What is “Next” in Event Processing?

Walker White, Mirek Riedewald,
Johannes Gehrke, Alan Demers*
Cornell University
Ithaca, NY 14853, USA
{wmwhite,mirek,johannes,ademers}@cs.cornell.edu

ABSTRACT

Event processing systems have wide applications ranging from managing events from RFID readers to monitoring RSS feeds. Consequently, there exists much work on them in the literature. The prevalent use of these systems is on-line recognition of patterns that are sequences of correlated events in event streams. Query semantics and implementation efficiency are inherently determined by the underlying *temporal model*: how events are sequenced (what is the “next” event), and how the time stamp of an event is represented. Many competing temporal models for event systems have been proposed, with no consensus on which approach is best.

We take a foundational approach to this problem. We create a formal framework and present event system design choices as axioms. The axioms are grouped into *standard axioms* and *desirable axioms*. Standard axioms are common to the design of all event systems. Desirable axioms are not always satisfied, but are useful for achieving high performance.

Given these axioms, we prove several important results. First, we show that there is a unique model up to isomorphism that satisfies the standard axioms and supports associativity, so our axioms are a sound and complete axiomatization of associative time stamps in event systems. This model requires time stamps with unbounded representations. We present a slightly weakened version of associativity that permits a temporal model with bounded representations. We show that adding the boundedness condition also results in a unique model, so again our axiomatization is sound and complete. We believe this model is ideally suited to be the standard temporal model for complex event processing.

Categories and Subject Descriptors

H.2.3 [**Database Management**]: Logical Design—*Data Models*

*This material is based upon work supported by the National Science Foundation under Grants 0621438 and 0121175, by the Xerox Foundation, and by a Sloan Foundation Fellowship. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

General Terms

Theory

Keywords

Axiomatization, Events, Temporal Models

1. INTRODUCTION

Complex event processing (CEP) systems are an important component of today’s information system infrastructures. Examples of the broad application space for CEP include supply chain management for RFID (Radio Frequency Identification) tagged products, real-time stock trading, monitoring of large computing systems to detect malfunctioning or attacks, and monitoring of sensor networks (e.g. for surveillance). A growing number of companies are developing products in this space [1].

Starting with early work on monitoring of computing systems, many designs have been proposed for event processing systems [2, 6, 7, 9, 14, 15, 18]. The input to a CEP system is a stream of events, generated by external processes. Users register long-running queries — also called subscriptions — to detect interesting *event patterns*, which are typically sequences of correlated events. One particularly important class of query is a *safety condition*, a query meant to detect when “something bad” happens between two events. For example, the event system SASE [18] is motivated by the following safety condition for RFID tracking in a retail store.

Query 1. Post a notification if an item, after being removed from the shelf, exits the store before being checked out at the counter.

To process such queries, all event systems have, as part of their pattern language, a *sequencing* (or immediate concatenation) operator. This operator is typically denoted $E_1 ; E_2$. It finds any event matching the subpattern E_1, and then finds the *first* match afterwards to the subpattern E_2. For example, assume our pattern language has an “or” (|) operator. To process the above safety condition, we can search for the pattern

$$\text{LEAVE SHELF};(\text{CHECKOUT}|\text{EXIT STORE}) \qquad (1)$$

and then filter out all matches that do not go through the checkout.

This example is in some ways deceptively simple. It implicitly relies on an assumption that events are instantaneous and totally ordered, an assumption which may not be true. Many, if not most, CEP systems are *compositional* — the outputs of a query are themselves events, which can be posted to the event stream and used in other queries. Such events are referred to as *composite* (or *complex*) events, as they are composed of several smaller events that together satisfy the query. For example, in a more realistic formulation of Query 1, the CHECKOUT event might be a complex event involving several steps, such as scanning an item, reading the purchaser’s credit card, and receiving validation from the credit card company.

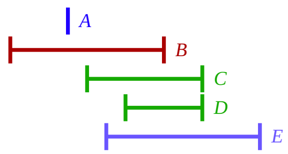

Figure 1: Possible Successors to Announcement A

Since it consists of multiple smaller events, a complex event has duration; it does not exist at a single point in time, but rather occupies an interval with distinct start and end times. Therefore, complex events can, and often do, overlap with each other. This can cause some difficulty in processing the sequencing operator; for events with duration the definition of "next" is not obviously unique.

To illustrate this, we present a slightly more subtle example. Consider an Internet retailer that uses an event system to to ensure that its website properly follows its business workflow processes. The retailer holds periodic giveaways to its customers. Each giveaway is announced on the website, and then the first (or equivalently n^{th}) customer to make a purchase after the announcement wins the prize. A purchase is itself a complex event comprising two other events: adding a item to the shopping cart and paying for the items in the cart. Therefore, the contest consists of the following event system query.

Query 2. After a giveaway is announced on the main website, identify the first new purchase (i.e. adding an object to the shopping cart and then paying for it).

Assuming that an announcement is an instantaneous event, processing Query 2 involves determining which of a number of intervals (the purchases) is the immediate successor of each announcement. But it is not obvious what the correct definition of successor should be in this case.

Consider the intervals shown in Figure 1. Here A represents an announcement, and the remaining intervals represent purchases (starting when an object is added to the cart, ending at the time of payment). If we choose successor only according to the end time of the interval, then interval B is the successor to A. However, B properly contains A, and so it makes sense to disqualify it as a possible successor to A (i.e. the retailer does not want customers "jumping the gun"). We should instead choose interval C or D as the next event, but it is unclear which.

A plausible choice might be to define the successors of A to include every interval that does not have another interval strictly between it and A, i.e., the natural definition of successor on the partial order of intervals. However, as we show in Section 2, this results in a nonassociative definition of sequencing. An associative sequencing operator is desirable for more than just aesthetic reasons: it is important for query optimization as well. To see this, note that Query 2 is naturally written as a right-associated query, of the form $E_0;(E_1;E_2)$. To process this query, we first match the purchase events $E_1;E_2$, and then match them to the giveaway announcement E_0. If giveaways are relatively infrequent, this approach can be inefficient, as it can generate many purchase events that will never be matched to giveaway announcements. If sequencing were associative, we could rewrite the query as the left-associated version $(E_0;E_1);E_2$. In this case the system would first match announcement and shopping cart pairs, limiting the number of purchases under consideration. This transformation is analogous to constructing a join plan in relational algebra by considering the selectivities of the joins. Hence associativity in the underlying temporal model is an important enabler for query optimization in an event system implementation.

A thorough survey of temporal models in the CEP literature shows that there is no unique answer for choosing a successor to A in Figure 1. Existing systems use many different kinds of time stamp, with different semantics and different implications for implementation efficiency. The purpose of this paper is to present a formal framework for the study of sequencing in event processing systems. From this framework we show how semantic and implementation concerns limit the possible definitions of "next".

1.1 Outline of Contributions

In Section 2 we present several existing systems and show how they differ in their definition of sequencing. From these examples we identify real-world implementation concerns that we use to guide the development of our formal framework.

We present this formal framework in Section 3 with the definition of a temporal model. This definition is capable of describing sequencing in *all* event systems we are aware of, and captures the subtle distinctions between event systems that have the same (partial) order on time stamps, but different successor definitions. This gives us a uniform framework in which to discuss existing design choices.

In order to chose the right temporal model, we take an *axiomatic* approach. In Section 4, we present event system design choices as axioms, algebraic properties that the temporal model must have in order to meet the associated system design goals. We distinguish between *standard axioms* and *desirable axioms*. The standard axioms are common to the design of all event systems, representing system behavior that a user would intuitively expect. The desirable axioms are useful for improving the performance.

Given these axioms, we prove in Section 5.1 that there is only one model up to isomorphism that satisfies all of the standard axioms and supports associativity. This resolves the debate illustrated in Figure 1. It also demonstrates that our axioms are a sound and complete axiomatization of associative time stamps in event systems.

The unique model in Section 5.1 requires time stamps with unbounded representation. In Section 5.2 we present a slightly weakened version of associativity that permits a temporal model with a bounded representation of time stamps. We show that, by adding the boundedness condition, this temporal model is again unique to isomorphism, so again our axioms are sound and complete. More importantly, we give arguments for adopting this model as the standard temporal model for CEP.

We end the paper with a discussion of related work (Section 6), concluding remarks and a discussion of future work (Section 7).

2. SUCCESSOR IN EXISTING SYSTEMS

To illustrate the importance of an event system's successor definition, we examine the behavior of Query 2 in several representative event systems taken from the literature.

Consider the sequence of events represented in Figure 2. In the figure, A represents the giveaway announcement. Each P_i, Q_i and R_i represents a purchase step by a customer. Dashed horizontal lines connect the shopping cart step with the payment. Intuitively, only purchase Q_1Q_2 satisfies the query. Purchase P_1P_2 is disqualified because the shopping cart was filled (event P_1) before the announcement (The retailer wants the contest to be fair, and does not want customers "lying in wait" for the next giveaway). Purchase R_1R_2 is disqualified because Q_1Q_2 clearly finished before it, and hence it cannot reasonably be considered "first".

Now consider the event stream of Figure 3. Intuitively, purchases Q_1Q_2 and R_1R_2 both satisfy the query: each is a purchase after the announcement and they start and finish *simultaneously*. Thus,

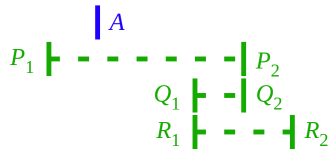

Figure 2: Overlapping Purchases

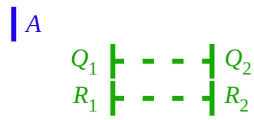

Figure 3: Simultaneous Purchases

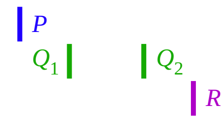

Figure 4: Associating Sequencing

the query result should be a notification for both events. Therefore, the event system must be able to support multiple successors – both Q_1Q_2 and R_1R_2 must be successors of A.

Given this query and these data streams, we now examine how they would be processed in several existing event systems. A number of systems such as Snoop [6] or EPL [14] use *point time stamps* taken from a discrete, totally ordered domain. In these systems, the composite event P_1P_2 has only a single time stamp — that of P_2. Hence the information that the item was added to the shopping cart before the announcement A is lost, and P_1P_2 incorrectly appears to satisfy the query. A similar criticism applies to the weak successors used in Active Office [15]. In order to express this query correctly, the successor relation must prohibit overlap between time *intervals*.

Event systems such as SnoopIB [2] and ODE [9] use interval time stamps. These systems order *non-overlapping* intervals in the natural way. They also use the standard definition of immediate successor for partially ordered sets: interval time stamp t_1 is a successor of t_0 if $t_0 < t_1$ and there is no t_2 with $t_0 < t_2 < t_1$. In this model both the Q_1Q_2 and R_1R_2 purchases from Figure 2 qualify as successors of the A event. In general, these systems allow an event to have multiple successors — even an unbounded set — and not all the successors are required to finish at the same time. Again, this definition of successor does not accurately reflect the intuitive meaning of the query.

The strong successor used in Active Office captures the correct intuitive meaning of our example query on the event stream of Figure 2. But even this model fails on the stream of Figure 3, because the strong successor rules of Active Office select only one of the two simultaneous events. It uses a tie-breaking scheme based on arbitrarily assigned, totally ordered unique identifiers.

The Cayuga system [7] uses interval time stamps like SnoopIB, but has a different successor definition. Specifically, $t = [t_0, t_1]$ is a successor of $s = [s_0, s_1]$ if $t_0 > s_1$ and there is no event with time stamp $p = [p_0, p_1]$ such that $s_1 < p_0 < p_1 < t_1$. In other words, t is a successor of s if t follows s without overlap, and no p that follows s without overlap finishes before t. This definition deals correctly with our motivating example query in all cases, and also avoids unbounded successor sets with their associated implementation difficulties.

As discussed in Section 1, associativity of sequencing can be viewed as an important enabler for query optimization in an event system implementation. Unfortunately, associativity has serious implications for the definition of successor. Consider the event stream of Figure 4. Here P is an event matching some event expression E_P, Q_1 and Q_2 are events matching E_Q, and R is an event matching E_R. Using any of the event systems discussed above, the expression $E_P\,;E_Q$ yields a single composite event PQ_1, and thus the left-associated expression $(E_P\,;E_Q)\,;E_R$ yields only the event PQ_1R. However, expression $E_Q\,;E_R$ produces *two* composite events, Q_1R and Q_2R. Therefore, the right-associated expression $E_P\,;(E_Q\,;E_R)$ yields the events PQ_1R and PQ_2R using any of systems above except Active Office (which eliminates one of the composite events due to its tie-breaking rule). While Active Office handles this particular expression correctly, it fails to be associative in general, as we show in Section 5.2.

