(e, r) \land \text{Theme}(e, (e'k'')) \land \text{Cu}(e, t)) \land H(e) \rightarrow (delivering(e') \land \text{Agent}(e', r) \land \text{Goal}(e', s) \land \text{Theme}(e', g)) \land \text{Sake}(e', e) \land \text{Unit}(e', g, u_g) \land \text{Quantity}(e', g, q_g) \land \text{Cu}(e', t') \land t' \leq t(e)) \land (H(e) \rightarrow \text{promising}(e)) \land (K(e) \rightarrow (paying(e'') \land \text{Agent}(e'', s) \land \text{Goal}(e'', r) \land \text{Theme}(e'', g)) \land \text{Sake}(e'', e') \land \text{Unit}(e''), \$, u_\$) \land \text{Quantity}(e'', \$, q_\$) \land \text{Cu}(e'', t'') \land t'' \leq t(e)))\]

where \( O \) is completely anonymous. Secondly, \( \text{\texttt{\#O_i^S}} \) defines in purely objective and operational terms what will happen if the obligation is violated. For example, if a buyer has the obligation towards the seller to bring about payment of the delivered goods, then the seller has the permission to bring about legal action against the buyer if he does not pay. Symbolically, we have:

\[\text{\texttt{\#O_i^S}}(\text{paying}(e'')) \text{def}= (\neg \text{\texttt{paying}(e'')} \Rightarrow s \text{\texttt{P}(\text{\texttt{Ela}_b)}})\]

where \( b \) is the buyer and \( s \) the seller.

The operational character of this definition gives not only a detailed explanation of what it means to have a contractual obligation between two trading partners, but it makes the notion of directed obligation also applicable to artificial agents, because it is only the operational behaviour of the agent that matters and not his internal mental states, which artificial agents probably lack.

With this definition (and attendant logic) we can write bridge laws or meaning postulates or Protokolssätze that connect the language of messaging with the language of directed obligation. Here are two examples:

\[\forall e, e', s, r \]

\[ (pe(e) \land Speaker(e, s) \land Addressed(e, r) \land accepting(e') \land Agent(e', r) \land Theme(e', e)) \]

\[ (\neg \text{\texttt{\#H}(e)} \Rightarrow s \text{\texttt{P}(\text{\texttt{Ela}_b)}}))\]

If you get a PO and you accept it (i.e., you send a message in which you say accepting(e') and Agent(e', r) \land Theme(e', e)), if you don’t honor it (deliver the goods),
the sender of the PO can initiate legal action against you.

\[
\forall e, e', s, r \\
(p(e) \land \text{Speaker}(e, s) \land \text{Address}(e, r) \land \\
H(e) \rightarrow (\neg s E K(e) \Rightarrow s \Rightarrow P(\xi 1a_e)))
\]

If you send a PO and it is honored (the goods are delivered) and you don’t pay, then the receiver of the PO can bring legal action against you.

Expression (21) shows clearly the technical difference between meaning unfolding and wrapping. The predicate \(K(e)\) is an essential part of the unfolding of the purchase order (Figure 2), but in this unfolding nothing is said yet about obligations. Nothing in Figure 2 says that \(K(e)\) is obliged! To express this deontic aspect we have to add the definition of directed obligation\((\neg s E K(e) \Rightarrow s \Rightarrow P(\xi 1a_e))\). Note that this adding of obligations, of course, critically depends on whether the purchase order is legally binding according to an interchange agreement. This gives us the following formal picture of the different stages in unfolding and wrapping illustrated with the example of promising. We start with the event of promising (now notionally expanded to include references to the buyer and the seller):

\[
\text{Promising}(e, s, r, \Phi)
\]

To this the meaning unfolding adds the meaning, creates the expression:

\[
\text{promising}(e) \land \text{Speaker}(e, s) \land \text{Address}(e, r) \land \\
(K(e) \rightarrow \exists e'(\Phi(e')))
\]

To this the wrapping adds the deontic aspect of obligations, as in Expressions (20) and (21). Note: Expression (23) is a simplified version of the expression in Figure 2, which itself is a simplified example. All these simplifications are for the sake of illustrating some larger points, about wrapping and unfolding. In a nutshell, we can think of the wrapper (including an interchange agreement and legal background) as a theory that—when formalized—logically implies the expression of a directed obligation, given an input message that has been unfolded.

Comments: (a) ESO theory has paid off handsomely here. Notice that the bridge laws ((20) and (21)) are completely general and use only the controlled vocabulary (verb predicates, thematic role predicates, and predicates from the logic of directed obligation). (b) The illocutionary auxiliary predicates (here \(H\) for “is honored” and \(K\) for “is kept”) of Kimbrough’s semantically lean theory of speech acts have a natural and essential place in the example bridge laws, allowing them to be general. (c) Point (b) leads one to wonder whether ESO theory might be used to analyze further the logic of directed obligation. We suspect so, but following up on that is subject for another paper. (d) Much remains to be done if wrappers for electronic commerce are to be substantially formalized and given a full semantics. That must be a subject for many subsequent papers, but we see no fundamental impediments to completing such a programme of research.

In sum, what we have presented here is an argument for a certain way of thinking about how to formalize transactions in electronic commerce (lean messaging, wrapping, unfolding, representation with ESO theory, directed obligation theory, etc.). We now turn to a discussion of the larger significance of what we are proposing.

References