We know of no existing system whose temporal model supports associative sequencing. As we show in Section 5.1, there is a good reason for this: up to isomorphism, the *only* temporal model that supports associative sequencing is the *complete-history* model. This model, as its name suggests, requires a system to store the time stamps of all the primitive events that make up a composite event. Since it has no upper bound on the size of a time stamp representation, complete-history can be prohibitively expensive.

3. THE FRAMEWORK

In Section 2, we saw how successor was defined differently in three event systems: SnoopIB, Active Office, and Cayuga. All three use intervals as time stamps, and they have the same partial order on these intervals: $[s_0, s_1] < [t_0, t_1]$ if and only if $s_1 < t_0$. However, the three systems differ in how they choose a successor.

Because all of these systems have the same time stamps with the same partial order, to study their differences, we have to extend previous work on temporal models [16] to include a successor operation that exists explicitly in the model. To be as general as possible, we define a *successor function* SUCC to be a function that takes as input a time stamp t together with a set of time stamps $\mathcal{F}$ and produces the set $\textsc{succ}(t, \mathcal{F})$ of immediate successors of t in $\mathcal{F}$. The intuition for this model is that *candidate set* $\mathcal{F}$ represents the set of time stamps from which the immediate successor is chosen. Given an event expression $E_1\,;E_2$, an event system matches this expression by doing the following:

(a) Determine the set of candidate time stamps $\mathcal{F}$ for events matching E_2.

(b) For each event matching E_1 at time t, compose it with any event matching E_2 at a time in $\textsc{succ}(t, \mathcal{F})$.

These three event systems define $\textsc{succ}([s_0, s_1], \mathcal{F})$ as follows.

SnoopIB:

$$\left\{[t_0,t_1]\in\mathcal{F} \;\middle|\; s_1<t_0 \text{ and } \neg\exists[r_0,r_1]\in\mathcal{F} \text{ s.t. } s_1<r_0\le r_1<t_0\right\}$$

Cayuga:

$$\left\{[t_0,t_1]\in\mathcal{F} \;\middle|\; s_1<t_0 \text{ and } \neg\exists[r_0,r_1]\in\mathcal{F} \text{ s.t. } s_1<r_0\le r_1<t_1\right\}$$

Active Office (strong successor):

$$\left\{[t_0,t_1]\in\mathcal{F} \;\middle|\; \begin{array}{l} s_1 < t_0 \wedge \neg\exists[r_0,r_1] \in \mathcal{F} \text{ s.t.} \\ s_1 < r_0 \wedge (r_1 < t_1 \vee (r_1 = t_1 \wedge r_0 < t_0)) \end{array}\right\}$$

The successor operation is not the only way that event models may differ. When we compute the result of a query like Query 2, we need to assign the composite event (the output of the query) a new time stamp. In SnoopIB, Cayuga, and Active Office, the composite event is timestamped with the smallest interval containing the intervals of all events that make up the query result. For example, Query 2 is made up of three events. If these events happen at times 1 (giveaway announcement), 2 (product added to cart), and 4 (payment received), then all three systems assign $[1, 4]$ as the result time stamp. However, the event system ODE is different. It keeps the history of all of the time stamps in the component events. In the example, it would store $[1, 2, 4]$ as the time stamp. Nevertheless, ODE still only uses the *boundaries* of this history when determining the immediate successor, treating the history like an interval; hence it is not appreciably different from SnoopIB.

For the remainder of the paper, our approach is more formal. Traditionally a *temporal model* is defined as $(T, \prec)$ where $\prec$ is a partial order on the set of time stamps T [16]. The elements of T can be points, intervals, sets of points, sets of intervals, and so on; there are no restrictions on the types of acceptable time stamps. To study both immediate successor and event composition, we extend this definition of a temporal model to a quadruple $\mathbb{T} = (T, \prec, \textsc{succ}, \otimes)$. In this model, T and $\prec$ are the same as in the traditional model. In addition, the successor function $\textsc{succ} : T \times 2^T \to 2^T$ takes a time stamp t together with a set of candidates $\mathcal{F}$ and produces the set of immediate successors. Finally, the *composition operation* $\otimes$ takes the time stamps s and t of two events and produces the time stamp $s \otimes t$ for the corresponding composite event. The time stamps $s \otimes t$ are also in T, since composite events may be added to the event stream for use in other queries. For convenience, we will identify $\mathbb{T}$ and T when the context is clear (e.g. a time stamp $t \in \mathbb{T}$).

While $\otimes$ behaves like a monoid operation, we do not always want it to be defined. For example, in an interval model like SnoopIB, we never want to compose two overlapping events. To avoid the use of partial operations, we introduce a special "undefined" time stamp $\perp$ to $\mathbb{T}$ such that for any t, $\mathcal{F}$, (a) $\perp \notin \textsc{succ}(t, \mathcal{F})$, (b) $\textsc{succ}(\perp, \mathcal{F}) = \emptyset$, and (c) $t \otimes \perp = \perp \otimes t = \perp$. We say that $s \otimes t$ *is defined* whenever $s \otimes t \neq \perp$.

3.1 Some Concrete Examples

We have already outlined how to express SnoopIB, Active Office, and Cayuga in our framework. As an illustrative example, we give a complete formalization of ODE. In ODE, all time stamps are monotonically increasing finite sequences over the discrete linear order $\mathbb{Z}$. In other words, the time stamps are sequences $\sigma = \sigma(0)\sigma(1)\ldots\sigma(k-1)$ where $\sigma(i) < \sigma(i+1)$ for all $i < \ell(\sigma) - 1$, with $\ell(\sigma) = k$ the *length* of the sequence. The partial order is defined as $\sigma \prec \tau$ exactly when $\sigma(\ell(\sigma) - 1) < \tau(0)$, (i.e. when the largest element of σ is less then the smallest element of τ). The successor operation is defined as

$$\textsc{succ}(\sigma, \mathcal{F}) = \{\, \tau \in \mathcal{F} \mid \sigma \prec \tau \text{ and } \neg\exists \rho \in \mathcal{F}, \sigma \prec \rho \prec \tau \,\}$$

Finally, for two events $\sigma < \tau$, the composition $\sigma \otimes \tau$ is the standard sequence composition (concatenation of sequences).

An interesting variation of the ODE temporal model is the *complete-history model*. In this model, T, $\prec$ and $\otimes$ are exactly the same as in ODE. However, the successor function is different. We define a linear ordering on time stamp histories by letting $\sqsubseteq$ be the lexicographical ordering from the end of the sequences. In other words, $\sigma \sqsubseteq \tau$ if either

- $\sigma(\ell(\sigma) - i) < \tau(\ell(\tau) - i)$, and $\sigma(\ell(\sigma) - k) = \tau(\ell(\sigma) - k)$ for $k < i$, or
- $\ell(\sigma) < \ell(\tau)$ and $\sigma(\ell(\sigma) - i) = \tau(\ell(\tau) - i)$ for all $i < \ell(\sigma)$.

We use this linear order to break ties, and thus define

$$\textsc{succ}(\sigma, \mathcal{F}) = \{\, \tau \in \mathcal{F} \mid \sigma \prec \tau \text{ and } \neg\exists \rho \in \mathcal{F}, \sigma \prec \rho \sqsubseteq \tau \,\}$$

Notice that this model is a generalization of the definition of successor in Active Office from intervals to complete histories. However, it does not use data elements (identifiers) to break ties; it only uses time stamp ordering.

4. AXIOMATIZING TEMPORAL MODELS

Our framework puts no restrictions on the definitions of $\textsc{succ}$ and $\otimes$ yet. As a result, there can be aberrant behavior (e.g. a model in which two time stamps are successors of each other, such as $t_0 \in \textsc{succ}(t_1, \mathcal{F})$, $t_1 \in \textsc{succ}(t_0, \mathcal{F})$). As in any algebraic model, we prevent such aberrant behavior by adding axioms that express properties of "reasonable" temporal models. Since adding axioms restricts the class of valid models, we want to be sure that our axioms are all properly motivated.

We distinguish between *standard axioms* and *desirable axioms*. The standard axioms are non-controversial; they are satisfied by the temporal models in all of the major event systems. The desirable axioms, on the other hand, are each violated by at least one major event system. However, as we shall demonstrate, there are compelling reasons for wanting our temporal models to satisfy the desirable axioms.

4.1 Standard Axioms

Many of the accepted axioms have already been implicitly mentioned in our discussion of temporal models. For the sake of completeness, in this section we will make all of these assumptions explicit. As we have several axioms, we organize them according to their defining feature: $\prec$, $\textsc{succ}$, or $\otimes$.

4.1.1 The $\prec$ Axioms

As in traditional temporal models, $\prec$ should be a partial order. The following two axioms capture this property.

AXIOM 1 (TRANSITIVITY). *If $t_0 \prec t_1$, $t_1 \prec t_2$, then $t_0 \prec t_2$.*

AXIOM 2 (IRREFLEXIVITY). *For any $t \in \mathbb{T}$, $t \not\prec t$.*

4.1.2 The $\textsc{succ}$ Axioms

Another implicit assumption of our discussion has been that we always chose the successor time stamp from the candidate set $\mathcal{F}$. This assumption is expressed by the following axiom.

AXIOM 3 (CANDIDATE PRESENCE). *For all $t \in \mathbb{T}$ and $\mathcal{F} \subseteq \mathbb{T}$, $\textsc{succ}(t, \mathcal{F}) \subseteq \mathcal{F}$*

Additionally, the idea of successor is tightly-coupled with the partial order $\prec$. For example, if b is a successor of a, we generally assume that a "happens before" b. We capture this idea with the following axiom.

AXIOM 4 (RESPECTING ORDER). *For any $t, s \in \mathbb{T}$, $t \prec s$ if and only if there is some $\mathcal{F}$ such that $s \in \textsc{succ}(t, \mathcal{F})$*

In all the major event systems, the elements of $\textsc{succ}(t, \mathcal{F})$ are *natural $\prec$-successors* of t. That is, $\textsc{succ}(t, \mathcal{F})$ contains only elements of $\mathcal{F}$ that follow t and have no $\prec$-intermediate elements. In fact, the existing event systems differ only in how they choose from these $\prec$-successors; SnoopIB and ODE take them all, while Cayuga and Active Office are more selective and "break ties". To ensure this type of behavior, we need two axioms on the usage of candidate sets. The first axiom ensures the following intuitive behavior: removing any time stamps other than a successor from the candidate set should have no effect on the current successor.

AXIOM 5 (THINNING). *Suppose $t_1 \in \textsc{succ}(t_0, \mathcal{F})$. Then for any $\mathcal{T} \subseteq \mathcal{F}$ with $t_1 \in \mathcal{T}$, $t_1 \in \textsc{succ}(t_0, \mathcal{T})$.*

This axiom also addresses another important issue. We know from Axiom 4 (RESPECTING ORDER) that $t_0 \prec t_1$ whenever there is *some* $\mathcal{F} \subseteq \mathbb{T}$ such that $t_1 \in \textsc{succ}(t_0, \mathcal{F})$. But this means we could have a temporal model that permits only singleton candidate sets (i.e. $\textsc{succ}(t, \mathcal{F}) = \emptyset$ if $|\mathcal{F}| > 1$). This would correspond to an event system that shuts down if it ever receives more than one future event. Clearly this is undesirable behavior. To prevent RESPECTING ORDER from degenerating as such, we need to be able to

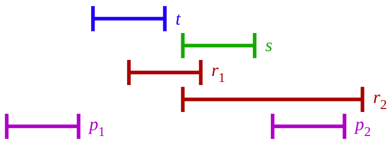

Figure 5: Adding Elements to a Candidate Set

add and remove elements from the candidate sets in limited ways. THINNING addresses this problem of removal.

To relate SUCC to $\prec$-successors, we need an axiom for adding to a candidate set; this is much more subtle. Consider the intervals illustrated in Figure 5. Suppose we are trying to pick the successors of t, and start with the candidate set $\mathcal{F} = \{ s \}$ (so trivially, s is the unique successor). If we extend $\mathcal{F}$ to the candidate set $\mathcal{F}' = \{ r_1, s \}$, then the successor depends on our choice of event system. In early event systems such as EPL and the original Snoop, the time of an event is identified with the end of its interval, and so r_1 is the successor in this model. However, in all event systems with interval time stamps, the successor is still s, since the interval r_1 started before the end of t. Similarly, the effect of adding r_2 to $\mathcal{F} = \{ s \}$ is also system dependent. In Active Office and Cayuga, the addition has no effect on the successor, as the interval ends later than s. However, in SnoopIB, r_2 is also a successor, and hence this addition changes the contents of $\text{SUCC}(t, \mathcal{F})$.

Fortunately, all event systems agree that the addition of time stamps like p_1 or p_2 to $\mathcal{F} = \{ s \}$ does not effect the value of $\text{SUCC}(t, \mathcal{F})$. These are time stamps that are either far in the past or far in the future, in that they do not overlap the time between t and any of its successors. Thus, in all event systems, we are permitted to add to a candidate set via the following axiom.

AXIOM 6 (THICKENING). *Let $\mathcal{A}$ be such that, for any $s \in \mathcal{A}$, either $s \prec t$ or $p \prec s$ for some $p \in \text{SUCC}(t, \mathcal{F})$. Then $\text{SUCC}(t, \mathcal{F}) = \text{SUCC}(t, \mathcal{F} \cup \mathcal{A})$.*

THICKENING is important for two reasons. First of all, in an event system the candidate set $\mathcal{F}$ is effectively infinite. It represents the time stamps of all the events that appear in the stream after t. An event system therefore never knows the full contents of $\mathcal{F}$ ahead of time; it only learns the values of these time stamps as they arrive. Hence, if we expect to have a real-time event processing system, the definition of successor cannot rely on such future events.

Additionally, as the following propositions demonstrate, Axioms 1-6 are enough to guarantee that $\text{SUCC}(t, \mathcal{F})$ only chooses elements from $\mathcal{F}$ that are the $\prec$-successors of t. For reasons of space, we omit all but a few important proofs in this paper; the proofs of all results may be found in [17].

PROPOSITION 1. *For any $t_0, t_1 \in \mathbb{T}$, $t_0 \prec t_1$ if and only if $t_1 \in \text{SUCC}(t_0, \{ t_1 \})$.*

PROPOSITION 2. *If $t_1 \in \text{SUCC}(t_0, \mathcal{F})$, then $t_0 \prec t_1$ and there is no $s \in \mathcal{F}$ with $t_0 \prec s \prec t_1$.*

A related but subtly different issue is the problem of *blocking*. By Axiom 4 (RESPECTING ORDER), we know that r_1 in Figure 5 can never be a successor of t. However, there is nothing to prevent us from saying that, since r_1 ends before s, it "blocks" s from being the successor of t, and hence $\text{SUCC}(t, \{ s, r_1 \}) = \emptyset$. In this case we have an element r_1 that is not the successor, but prevents other events from being successors as well. Such semantics is not intuitive and is not found in any of the existing event systems.

AXIOM 7 (NON-BLOCKING). *If $\text{SUCC}(t, \mathcal{F}) \cap \mathcal{A} = \emptyset$, then $\text{SUCC}(t, \mathcal{F}) = \text{SUCC}(t, \mathcal{F} \setminus \mathcal{A})$.*

4.1.3 The $\otimes$ Axioms

The $\otimes$ operator is used to combine time stamps from sequenced events. Hence our first axiom for it is concerned with when sequencing is defined. In particular, $t_0 \otimes t_1$ should only be defined if the t_i are the time stamps to two events that can be sequenced.

AXIOM 8 (CONSERVATIVE COMPOSITION). *$t_0 \otimes t_1$ is defined if and only if $t_0 \prec t_1$.*

Because event systems must process events in real-time, event sequencing should happen in a "timely" manner. In other words, the sequenced event should have a time stamp that allows us to add it to the output stream immediately. For example, suppose we compose two events with time stamps $t_0 = [0, 1]$ and $t_1 = [2, 3]$. We should not allow $t_0 \otimes t_1 = [0, 5]$ as the interval $[4, 4]$ follows t_1, but not $t_0 \otimes t_1$; hence we could not add $t_0 \otimes t_1$ to the stream until we are sure that all events with time $[4, 4]$ have passed. This constraint is implemented in all event systems by ensuring that $t_0 \otimes t_1$ and t_1 always share the same successors.

AXIOM 9 ($\otimes$-ELIMINATION). *Suppose $t_0 \prec t_1$. Then $t_2 \in \text{SUCC}(t_0 \otimes t_1, \mathcal{F})$ if and only if $t_2 \in \text{SUCC}(t_1, \mathcal{F})$.*

4.1.4 The $\mathbb{T}^\circ$ Axioms

The operation $\otimes$ is used to construct time stamps created by the sequencing operation. Intuitively all time stamps should ultimately be derived via $\otimes$ from some universe of "base" time stamps (e.g. the universe of clock ticks that define some event occurrence). These are the time stamps assigned to primitive events; the time stamp for a composite event is constructed by applying $\otimes$ to the base time stamps of the primitive events making up this composite event. We refer to this set of base time stamps as $\mathbb{T}^\circ$.

AXIOM 10 (PRIMITIVE REPRESENTATION). *There is a set $\mathbb{T}^\circ \subseteq \mathbb{T}$ such that*

- *for any $s \in \mathbb{T}^\circ$, there is no $t_0, t_1 \in \mathbb{T}$ with $s = t_0 \otimes t_1$.*
- *for any $t \in \mathbb{T}$, there are $s_i \in \mathbb{T}^\circ$ such that $t = s_0 \otimes \cdots \otimes s_n$.*

PROPOSITION 3. *The set $\mathbb{T}^\circ$ in PRIMITIVE REPRESENTATION is unique. That is, if $\mathbb{T}^\circ$ and $\mathbb{S}^\circ$ both satisfy the conditions above, then $\mathbb{T}^\circ = \mathbb{S}^\circ$.*

In essence, PRIMITIVE REPRESENTATION asserts that $\mathbb{T}$ is a free monoid with respect to $\otimes$ over $\mathbb{T}^\circ$. Note that this axiom only says that base time stamps exist, and does not require them to be points, intervals, or anything in particular. Furthermore, the decomposition in PRIMITIVE REPRESENTATION need not be unique. For example, in Active Office, $[1, 3] = [1, 1] \otimes [3, 3] = [1, 1] \otimes [2, 2] \otimes [3, 3]$.

All existing event systems have a global clock and all time stamps of primitive events are defined in terms of values of this clock. Thus there is an implicit linear order on the base time stamps. Notice that this does not imply that *event* time stamps are linearly ordered. For example, even though the natural numbers are linearly ordered, intervals of natural numbers can overlap and hence are only partially ordered (as pointed out earlier). The underlying assumption of a global clock is formalized by asserting that $\mathbb{T}^\circ$ is isomorphic to the linear order $\mathbb{Z}$.

AXIOM 11 (LINEARITY). *Let $\mathbb{T}^\circ$ be the unique set identified in PRIMITIVE REPRESENTATION. The ordering $\prec$ is an infinite discrete linear ordering on $\mathbb{T}^\circ$.*

From this axiom it may appear that we cannot handle real-valued time stamps. However, we can remove the discrete requirement from LINEARITY provided that we stipulate that all candidate sets are well-founded. If we had a non-well-founded candidate set $\mathcal{F}$ with an infinite descending sequence converging to t, then $\text{SUCC}(t, \mathcal{F})$ would not be well-defined, even though there are elements in $\mathcal{F}$ after t. As $\mathcal{F}$ corresponds to a set of time stamps for incoming events, well-foundedness is a realistic assumption. Furthermore, as all event expressions are finite, there is no distinguishable difference between requiring that $\mathbb{T}^{\circ}$ be discrete and requiring that all $\mathcal{F}$ be well-founded. Therefore, for simplicity, we keep the discreteness assumption.

4.2 Desirable Axioms

All of the axioms in the previous section are satisfied by the existing event systems. However, there are several axioms that we would like our models to satisfy for implementation reasons. In this section we introduce these axioms.

4.2.1 The "Time-Out" Axiom

In Section 2, we saw an important problem that occurs in the SnoopIB system. In SnoopIB, overlapping pairs of events in Query 2 can result in an unbounded number of matches for each purchase. All we need is for each customer to add their item to the shopping cart immediately, and then pay sometime in the future. The payment can be received in an hour, a day, or even years from now; as all these event pairs overlap, they are all successors to the giveaway announcement.

For a more formal illustration of this problem, in SnoopIB,

$$\text{SUCC}([0,0], \{\, [1,x] \mid 1 \leq x \,\}) = \{\, [1,x] \mid 1 \leq x \,\} \tag{2}$$

Hence this definition of successor is very difficult to implement in an event system. Even though the time stamps may be partially ordered, the events necessarily arrive real-time in a linear fashion. In models with interval time stamps, they typically arrive to the stream at the time corresponding to the end of the interval. (This is the time when the event "happens".) Hence for the candidate set $\mathcal{F} = \{\, [1,1], [2,2], [1,3] \,\}$, $[2,2]$ will arrive before an event with time stamp $[1,3]$, even though $[1,3]$ is a successor time stamp to $[0,0]$ and $[2,2]$ is not. In general, there could always be an event with interval $[1,x]$ arriving at some future time x. Hence for every match to E_1, the system might have to keep looking for successors indefinitely. As a result, old query state cannot be garbage collected and memory usage grows without bound. Therefore, we would like an axiom that limits the effect that events with arbitrary long duration can have on the system.

AXIOM 12 (STRONG THICKENING). *Let* $t \in \text{SUCC}(s, A)$. *For any* $u, v \in \mathbb{T}$, *if* $t \prec v$, *then* $\text{SUCC}(s, A) = \text{SUCC}(s, A \cup \{\, u \otimes v \,\})$.

In essence, STRONG THICKENING is a "time-out" axiom. It guarantees that once we see at least one successor, we can ignore any events that happen afterwards.

4.2.2 Associativity

In Section 2 we saw that it would be advantageous for us to associate event sequencing. Our next desirable axiom is one that guarantees associativity. Naively, it would seem to be enough for us to require that $\otimes$ is associative.

AXIOM 13 ($\otimes$-ASSOCIATIVITY). $(t_0 \otimes t_1) \otimes t_2 = t_0 \otimes (t_1 \otimes t_2)$, *for all* $t_i \in \mathbb{T}$.

However, this axiom is satisfied by both SnoopIB and Cayuga, which we have already seen are not associative. In fact, the only systems that violate this axiom are the point models of Snoop and EPL. Recall that we denote $t_0 \otimes t_1 = \bot$ if $t_0 \otimes t_1$ is undefined. So $\otimes$-ASSOCIATIVITY implicitly guarantees that $(t_0 \otimes t_1) \otimes t_2$ is defined exactly when $t_0 \otimes (t_1 \otimes t_2)$ is. In Snoop and EPL, the time stamp $2 \otimes (1 \otimes 3) = 3$ is defined, but $(2 \otimes 1) \otimes 3$ is not. In fact, this is the reason for the observation from [3, 8] that the two sequencings

$$E_1 \,\mathbf{;}(E_2 \,\mathbf{;}\, E_3) \quad \text{and} \quad E_2 \,\mathbf{;}(E_1 \,\mathbf{;}\, E_3) \tag{3}$$

are equivalent for event systems with point time stamps. So while $\otimes$-ASSOCIATIVITY does not give us sequencing associativity, it is important in that it prevents us from sequencing events that should not be sequenced. In fact, we can express this observation as the following proposition.

PROPOSITION 4. $(t_0 \otimes t_1) \otimes t_2$ *is defined if and only if* $t_0 \prec t_1 \prec t_2$.

In order to find the correct axiom for associativity, we first need to formally understand what it means for sequencing to be associative. An event system processes expressions on a stream S of events. Events in a data stream consist of both data fields (which define the type of the event) and a time stamp. We typically denote these elements $\langle a, t \rangle \in D \times \mathbb{T}$ where $a \in D$ is the data and $t \in \mathbb{T}$ is the time stamp. As it is not relevant to our discussion, we make no stipulation on the nature of the data domain D. In traditional event systems D is the finite set of all event symbols, while in parameterized event systems such as Cayuga, D can be an infinite set of data tuples.

Given an event expression E, an event system returns $[\![E]\!]_S$, the set of all events in S that match E. For the sequencing operator, this set is defined as

$$[\![E_1\mathbf{;}E_2]\!]_S = \left\{ \langle a_1 \oplus a_2, t_1 \otimes t_2 \rangle \;\middle|\; \begin{array}{l} \langle a_1, t_1 \rangle \in [\![E_1]\!]_S, \langle a_2, t_2 \rangle \in [\![E_2]\!]_S, \\ t_2 \in \text{SUCC}(t_1, \{ s \mid \langle b, s \rangle \in [\![E_2]\!]_S \}) \end{array} \right\} \tag{4}$$

Note that the data domain of the complex event $E_1\mathbf{;}E_2$ is the set $\{a \oplus b | a \in D_1, b \in D_2\}$, where $a \oplus b$ is some data composition of data values a and b. We give no semantics for this operation as it will not be relevant to the discussion; in practice it is usually tuple concatenation.

From (4), we see that there are actually two parts to ensuring that sequencing is associative. The first is that the data composition $\oplus$ is associative; the second is the associativity of the time stamps. Because the definition of successor only uses time stamps and not data elements, we can safely separate these two components. Some event systems do use data elements in their definition of successor. For example, Active Office uses the element ID of an event to break ties when determining successors. However, we can still separate data from the time stamps in these systems by assuming that the relevant ID information is added as part of the time stamp. As data can be real-valued, this assumption does require our models to support real-valued time stamps. However, as we discussed in Section 4.1.4, this is not a problem.

From the definition of sequencing in (4), associativity requires that for any event expressions E_0, E_1, E_2 and stream S,

$$\begin{aligned} &[\![(E_0\,\mathbf{;}\,E_1)\,\mathbf{;}\,E_2]\!]_S \\ &= \left\{ \left\langle \begin{array}{c} (a_0 \oplus a_1) \oplus a_2, \\ (t_0 \otimes t_1) \otimes t_2 \end{array} \right\rangle \;\middle|\; \begin{array}{l} \langle a_i, t_i \rangle \in [\![E_i]\!]_S, t_1 \in \text{SUCC}(t_0, \mathcal{F}_{E_1}), \\ t_2 \in \text{SUCC}(t_0 \otimes t_1, \mathcal{F}_{E_2}) \end{array} \right\} \\ &= \left\{ \left\langle \begin{array}{c} a_0 \oplus (a_1 \oplus a_2), \\ t_0 \otimes (t_1 \otimes t_2) \end{array} \right\rangle \;\middle|\; \begin{array}{l} \langle a_i, t_i \rangle \in [\![E_i]\!]_S, t_2 \in \text{SUCC}(t_1, \mathcal{F}_{E_2}), \\ t_1 \otimes t_2 \in \text{SUCC}(t_0, \mathcal{F}_{E_1\,;\,E_2}) \end{array} \right\} \\ &= [\![E_0\,\mathbf{;}\,(E_1\,\mathbf{;}\,E_2)]\!]_S \end{aligned} \tag{5}$$

where $\mathcal{F}_E = \{ t \,|\, \langle a, t \rangle \in [\![E]\!]_S \}$. Note that this equation entails a relationship between $\mathcal{F}_{E_i}$ and $\mathcal{F}_{E_1\,;\,E_2}$. For candidate sets $\mathcal{F}_0$, $\mathcal{F}_1$, we define

$$\mathcal{F}_0\,;\mathcal{F}_1 = \{\, t_0 \otimes t_1 \mid t_0 \in \mathcal{F}_0, t_1 \in \textsc{succ}(t_0, \mathcal{F}_1) \,\} \qquad (6)$$

From this observation it is obvious that the data element component of associativity is trivial; we only need to ensure that the composition operator $\oplus$ is itself associative. Therefore we can focus on the time stamp component of associativity. It should be clear then that (5) implies the following axiom.

AXIOM 14 ($\otimes$-DECOMPOSITION). *Suppose $t_0, t_1, t_2 \in \mathbb{T}$, with $t_1 \prec t_2$, and $\mathcal{F}_1, \mathcal{F}_2 \subseteq \mathbb{T}$. Also suppose that $t_2 \in \textsc{succ}(t_1, \mathcal{F}_2)$. Then $t_1 \in \textsc{succ}(t_0, \mathcal{F}_1)$ if and only if $t_1 \otimes t_2 \in \textsc{succ}(t_0, \mathcal{F}_1\,;\mathcal{F}_2)$.*

It is apparent from (5) that both $\otimes$-ASSOCIATIVITY and $\otimes$-DECOMPOSITION are necessary for associativity. The following proposition establishes that they are sufficient as well.

PROPOSITION 5. *Suppose $\mathbb{T}$ is a temporal model satisfying Axiom 9 ($\otimes$-ELIMINATION), Axiom 13 ($\otimes$-ASSOCIATIVITY), and Axiom 14 ($\otimes$-DECOMPOSITION). Let E_1, E_2, E_3 be event expressions, and suppose $\oplus$ is associative over the data elements of the event stream S. Then $[\![(E_0\,;E_1)\,;E_2]\!]_S = [\![E_0\,;(E_1\,;E_2)]\!]_S$.*

5. ANALYSIS OF TEMPORAL MODELS

Now that we have stated our axioms, we would like to find the "best" model that satisfies all of these axioms. Note that none of the definitions of successor in Section 1 satisfy all axioms. Cayuga and Active Office violate $\otimes$-DECOMPOSITION, and hence do not support associativity. SnoopIB and ODE also violate this axiom, and in addition violate STRONG THICKENING. Systems with point time stamps even violate $\otimes$-ASSOCIATIVITY. Hence to satisfy all of the axioms, we need to find a new temporal model for event systems.

In this section we characterize the models that satisfy all of our axioms up to isomorphism. We also identify the trade-offs that the systems in Section 2 make by violating one or more of the desirable axioms.

5.1 Satisfying All Axioms

There is at least one model that satisfies all of the axioms. That is the complete-history model from Section 3.1; we leave verification of this fact as an exercise for the reader. Unfortunately, this particular model is impractical because of its memory requirements. In any event system, each base time stamp (i.e. an element of $\mathbb{T}^\circ$) requires a memory word. A complete history of time stamps for a composite event would require as many words as there are primitive events that form the composite event. This is particularly bad for queries in which the history can grow without bound. In addition to regular sequencing, all of the major event systems have an iterated sequencing operator, similar to Kleene-*. This operator is illustrated by the following stock monitoring query.

Query 3. Notify me when a stock price has been monotonically increasing for at least 30 minutes.

This sequence can be composed of any number of stock quotes. In the complete-history model, we have to store and remember the time stamps for all of the quotes in the sequence.

To get a model that uses bounded memory for time stamps, we need to compress the time stamp representation. For example, interval time stamp models have bounded representation because we can drop any intermediate information. For example,

$$[0,1] \otimes [2,3] \otimes [4,5] = [0,1] \otimes [4,5] = [1,5]$$

Formally, we want some fixed n such that every $t \in \mathbb{T}$ can be written $t = p_0 \otimes \cdots \otimes p_n$ for $p_i \in \mathbb{T}^\circ$. Unfortunately, as the following theorem demonstrates, this is impossible.

THEOREM 1. *Assume $\mathbb{T}$ is a temporal model satisfying Axioms 1-14. For each $t \in \mathbb{T}$, there is a unique sequence $p_0, \ldots, p_n \in \mathbb{T}^\circ$ with $t = p_0 \otimes \cdots \otimes p_n$, where n depends upon t.*

From this theorem we see that any temporal model that satisfies all of the axioms must keep a complete history of the time stamps. Intuitively this is the case because any time stamp in the history can be used to determine its order with respect to another history. From this theorem, we can prove an even stronger result, namely that complete-history model is the *only* model of the axioms, up to isomorphism.

THEOREM 2. *Let $\mathbb{T}$ be a temporal model satisfying Axioms 1-14. Let $\mathbb{S}$ be the complete-history model. If we identify $\mathbb{T}^\circ$ with $\mathbb{Z}$, the mapping $t_0 \otimes \cdots \otimes t_n \mapsto \sigma$ where $\sigma(i) = t_i$ is an isomorphism.*

As an interesting technical aside, Theorem 2 demonstrates that our axioms are a sound and complete axiomatization of the theory of the complete-history model. They are sound because the complete-history model satisfies the axioms. They are complete because they have only one model up to isomorphism, and so their logical consequences are exactly those statements true in the complete-history model. However, this result is only of theoretical interest as we are not interested in using our axioms for validation, but only in characterizing those temporal models that are acceptable.

As these two theorems are the primary result of this paper, the remainder of this section is a outline of their proof. We present the important steps of the proof as propositions, which are themselves stated without proof.

5.1.1 *Proof of Theorem 1*

To prove Theorem 1, we will assume from here on that $\mathbb{T}$ is a temporal model satisfying all of the axioms (Axioms 1-14). Before we prove Theorem 1, we first need a way of distinguishing time stamps. To do this, we introduce two equivalence relations.

Definition 1. For any $t_0, t_1 \in \mathbb{T}$, we say t_0, t_1 have the *same end time* (denoted $t_0 \sim_E t_1$) when, for any $s \in \mathbb{T}$, $t_0 \prec s$ if and only if $t_1 \prec s$. Similarly, t_0, t_1 have the *same start time* (denoted $t_0 \sim_S t_1$) when, for any $s \in \mathbb{T}$, $s \prec t_0$ if and only if $s \prec t_1$.

Intuitively, these relations give us an abstract way to identify the start and end time of a time stamp without having to assume our time stamps are actually intervals. The following propositions below guarantee that every time stamp t has a unique start time $t \sim_S s_0 \in \mathbb{T}^\circ$, and a unique end time $t \sim_E s_1 \in \mathbb{T}^\circ$. Thus we can unambiguously speak of a time stamp "interval" in an abstract sense.

PROPOSITION 6. *Suppose $t_0 \prec t_1$. Then $t_0 \otimes t_1 \sim_E t_1$ and $t_0 \otimes t_1 \sim_S t_0$.*

PROPOSITION 7. *Suppose $p_0, p_1 \in \mathbb{T}^\circ$. Then $p_0 = p_1$ if and only if $p_0 \sim_E p_1$. Similarly, $p_0 = p_1$ if and only if $p_0 \sim_S p_1$*

To prove Theorem 1, we will need to induct over the length of a decomposition $t = p_0 \otimes \cdots \otimes p_n$ of t. We can reduce a time stamp to one with smaller decomposition length by using Axiom 14 ($\otimes$-DECOMPOSITION). However, in order to make use of this axiom, we need to understand what happens when we apply SUCC twice. Proposition 8 tells us that all of the successors have the same end time. Hence by Axiom 9 ($\otimes$-ELIMINATION), if t_0, t_1 are both successors of s from the same candidate set, $\text{SUCC}(t_0, \mathcal{F}) = \text{SUCC}(t_1, \mathcal{F})$.

PROPOSITION 8. *Suppose* $t_0, t_1 \in \text{SUCC}(s, \mathcal{F})$. *Then* $t_0 \sim_E t_1$.

COROLLARY 1. *Suppose* $t_0, t_1 \in \text{SUCC}(s, \mathcal{F})$ *with* $t_0, t_1 \in \mathbb{T}^\circ$. *Then* $t_0 = t_1$.

While these propositions appear fairly technical, they are enough to prove that every time stamp has at most one successor. This is such a powerful result that we state it as a theorem in its own right.

THEOREM 3. *Let* $s, t_0, t_1 \in \mathbb{T}$. *For all* $\mathcal{F} \subseteq \mathbb{T}$, *if* $t_0, t_1 \in \text{SUCC}(s, \mathcal{F})$, *then* $t_0 = t_1$.

PROOF. Using Axiom 10 (PRIMITIVE REPRESENTATION), suppose $t_0 = u_0 \otimes \cdots \otimes u_n$, $t_1 = v_0 \otimes \cdots \otimes v_m$, where $u_i, v_j \in \mathbb{T}^\circ$. We proceed by induction on n and m. The case for $n, m = 1$ is covered by Corollary 1. Suppose we know it is true for any $n, m \leq k$, and take some t_0, t_1 with $n, m \leq k + 1$. Without loss of generality, $m = k + 1$.

First we consider the case for $n > 1$. By Propositions 7 and 8, we have that $u_n = v_m$. Let $q = u_n$, and let $p_0 = u_0 \otimes \cdots \otimes u_{n-1}$ and $p_1 = v_0 \otimes \cdots \otimes v_{m-1}$. Hence $t_0 = p_0 \otimes q$, $t_1 = p_1 \otimes q$. By Axiom 5 (THINNING), $p_0 \otimes q, p_1 \otimes q \in \text{SUCC}(s, \{ p_0 \otimes q, p_1 \otimes q \})$. Hence $p_0, p_1 \in \text{SUCC}(s, \{ p_0, p_1 \})$ and $q \in \text{SUCC}(p_i, \{ q \})$ by Axiom 14 ($\otimes$-DECOMPOSITION). Then $p_0 = p_1$ by our induction hypothesis, and so we are done.

Now suppose $n = 1$, i.e., $t_0 \in \mathbb{T}^\circ$. By Propositions 8 and 7, $v_m = t_0$. Now let $r \prec s$. Then $s \in \text{SUCC}(r, \{ s \})$. As $\text{SUCC}(s, \{ t_0, t_1 \}) = \{ t_0, t_1 \}$, $\otimes$-DECOMPOSITION gives us

$$\begin{aligned} &\{ s \otimes t_0, s \otimes v_0 \otimes \cdots \otimes v_{m-1} \otimes t_0 \} \\ &\qquad = \text{SUCC}(r, \{ s \otimes t_0, s \otimes v_0 \otimes \cdots \otimes t_0 \}) \end{aligned}$$

Again by $\otimes$-DECOMPOSITION, we have that $s, s \otimes \cdots \otimes v_{m-1} \in \text{SUCC}(r, \{ s, s \otimes \cdots \otimes v_{m-1} \})$. However, $s \prec v_0 \prec v_{m-1}$, and so this case contradicts Proposition 8. $\square$

To prove Theorem 1, we need one more result. Proposition 9 establishes that a single usage of $\otimes$ cannot collapse two different time stamps into a single time stamp.

PROPOSITION 9. *Suppose* $t_0, t_1, s \in \mathbb{T}$ *with* $t_0, t_1 \prec s$ *and* $t_0 \neq t_1$. *Then* $t_0 \otimes s \neq t_1 \otimes s$

PROOF OF THEOREM 1. Let $t_0 = u_0 \otimes \cdots \otimes u_n$, $t_1 = v_0 \otimes \cdots \otimes v_m$. Also suppose that $n \neq m$, or $n = m$ and $u_i \neq v_i$ for some $i \leq n$. We need to show that $t_0 \neq t_1$.We proceed by induction on n and m. The case for $n = m = 1$ is obvious. So suppose we know that the $t_0 \neq t_1$ for $n, m \leq k$. Let $m = k + 1$ and $n \leq m$.

First we consider the case where $n > 1$. Suppose for a contradiction that $t_0 = t_1$. Then by Proposition 6 and 7, $u_n = v_m = q$. Let $p_0 = u_0 \otimes \cdots \otimes u_{n-1}$ and $p_1 = v_0 \otimes \cdots \otimes v_{m-1}$. By Proposition 9, $p_0 = p_1$, contradicting our induction hypothesis.

Now suppose $n = 1$. Then $n \neq m$ and so $u_0 \neq v_0$. However, $t_0 \sim_S t_1$ and so this contradicts Proposition 7. $\square$

5.1.2 *Proof of Theorem 2*

Theorem 3 proves that there is at most one successor at any time. While this is true in complete-history, this is not enough to establish complete-history as the unique temporal model. We need to prove that this unique successor is structurally identical to the one in complete-history. In particular, we need to know that the partial order $\prec$ behaves just like the interval partial order $<$. This fact follows from the next proposition.

PROPOSITION 10. *Suppose* $t_0, t_1 \in \mathbb{T}$ *with* $s_0, s_1 \in \mathbb{T}^\circ$ *such that* $t_0 \sim_E s_0$ *and* $t_1 \sim_S s_1$. *Then* $t_0 \prec t_1$ *if and only if* $s_0 \prec s_1$. *In other words,* t_1 *follows* t_0 *if and only if the start time of* t_1 *follows the end time of* t_0.

Thus the only difference between any two models that satisfy all the axioms could lie in how they break ties between overlapping "intervals". We therefore need to establish that any two such models must break ties in the same way. Complete-history uses the linear order $\sqsubseteq$ to break ties. Because of Theorem 1, we can extend the definition of $\sqsubseteq$ to arbitrary temporal models in the usual way, identifying $t_0 \otimes \cdots \otimes t_n$ with σ as specified in Theorem 2. As the following proposition demonstrates, for small candidacy sets, $\sqsubseteq$ is our only option to choose a successor.

PROPOSITION 11. *Suppose* $\text{SUCC}(s, \{ t_0, t_1 \}) = \{ t_0 \}$ *with* $t_i \in \mathbb{T}$. *If* $s \prec t_1$, *then* $t_0 \sqsubseteq t_1$.

Theorem 3 guarantees that there is at most one successor, and Proposition 11 suggests that when we have a successor, we always use $\sqsubseteq$ to determine which one it is. Therefore, to prove Theorem 2, we only need to guarantee that, when there is some $s \in \mathcal{F}$ with $t \prec s$, then there is *at least* one element in $\text{SUCC}(t, \mathcal{F})$. Fortunately, this follows from Axiom 7 (NON-BLOCKING).

PROOF OF THEOREM 2. By Theorem 1, the mapping $t_0 \otimes \cdots \otimes t_n \mapsto \sigma$ is well-defined; it is clearly a bijection. We need to show that this mapping preserves the successor operation. We already know from Proposition 10 that $\prec$ and the interval order are the same. So we need only show that we break ties properly on all candidate sets.

Suppose $t_0 \in \text{SUCC}(s, \mathcal{F})$ with $s \prec t_1 \in \mathcal{F}$. Applying NON-BLOCKING to Proposition 11, we see that $t_0 \sqsubset t_1$ whenever $t_0 \neq t_1$. Thus $\sqsubseteq$ is the only way to break ties over arbitrary candidate sets. The only thing left to show is that $\text{SUCC}(s, \mathcal{F}) \neq \emptyset$ whenever $t \in \mathcal{F}$ with $s \prec t$. Suppose for a contradiction that $t \in \mathcal{F}$ with $s \prec t$, but $\text{SUCC}(s, \mathcal{F}) = \emptyset$. Then by NON-BLOCKING, $\text{SUCC}(s, \{ t \}) = \emptyset$. But this contradicts Proposition 1. $\square$

5.2 Relaxing the Desirable Axioms

The moral of Section 5.1 is that to satisfy all axioms, the temporal model needs to rely on time stamps of unbounded size. If we want models with time stamps of bounded size, we need to relax our demands. This means that our primary goal now is to identify the least number of axioms that we need to relax in order to get such temporal model, e.g., an interval model.

Definition 2. An *interval model* is a model $\mathbb{T}$ in which

$$t_0 \otimes t_1 \otimes t_2 = t_0 \otimes t_2 \text{ for any } t_0, t_1, t_2 \in \mathbb{T} \qquad (7)$$

An interval model allows us the most compact representation, as we only need to remember two primitive time stamps for each element of $\mathbb{T}^\circ$ (see Proposition 12 below). While this may seem like a fairly extreme restriction, our results in this section generalize for any model with bounded representation (i.e. there is some fixed n such that for each t, $t = p_0 \otimes \cdots \otimes p_n$ for some $p_i \in \mathbb{T}^\circ$). Thus we consider only interval models in order to simplify our analysis.

We still require any temporal model to satisfy the standard axioms (Axioms 1-11). Furthermore, of all the desirable axioms in Section 4.2, we do not want to drop Axiom 13 ($\otimes$-ASSOCIATIVITY). That axiom is necessary to prevent the pathological behavior equating the two expressions in (3), which is clearly undesirable. Therefore, in this section we will determine what types of interval models we get if we relax either Axiom 12 (STRONG THICKENING) or Axiom 14 ($\otimes$-DECOMPOSITION).

As many of the of the propositions in Section 5.1 did not require the use of axioms in Section 4.2, we can still say a lot about these models. In particular, Propositions 6 and 7 require neither STRONG THICKENING nor $\otimes$-DECOMPOSITION. Therefore, we can prove the following result, which shows that our name "interval model" is indeed appropriate.

PROPOSITION 12. *Let $\mathbb{T}$ be any interval model satisfying the accepted axioms and let $t \in \mathbb{T} \setminus \mathbb{T}^\circ$. There are unique $t_0, t_1 \in \mathbb{T}^\circ$ such that $t_0 \sim_S t$, $t_1 \sim_E t$. Furthermore, $t = t_0 \otimes t_1$.*

5.2.1 *Relaxing* STRONG THICKENING

STRONG THICKENING is an important part of the proof of Theorem 1, which prevents any model of the axioms from being an interval model. As an illustrative example, suppose that t has two representations

$$t = t_0 \otimes p \otimes t_1 = t_0 \otimes q \otimes t_1$$

as is the case in an interval model. Also suppose that $p \prec q$. Then by $\otimes$-DECOMPOSITION, for any $s \prec t$,

$$\text{SUCC}(s, \{\, t_0 \otimes p, t_0 \otimes q \,\}) = \{\, t_0 \otimes p, t_0 \otimes q \,\}$$

However, this violates STRONG THICKENING, since $t_0 \otimes p \prec q$ by $\otimes$-ELIMINATION.

This example suggests that might be able to get an associative interval model by relaxing this axiom. However, as we saw in Section 2, none of the existing interval models are associative. Furthermore, as the following theorem shows, there is no way to get an associative interval model of the standard axioms.

THEOREM 4. *There is no interval model of the standard axioms that is also associative.*

This theorem is true because any associative model must satisfy both Axiom 13 ($\otimes$-ASSOCIATIVITY) and Axiom 14 ($\otimes$-DECOMPOSITION). And any interval model of these two axioms can never have more than one successor. Suppose event E_1 has time stamp $[0, 0]$ and there are two instances of $(E_2 ; E_3)$ with time stamps $[1, 3]$ and $[2, 3]$, respectively. We cannot tell from the time stamp $[1, 3]$ whether E_2 had time stamp $[1, 1]$ or $[1, 2]$. So if we choose both the event at $[1, 3]$ and the one at $[2, 3]$ as the next occurrence of $(E_2 ; E_3)$, and the two E_2 events have time stamps $[1, 1]$ and $[2, 2]$, respectively, then we must choose both of them as the next E_2 event after E_1. However, this violates Axiom 6 (THICKENING), which is an standard axiom. In fact, as the following proposition shows, we can never limit the successor in an associative interval model to a single choice.

PROPOSITION 13. *Let $\mathbb{T}$ be any interval model of the accepted axioms which is associative. Let $t \in \mathbb{T}$ and let $\mathcal{F} \subseteq \mathbb{T}$ be such that $s_1 \sim_E s_2$ and $t \prec s_1$ for all $s_1, s_2 \in \mathcal{F}$. Then $\text{SUCC}(t, \mathcal{F}) = \mathcal{F}$.*

As a result of Theorem 4, there is no obvious benefit for relaxing STRONG THICKENING.

5.2.2 *Relaxing* $\otimes$-DECOMPOSITION

Even though there is no hope for an associative interval model, we may still be able to construct an interval model that *approximates* associativity. All of the interval models in Section 2 satisfy $\otimes$-ASSOCIATIVITY. The only problem is how we treat the candidate sets of composite events. For full associativity, we require $[\![(E_0 ; E_1) ; E_2]\!]_S = [\![E_0 ;(E_1 ; E_2)]\!]_S$. Suppose instead that we have a model in which $[\![(E_0 ; E_1) ; E_2]\!]_S \supseteq [\![E_0 ;(E_1 ; E_2)]\!]_S$. In such a model we could rewrite the expression $E_0 ;(E_1 ; E_2)$ as a left-associated expression, and eliminate false positives in post-processing. However, even this is impossible in an interval model.

THEOREM 5. *Let $\mathbb{T}$ be an interval model satisfying all axioms but $\otimes$-DECOMPOSITION. Then there are expressions E_i and a stream S such that $[\![(E_0 ; E_1) ; E_2]\!]_S \not\supseteq [\![E_1 ;(E_1 ; E_2)]\!]_S$.*

It is also possible to approximate associativity when $[\![(E_0 ; E_1) ; E_2]\!]_S \subseteq [\![E_0 ;(E_1 ; E_2)]\!]_S$. This property guarantees that we can rewrite $(E_0 ; E_1) ; E_2$ as a right-associated expression, and eliminate the false positives in post-processing, thus allowing us to take advantage of those cases where the pattern E_2 is selective, but E_0 and E_1 are not. Furthermore, this property guarantees that we will never produce false positives if we rewrite $E_0 ;(E_1 ; E_2)$ as a left-associated expression.

Satisfying this half of associativity requires the forward direction of $\otimes$-DECOMPOSITION, namely

$$\begin{aligned} t_1 \in \text{SUCC}(t_0, \mathcal{F}_1),\ t_2 \in \text{SUCC}(t_1, \mathcal{F}_2) \\ \Rightarrow t_1 \otimes t_2 \in \text{SUCC}(t_0, \mathcal{F}_1 ; \mathcal{F}_2) \end{aligned} \tag{8}$$

It is easy to verify that Cayuga has this property. Furthermore, any interval model with this property must accept almost all $\prec$-successor time stamps with the minimal end time, and thus is at most a minor variation of the Cayuga model. In particular, the following proposition demonstrates that a model like the one used in Active Office does not approximate associativity in either direction.

PROPOSITION 14. *Let $\mathbb{T}$ be an interval model satisfying* (8) *and all axioms but* $\otimes$-DECOMPOSITION. *Let $\mathcal{F}$ be a candidate set in which $t = r_0 \otimes r_1 \otimes r_2$ for every $t \in \mathcal{F}$. Then*

$$\text{SUCC}(s, \mathcal{F}) = \{t | s \prec t \in \mathcal{F} \textit{ and end time } t \sim_E q \in \mathbb{T}^\circ \textit{ is least}\}.$$

It is possible for an interval model satisfying (8) to have arbitrary behavior on time stamps of very short duration (i.e., the composition of one or two base time stamps), as they are too short for associativity to apply. However, in addition to (8), Cayuga also has a very weak form of associativity that applies when it is sequencing a stream of events with itself (i.e. an expression of the form $E_1 ;(E_2 ; E_2)$). In Cayuga, if there are no overlapping E_2 events in S, then $[\![E_1 ;(E_2 ; E_2)]\!]_S = [\![(E_1 ; E_2) ; E_2]\!]_S$. This property follows from a weaker version of $\otimes$-DECOMPOSITION, namely

$$\text{SUCC}(t, \{\, p_i \otimes s \,\}_{i \in I}) = \text{SUCC}(t, \{\, p_i \,\}_{i \in I}) ; \{\, s \,\} \tag{9}$$

This property, in addition to (8), uniquely characterizes Cayuga up to isomorphism, suggesting that this temporal model is the closest we can get to an associative interval model.

THEOREM 6. *Let $\mathbb{T}$ be an interval model satisfying* (8), (9) *and all axioms but* $\otimes$-DECOMPOSITION. *Let $\mathbb{S}$ be the Cayuga temporal model. If we identify $\mathbb{T}^\circ$ with $\mathbb{Z}$, the mapping $t_0 \otimes t_1 \mapsto [t_0, t_1] \in \mathbb{S}$, where $t_i \in \mathbb{T}^\circ$, is an isomorphism.*

Again, we note as an aside that Theorem 6 shows that these properties are a sound and complete axiomatization of the weakly associative interval time stamps. We also note that SnoopIB and ODE satisfy both (8) and (9), and thus approximate associativity equally well as Cayuga. Still, they do not satisfy STRONG THICKENING. As our results show, there is no gain from eliminating STRONG THICKENING and hence there is no apparent advantage to adopting the temporal models of SnoopIB or ODE over Cayuga.

6. RELATED WORK

Initial implementations of event composition systems, such as Snoop [6] and EPL [14], used a linear temporal model based on point time stamps. Results from the Knowledge Representation community [3, 8] demonstrated that this temporal model did not correctly implement the semantics of sequencing in right-associated queries. Other attempts at event systems [2, 7, 9, 15] all use interval or history models. However, there has been no research into which definition of successor is most appropriate.

The work on EPL [14] is particularly notable as it provides a formal semantics for event languages. However, even though the language is well-defined, it still exhibits unusual behavior like equating the queries in (3). Instead of presenting yet another formal semantics, our work in this paper has been to determine criteria for evaluating and comparing alternate semantics.

The theory of temporal logic has covered many aspects of temporal models; an excellent survey can be found in van Benthem [16]. Bohlen et al [5] have examined the difference between point and interval models in database systems. Our temporal model is a general framework that includes all of these types of models, and many of our axioms in Section 4.1 were motivated by work in this area. To our knowledge, our paper is the first formulation of a temporal model that examines the definition of a successor operation different from the usual one defined by the partial order on time.

Kraemer and Seeger [10] have examined the difficulty of implementing a window join operation on streaming data with interval time stamps. However, their analysis only looks at implementing a specific temporal model, and is not an attempt to characterize all possible implementations, such as we have done in this paper.

Finally, there has been much work on the theory of specific temporal models for event systems. Interval temporal logic [13] is a framework for first-order reasoning about intervals. Bickford and Constable [4] also have a logic for reasoning about events in general distributed processes. However, these approaches assume a fixed temporal model and provide rules for making inferences within that model. Our approach differs in that it is answers a higher level question; we do not assume a fixed temporal model, but use generic properties of event systems to reason which temporal models are best.

7. CONCLUSIONS AND FUTURE WORK

While our approach has been motivated by practical implementation concerns, we have attempted to give a formal and rigorous analysis of the different ways in which we can define a sequencing operator in event composition systems. Admitting that two of the axioms in Section 4.2 are controversial, we have identified two canonical temporal models. One of the two models — complete-history — has serious implementation issues because it requires time stamps of unbounded size. The interval-based time stamp model of Cayuga appears to be the best trade-off between ease of implementation and support of sequencing associativity and right-associated queries.

Notice that our results were obtained for what one might call a “minimal CEP system”, which only has the sequencing operator. Considering additional operators, and hence possibly adding more axioms about their properties, can only introduce further constraints that limit the choice of temporal models. Hence intuitively, the best temporal model identified for this minimal system constitutes the ideal case for any CEP system with additional operators.

There are two axioms in Section 4.1 which, while accepted by all event composition systems, are controversial in the temporal logic community. In particular, while Axiom 11 (LINEARITY) is appropriate for synchronous event systems, it is not applicable to distributed event systems as initially studied by Lamport [11] and later by Liebig et al [12]. Future work is needed to determine the effect of removing this axiom from our framework.

An even more interesting solution to the synchronous assumption would be to remove both LINEARITY and Axiom 10 (PRIMITIVE REPRESENTATION). While the base time stamps are fundamental to our arguments, we can artificially construct them as equivalence classes over the relations $\sim_S$ and $\sim_E$. Further research is needed to determine what temporal models arise when we extend an existing model with these equivalence classes as time stamps.

8. REFERENCES

[1] Complex event processing. www.complexevents.com.

[2] R. Adaikkalavan and S. Chakravarthy. SnoopIB: Interval-based event specification and detection for active databases. In *Proc. ADBIS*, pages 190–204, 2003.

[3] J. Allen and G. Ferguson. Actions and events in interval temporal logic. Technical Report 521, University of Rochester, jul 1994.

[4] Mark Bickford and Robert Constable. A logic of events. Technical Report 2003-1893, Cornell University, 2003.

[5] M. H. Bohlen, R. Busatto, and C. S. Jensen. Point-versus interval-based temporal data models. In *Proc. ICDE*, pages 192–200, 1998.

[6] S. Chakravarthy, V. Krishnaprasad, E. Anwar, and S.-K. Kim. Composite events for active databases: Semantics, contexts and detection. In *Proc. VLDB*, pages 606–617, 1994.

[7] A. Demers, J. Gehrke, M. Hong, M. Riedewald, and W. White. Towards expressive publish/subscribe systems. In *Proc. EDBT*, pages 627–644, 2006.

[8] A. Galton and J. C. Augusto. Two approaches to event definition. In *Proc. DEXA*, pages 547–556, 2002.

[9] N. H. Gehani, H. V. Jagadish, and O. Shmueli. Composite event specification in active databases: Model and implementation. In *Proc. VLDB*, pages 327–338, 1992.

[10] J. Kraemer and B. Seeger. A temporal foundation for continuous queries over data streams. In *Proc. COMAD*, pages 72–80, January 6-8, 2005.

[11] Leslie Lamport. Time, clocks, and the ordering of events in a distributed system. *Commun. ACM*, 21(7):558–565, 1978.

[12] C. Liebig, M. Cila, and A. Buchmann. Event composition in time-dependent distributed systems. In *Proc. Int. Conf. on Cooperative Information Systems (CoopIS)*, pages 70–78. IEEE Computer Society, 1999.

[13] Ben Moszkowski. *Executing Temporal Logic Programs*. Cambridge University Press, 1986.

[14] I. Motakis and C. Zaniolo. Formal semantics for composite temporal events in active database rules. *Journal of Systems Integration*, 7(3-4):291–325, 1997.

[15] P. R. Pietzuch, B. Shand, and J. Bacon. A framework for event composition in distributed systems. In *Proc. ACM/IFIP/USENIX International Middleware Conference*, pages 62–82, 2003.

[16] J. van Benthem. *The Logic of Time*. Kluwer Academic Publishers, 2nd edition edition, 1991.

[17] W. White, M. Riedewald, J. Gehrke, and A. Demers. What's “next”? Technical Report TR2006-2033, Cornell University, 2006. http://techreports.library.cornell.edu.

[18] E. Wu, Y. Diao, and S. Rizvi. High-performance complex event processing over streams. In *Proc. SIGMOD*, pages 407–418, 2006.

APPENDIX

A. PROOFS OMITTED FROM PAPER

A.1 Propositions from Section 4

In this section, we present the proofs of all of the propositions posed in Section 4. All of these proofs follow from algebraic manipulation and direct application of the axioms, and so are presented without comment.

PROPOSITION 1. *For any* $t_0, t_1 \in \mathbb{T}$, $t_0 \prec t_1$ *if and only* $t_1 \in \textsc{succ}(t_0, \{ t_1 \})$.

PROOF. By Axiom 4 (RESPECTING ORDER), $t_0 \prec t_1$ if and only there is some $\mathcal{F}$ such that $t_1 \in \textsc{succ}(t_0, \mathcal{F})$. By Axiom 5 (THINNING), we can choose $\mathcal{F} = \{ t_1 \}$. $\square$

PROPOSITION 2. *If* $t_1 \in \textsc{succ}(t_0, \mathcal{F})$*, then* $t_0 \prec t_1$ *and there is no* $s \in \mathcal{F}$ *with* $t_0 \prec s \prec t_1$.

PROOF. Suppose that there is such an s. As $s \in \mathcal{F}$, we get $t_1 \in \textsc{succ}(t_0, \{ s, t_1 \})$ from Axiom 5 (THINNING). As $t_0 \prec s$, $\{ s \} = \textsc{succ}(t_0, \{ s \})$ by Proposition 1. Hence $\{ s \} = \textsc{succ}(t_0, \{ s, t_1 \})$, by Axiom 6 (THICKENING), a contradiction. $\square$

PROPOSITION 3. *The set* $\mathbb{T}^\circ$ *in* PRIMITIVE REPRESENTATION *is unique.*

PROOF. Let $\mathbb{T}_1$ and $\mathbb{T}_2$ both satisfy the properties of $\mathbb{T}^\circ$ in Axiom 10 (PRIMITIVE REPRESENTATION), and suppose $\mathbb{T}_1 \neq \mathbb{T}_2$. Without loss of generality, there is some $s \in \mathbb{T}_1$ with $s \notin \mathbb{T}_2$. As $s \notin \mathbb{T}_2$, there is some $p_0, \ldots, p_n \in \mathbb{T}_2$, $n \geq 1$ such that $s = p_0 \otimes (p_1 \otimes \cdots \otimes p_n)$. However, as $s \in \mathbb{T}_1$, this is a contradiction. $\square$

PROPOSITION 4. $(t_0 \otimes t_1) \otimes t_2$ *is defined if and only if* $t_0 \prec t_1 \prec t_2$.

PROOF. Suppose $(t_0 \otimes t_1) \otimes t_2$ is defined. Then $t_0 \prec t_1$ by Axiom 8 (CONSERVATIVE COMPOSITION). Furthermore, by Axiom 13 ($\otimes$-ASSOCIATIVITY), we know that $t_0 \otimes (t_1 \otimes t_2)$ is defined and hence $t_1 \prec t_2$.

Now suppose $t_0 \prec t_1 \prec t_2$. By CONSERVATIVE COMPOSITION, $t_0 \otimes t_1$ is defined, and hence $t_0 \otimes t_1 \prec t_2$ by Axiom 9 ($\otimes$-ELIMINATION). Therefore, $(t_0 \otimes t_1) \otimes t_2$ is defined. $\square$

PROPOSITION 5. *Suppose* $\mathbb{T}$ *is a temporal model satisfying Axiom 9 (*$\otimes$*-*ELIMINATION*), Axiom 13 (*$\otimes$*-*ASSOCIATIVITY*), and Axiom 14 (*$\otimes$*-*DECOMPOSITION*). Let* E_1, E_2, E_3 *be event expressions, and suppose* $\oplus$ *is associative over the data elements of the event stream* S*. Then* $[\![(E_0 ; E_1) ; E_2]\!]_S = [\![E_0 ;(E_1 ; E_2)]\!]_S$.

PROOF. Suppose that

$$\langle (a_0 \oplus a_1) \oplus a_2, (t_0 \otimes t_1) \otimes t_2 \rangle \in [\![(E_0 ; E_1) ; E_2]\!]_S$$

with $\langle a_i, t_i \rangle \in [\![E_i]\!]_S$. As $\oplus$ is associative, and $\otimes$ is associative by Axiom 13 ($\otimes$-ASSOCIATIVITY)

$$\begin{aligned} &\langle (a_0 \oplus a_1) \oplus a_2, (t_0 \otimes t_1) \otimes t_2 \rangle \\ &\quad = \langle a_0 \oplus (a_1 \oplus a_2), t_0 \otimes (t_1 \otimes t_2) \rangle \end{aligned} \tag{10}$$

Furthermore, $t_1 \in \textsc{succ}(t_0, \mathcal{F}_{E_1})$ and $t_2 \in \textsc{succ}(t_0 \otimes t_1, \mathcal{F}_{E_2})$. By Axiom 9 ($\otimes$-ELIMINATION), we have $t_2 \in \textsc{succ}(t_1, \mathcal{F}_{E_2})$. Hence $t_1 \otimes t_2 \in \textsc{succ}(t_0, \mathcal{F}_{E_1 ; E_2})$ by Axiom 14 ($\otimes$-DECOMPOSITION), and thus

$$\langle (a_0 \oplus a_1) \oplus a_2, (t_0 \otimes t_1) \otimes t_2 \rangle \in [\![E_0 ;(E_1 ; E_2)]\!]_S$$

Now suppose

$$\langle a_0 \oplus (a_1 \oplus a_2), t_0 \otimes (t_1 \otimes t_2) \rangle \in [\![E_0 ;(E_1 ; E_2)]\!]_S$$

Again, $\otimes$-ASSOCIATIVITY gives us (10). Furthermore, $t_2 \in \textsc{succ}(t_1, \mathcal{F}_{E_2})$ and $t_1 \otimes t_2 \in \textsc{succ}(t_0, \mathcal{F}_{E_1 ; E_2})$. So $t_1 \in \textsc{succ}(t_0, \mathcal{F}_{E_1})$ by $\otimes$-DECOMPOSITION. $\square$

A.2 Propositions from Section 5.1

Once we are given the definition of the equivalence relations $\sim_S$ and $\sim_E$, the propositions in this section all follow from algebraic manipulation and direct application of the axioms. Therefore, they are presented without comment.

PROPOSITION 6. *Suppose* $t_0 \prec t_1$*. Then* $t_0 \otimes t_1 \sim_E t_1$ *and* $t_0 \otimes t_1 \sim_S t_0$.

PROOF. $t_0 \otimes t_1 \sim_E t_1$ is immediate from Axiom 9 ($\otimes$-ELIMINATION), so we need only prove $t_0 \otimes t_1 \sim_S t_0$. First suppose $s \prec t_0 \otimes t_1$. By Axiom 8 (CONSERVATIVE COMPOSITION), $s \otimes (t_0 \otimes t_1)$ is defined. Thus $s \prec t_0$ by Axiom 13 ($\otimes$-ASSOCIATIVITY) and Proposition 4. Now suppose $s \prec t_0$. By Proposition 4, $(s \otimes t_0) \otimes t_1$ is defined. So $s \otimes (t_0 \otimes t_1)$ is defined by $\otimes$-ASSOCIATIVITY, and hence $s \prec t_0 \otimes t_1$. $\square$

PROPOSITION 7. *Suppose* $p_0, p_1 \in \mathbb{T}^\circ$*. Then* $p_0 = p_1$ *if and only if* $p_0 \sim_E p_1$*. Similarly,* $p_0 = p_1$ *if and only if* $p_0 \sim_S p_1$

PROOF. If $t_0 = t_1$ then $t_0 \sim_E t_1$ is clear. Suppose then that $t_0 \sim_E t_1$ but $t_0 \neq t_1$. By Axiom 11 (LINEARITY) we can assume $t_0 \prec t_1$ without loss of generality. But as $t_0 \sim_E t_1$, $t_1 \prec t_1$, which contradicts Axiom 2 (IRREFLEXIVITY).

The proof for $\sim_S$ is analogous. $\square$

PROPOSITION 8. *Suppose* $t_0, t_1 \in \textsc{succ}(s, \mathcal{F})$*. Then* $t_0 \sim_E t_1$.

PROOF. Suppose that $t_0, t_1 \in \textsc{succ}(s, \mathcal{F})$ with $t_0 \not\sim_E t_1$. As elements of $\mathbb{T}$ are built up from $\mathbb{T}^\circ$, we know from Proposition 6 that there are $p_0, p_1 \in \mathbb{T}^\circ$ with $p_i \sim_E t_i$. As $\sim_E$ is an equivalence relation, $p_0 \not\sim_E p_1$. So from Axiom 13 ($\otimes$-ASSOCIATIVITY) and Axiom 11 (LINEARITY), we can assume without loss of generality that $p_0 \prec p_1$. Thus as $p_0 \sim_E t_0$, we have $t_0 \prec p_1$. We now consider two cases.

First, suppose $t_1 \in \mathbb{T}^\circ$. In this case $t_1 = p_1$ and so $t_0 \prec t_1$. As $t_0, t_1 \in \textsc{succ}(s, \mathcal{F})$, $\{ t_0, t_1 \} = \textsc{succ}(s, \{ t_0, t_1 \})$ by Axiom 3 (CANDIDATE PRESENCE) and Axiom 5 (THINNING). Then, again by these two axioms, $\{ t_0 \} = \textsc{succ}(s, \{ t_0 \})$. But $t_0 \prec t_1$, and this so this contradicts Axiom 6 (THICKENING).

Now suppose $t_1 \notin \mathbb{T}^\circ$. We write $t_1 = v_0 \otimes \cdots \otimes v_m \otimes p_1$ where $v_i \in \mathbb{T}^\circ$. Again by CANDIDATE PRESENCE and THINNING we have that $\{ t_0, t_1 \} = \textsc{succ}(s, \{ t_0, t_1 \})$ and $\{ t_0 \} = \textsc{succ}(s, \{ t_0 \})$. But as $t_0 \prec p_1$, Axiom 12 (STRONG THICKENING) gives us $\{ t_0 \} = \textsc{succ}(s, \{ t_0, t_1 \})$, a contradiction. $\square$

COROLLARY 1. *Suppose* $t_0, t_1 \in \textsc{succ}(s, \mathcal{F})$ *with* $t_0, t_1 \in \mathbb{T}^\circ$*. Then* $t_0 = t_1$.

PROOF. Apply Proposition 7 to Proposition 8. $\square$

PROPOSITION 9. *Suppose* $t_0, t_1, s \in \mathbb{T}$ *with* $t_0, t_1 \prec s$ *and* $t_0 \neq t_1$*. Then* $t_0 \otimes s \neq t_1 \otimes s$

PROOF. By Proposition 1, $s \in \textsc{succ}(t_i, \{s\})$ for each i. Suppose for a contradiction that $t_0 \otimes s = t_1 \otimes s$. Let $r \in \mathbb{T}^\circ$ be such that $r \prec t_0$. By Proposition 6, $r \prec t_0 \otimes s = t_1 \otimes s$. Hence

$$t_0 \otimes s = t_1 \otimes s \in \textsc{succ}(r, \{t_0 \otimes s, t_1 \otimes s\})$$

By Axiom 14 ($\otimes$-DECOMPOSITION), $t_0, t_1 \in \textsc{succ}(r, \{t_0, t_1\})$. But this contradicts Theorem 3. $\square$

PROPOSITION 10. *Suppose $t_0, t_1 \in \mathbb{T}$ with $s_0, s_1 \in \mathbb{T}^\circ$ such that $t_0 \sim_E s_0$ and $t_1 \sim_S s_1$. Then $t_0 \prec t_1$ if and only if $s_0 \prec s_1$. In other words, t_1 follows t_0 if and only if the start time of t_1 follows the end time of t_0.*

PROOF. Suppose $t_0 \sim_E s_0$ and $t_1 \sim_S s_1$. First suppose that $t_0 \prec t_1$. As $t_0 \sim_E s_0$, $s_0 \prec t_1$ by the definition of $\sim_E$. Similarly, $s_0 \prec s_1$ as $t_1 \sim_S s_1$. The proof for when $s_0 \prec s_1$ is analogous. $\square$

PROPOSITION 11. *Suppose $\textsc{succ}(s, \{t_0, t_1\}) = \{t_0\}$ with $t_i \in \mathbb{T}$. If $s \prec t_1$, then $t_0 \sqsubseteq t_1$.*

PROOF. Suppose $\textsc{succ}(s, \{t_0, t_1\}) = \{t_0\}$ with $t_i \in \mathbb{T}$, and also that $s \prec t_1$. Furthermore, suppose for a contradiction that $t_0 \not\sqsubseteq t_1$. As $\sqsubset$ is a linear order, $t_1 \sqsubset t_0$.

We first prove our claim assuming that $t_0 \not\sim_E t_1$. By Axiom 10 (PRIMITIVE REPRESENTATION) and Proposition 6, there are $p_0, p_1 \in \mathbb{T}^\circ$ with $p_i \sim_E t_i$. Then $p_0 \not\sim_E p_1$, and so from Axiom 13 ($\otimes$-ASSOCIATIVITY) and Axiom 11 (LINEARITY), either $p_0 \prec p_1$ or $p_1 \prec p_0$. As $t_1 \sqsubset t_0$, it is clear from Proposition 6 and the definition of $\sqsubseteq$ that $p_1 \prec p_0$. As $p_1 \sim_E t_1$, we have that $t_1 \prec p_0$. We now split into two cases.

First assume $t_0 \in \mathbb{T}^\circ$. In that case $t_0 = p_0$, and so $t_1 \prec t_0$. Thus we have $\textsc{succ}(s, \{t_0, t_1\}) = \{t_1\}$ by the arguments in the proof of Proposition 8.

Now assume $t_0 \notin \mathbb{T}^\circ$. By PRIMITIVE REPRESENTATION, we can write $t_0 = v_0 \otimes \cdots \otimes v_m \otimes p_0$. By Axiom 3 (CANDIDATE PRESENCE) and Axiom 5 (THINNING), we have that $\{t_0, t_1\} = \textsc{succ}(s, \{t_0, t_1\})$ and $\{t_1\} = \textsc{succ}(s, \{t_1\})$. But as $t_1 \prec p_0$, Axiom 12 (STRONG THICKENING) gives $\{t_1\} = \textsc{succ}(s, \{t_0, t_1\})$, a contradiction.

We now consider the case where $t_0 \sim_E t_1$. We decompose $t_0 = v_0 \otimes \cdots \otimes v_m$, $t_1 = u_0 \otimes \cdots \otimes u_n$. As $t_0 \sim_E t_1$, $v_m = u_n$ by Proposition 6. Again we have two possibilities.

The first possibility is that there is some $k > 0$ such that $v_{m-k} < u_{n-k}$ and $v_{m-i} < u_{n-i}$ for $i < k$. In that case $m, n > 0$, so we let $p_0 = v_0 \otimes \cdots \otimes v_{m-1}$, $p_1 = u_0 \otimes \cdots \otimes u_{n-1}$, and $q = v_m = u_n$. So $t_0 = p_0 \otimes q$ and $t_1 = p_1 \otimes q$. As $\textsc{succ}(s, \{t_0, t_1\}) = \{t_0\}$, we have $\{p_0\} = \textsc{succ}(s, \{p_0, p_1\})$ by Axiom 14 ($\otimes$-DECOMPOSITION). As $s \prec t_1$, Axiom 5 (THINNING) gives $\{t_1\} = \textsc{succ}(s, \{t_1\})$, and thus $s \prec p_1$ by $\otimes$-DECOMPOSITION. Therefore, $p_0 \sqsubseteq p_1$ by our induction hypothesis, and hence $t_0 \sqsubseteq t_1$.

The second possibility is that $m > n$ and $v_{m-i} = u_{n-i}$ for all $i \leq n$. This time we let $p = v_0 \otimes \cdots \otimes v_{m-n-1}$ and $q = v_{m-n} \otimes \cdots \otimes v_m$, and so $t_0 = p \otimes q$, $t_1 = q$. By LINEARITY, pick $r \prec s$. Then $s \in \textsc{succ}(r, \{s\})$ and so $\otimes$-DECOMPOSITION gives

$$\{s \otimes p \otimes q\} = \textsc{succ}(r, \{s \otimes p \otimes q, s \otimes q\})$$

Then again by $\otimes$-DECOMPOSITION

$$\{s \otimes p\} = \textsc{succ}(r, \{s \otimes p, s\})$$

As $s \prec p$, $s \otimes p \not\sqsubseteq p$ and $p \not\sim_E s$, which contradicts our proof of the case $t_0 \not\sim_E t_1$. $\square$

A.3 Proofs from Section 5.2

Given the equivalence relations $\sim_S$ and $\sim_E$ introduced in the previous section, the remainder of the proofs have a very similar style. We present them without further comment.

PROPOSITION 12. *Let $\mathbb{T}$ be any interval model satisfying the accepted axioms and let $t \in \mathbb{T} \setminus \mathbb{T}^\circ$. There are unique $t_0, t_1 \in \mathbb{T}^\circ$ such that $t_0 \sim_S t$, $t_1 \sim_E t$. Furthermore, $t = t_0 \otimes t_1$.*

PROOF. By Axiom 10 (PRIMITIVE REPRESENTATION), we have that $t = v_0 \otimes \cdots \otimes v_n$ with $v_i \in \mathbb{T}^\circ$. By Proposition 6, $v_0 \sim_S t$ and $v_n \sim_E t$. Also as $\mathbb{T}$ is an interval model, $t = v_0 \otimes v_n$. Let $t_0 = v_0$, $t_1 = v_n$. We need only show that they are unique.

Suppose $t = s_0 \otimes s_1$ with $s_i \in \mathbb{T}^\circ$. By Proposition 6, $s_0 \sim_S t$ and hence $s_0 \sim_S t_0$. Thus $s_0 = t_0$ by Proposition 7. A similar argument shows that $s_1 = t_1$. $\square$

PROPOSITION 13. *Let $\mathbb{T}$ be any interval model of the accepted axioms which is associative. Let $t \in \mathbb{T}$ and let $\mathcal{F} \subseteq \mathbb{T}$ be such that $s_1 \sim_E s_2$ and $t \prec s_1$ for all $s_1, s_2 \in \mathcal{F}$. Then $\textsc{succ}(t, \mathcal{F}) = \mathcal{F}$.*

PROOF. By Proposition 12, there is some $t_0 \in \mathbb{T}^\circ$ with $t_0 \sim_S t$. Similarly, for each $s \in \mathcal{F}$ there is some $s_1 \in \mathbb{T}^\circ$ such that $s_1 \sim_E s$. Furthermore, as $p_1 \sim_E p_2$ for each $p_i \in \mathcal{F}$, by Proposition 7, there is a unique s_1 that works for all elements of $\mathcal{F}$. Now take any $p \in \mathcal{F}$. As $t \prec p$, $t \otimes p$ is defined. By Proposition 6, $t_0 \sim_S t \sim_S t \otimes p$ and $s_1 \sim_E p \sim_E t \otimes p$. Thus $t \otimes p = t_0 \otimes s_1$ for all $p \in \mathcal{F}$, and hence $\{t_0 \otimes s_1\} = \{t\} \mathbin{;} \mathcal{F}$.

By Axiom 11 (LINEARITY), there is some $r \in \mathbb{T}^\circ$ such that $r \prec t_0$. Hence $r \prec t_0 \otimes s_1$ by Proposition 6. Thus $\textsc{succ}(r, \{t_0 \otimes s_1\}) = \{t_0 \otimes s_1\}$ by Proposition 1. As $\{t_0 \otimes s_1\} = \{t\} \mathbin{;} \mathcal{F}$, $\textsc{succ}(t, \mathcal{F}) = \mathcal{F}$ by Axiom 14 ($\otimes$-DECOMPOSITION). $\square$

THEOREM 4. *There is no interval model of the accepted axioms that is also associative.*

PROOF. By Axiom 11 (LINEARITY), let $t_i \in \mathbb{T}^\circ$ with $0 \leq i \leq 3$ and $t_0 \prec t_1 \prec t_2 \prec t_3$. By Proposition 13,

$$\textsc{succ}(t_0, \{t_1 \otimes t_3, t_2 \otimes t_3\}) = \{t_1 \otimes t_3, t_2 \otimes t_3\}$$

Then by Axiom 14 ($\otimes$-DECOMPOSITION), $\textsc{succ}(t_0, \{t_1, t_2\}) = \{t_1, t_2\}$. However, $\textsc{succ}(t_0, \{t_1\}) = \{t_1\}$ and $t_1 \prec t_2$, which violates Axiom 6 (THICKENING). $\square$

THEOREM 5. *Let $\mathbb{T}$ be an interval model satisfying all axioms but $\otimes$-DECOMPOSITION. Then there are expressions E_i and a stream S such that $[\![(E_0 \mathbin{;} E_1) \mathbin{;} E_2]\!]_S \not\supseteq [\![E_1 \mathbin{;} (E_1 \mathbin{;} E_2)]\!]_S$.*

PROOF. Suppose for a contradiction that $[\![(E_0 \mathbin{;} E_1) \mathbin{;} E_2]\!]_S \supseteq [\![E_0 \mathbin{;} (E_1 \mathbin{;} E_2)]\!]_S$ for any E_i and S. This means that we get the reverse direction of $\otimes$-DECOMPOSITION. In other words, for any t_0, t_1, t_2, and $\mathcal{F}_1, \mathcal{F}_2$,

$$\begin{aligned} t_1 \otimes t_2 \in \textsc{succ}(t_0, \mathcal{F}_1 \mathbin{;} \mathcal{F}_2),\ t_2 \in \textsc{succ}(t_1, \mathcal{F}_2) \\ \Rightarrow t_1 \in \textsc{succ}(t_0, \mathcal{F}_1) \end{aligned} \tag{11}$$

Now suppose we have $s \prec t_0 \prec p \prec q \prec t_1$, all elements of $\mathbb{T}^\circ$. By (7), we have that

$$t \otimes p \otimes t_1 = t_0 \otimes q \otimes t_1 = t_0 \otimes t_1 \tag{12}$$

As $t_0 \prec p, q$, $t_0 \prec p \otimes t_1, q \otimes t_1$ by Proposition 6. Consider now the set $\textsc{succ}(t_0, \{p \otimes t_1, q \otimes t_1\})$. We know by Axiom 3 (CANDIDATE PRESENCE) and Axiom 4 (RESPECTING ORDER) that $\textsc{succ}(t_0, \{p \otimes t_1\}) = \{p \otimes t_1\}$ and

$\text{SUCC}(t_0, \{ q \otimes t_1 \}) = \{ q \otimes t_1 \}$. Therefore, by Axiom 7 (NON-BLOCKING), $\text{SUCC}(t_0, \{ p \otimes t_1, q \otimes t_1 \})$ cannot be empty. By Axiom 3 (CANDIDATE PRESENCE), it must contain either $p \otimes t_1$ or $q \otimes t_1$. Therefore, by (12), we have that

$$\{ t_0 \} ; \{ p \otimes t_1, q \otimes t_1 \} = \{ t_0 \otimes t_1 \} \tag{13}$$

As $s \prec t_0$, we also have that $s \prec t_0 \otimes t_1$ by Proposition 6. Thus $\text{SUCC}(s, \{ t_0 \otimes t_1 \}) = \{ t_0 \otimes t_1 \})$ as before. So, in particular, the equivalence in (12) gives us

$$t \otimes p \otimes t_1, t_0 \otimes q \otimes t_1 \in \text{SUCC}(s, \{ t_0 \} ; \{ p \otimes t_1, q \otimes t_1 \})$$

By (11) and CANDIDATE PRESENCE, we have that $\text{SUCC}(t_0, \{ p \otimes t_1, q \otimes t_1 \}) = \{ p \otimes t_1, q \otimes t_1 \}$. Since $p, q \prec t_1$, arguing as before

$$\{ p \otimes t_1, q \otimes t_1 \} = \{ p, q \} ; \{ t_1 \}$$

Therefore, again by (11) and CANDIDATE PRESENCE, $\text{SUCC}(t_0, \{ p, q \}) = \{ p, q \}$. However, we know that $\text{SUCC}(t_0, \{ p \}) = \{ p \}$ by CANDIDATE PRESENCE and RESPECTING ORDER. As $p \prec q$, Axiom 6 (THICKENING) yields $\text{SUCC}(t_0, \{ p, q \}) = \{ p \}$, which is a contradiction. $\square$

PROPOSITION 14. *Let $\mathbb{T}$ be an interval model satisfying* (8) *and all axioms but $\otimes$-DECOMPOSITION. Let $\mathcal{F}$ be a candidate set in which $t = r_0 \otimes r_1 \otimes r_2$ for every $t \in \mathcal{F}$. Then*

$$\text{SUCC}(s, \mathcal{F}) = \{ t | s \prec t \in \mathcal{F} \textit{ and end time } t \sim_E q \in \mathbb{T}^\circ \textit{ is least} \}$$

PROOF. The key idea of this proof is to use the interior element r_1 of each time stamp, together with Axiom 12 (STRONG THICKENING), to show that no time stamp can block another with the same end time. Given Proposition 10 and several other axioms, we can assume without loss of generality that all events in $\mathcal{F}$ have the same end time and follow s. We let $q \in \mathbb{T}^\circ$ be the unique end time of all these elements. Then by (7), every element of $\mathcal{F}$ can be expressed as $p_i \otimes q$ with $p_i \in \mathbb{T}^\circ$.

Let $\mathcal{F}' = \{ p_i \mid p_i \otimes q \in \mathcal{F}, p_i \in \mathbb{T}^\circ \}$ be the set of start times of all these time stamps. As $\mathcal{F}'$ is linearly ordered, pick $p_0 \otimes q \in \mathcal{F}$ such that $p_0 \in \mathcal{F}'$ is least. By Axiom 6 (THICKENING), $p_0 \in \text{SUCC}(s, \mathcal{F}')$, and thus $p_0 \otimes q \in \text{SUCC}(s, \mathcal{F})$ by (8).

As every element in $\mathcal{F}$ has form $r_0 \otimes r_1 \otimes q$, there is some $r \in \mathbb{T}^\circ$ such that $p_i \prec r \prec q$ for all $p_i \in \mathcal{F}'$. Let r be the greatest such primitive time stamp. Take any $t = p_i \otimes q \in \mathcal{F}$, and define $\mathcal{F}_t = \{ p_j \otimes r \mid i \neq j \} \cup \{ p_i \}$. Note that $\mathcal{F}_t ; \{ q \} = \mathcal{F}$. By STRONG THICKENING, $p_i \in \text{SUCC}(s, \mathcal{F}_t)$, and so $t \in \text{SUCC}(s, \mathcal{F})$ by (8). $\square$

THEOREM 6. *Let $\mathbb{T}$ be an interval model satisfying* (8), (9) *and all axioms but $\otimes$-DECOMPOSITION. Let $\mathbb{S}$ be the Cayuga temporal model. If we identify $\mathbb{T}^\circ$ with $\mathbb{Z}$, the mapping $t_0 \otimes t_1 \mapsto [t_0, t_1] \in \mathbb{S}$, where $t_i \in \mathbb{T}^\circ$, is an isomorphism.*

PROOF. Proposition 12 guarantees that the mapping is a well-defined bijection. The proof of Proposition 8 does not require $\otimes$-DECOMPOSITION. Hence by Axiom 7 (NON-BLOCKING) and Axiom 12 (STRONG THICKENING), it is sufficient to show that $\text{SUCC}(t, \mathcal{F}) = \mathcal{F}$ for any $\mathcal{F}$ such that $p_1 \sim_E p_2$ and $t \prec p_1$ for all $p_i \in \mathcal{F}$.

Take any such t, $\mathcal{F}$. By Propositions 7 and 12, there is a unique s such that $p \sim_E s$ for all $p \in \mathcal{F}$. By Axiom 11 (LINEARITY), there is some $s \prec u \prec v$. By (9),

$$\text{SUCC}(t, \{ p_i \otimes (u \otimes v) \}_{p_i \in \mathcal{F}}) = \text{SUCC}(t, \mathcal{F}) ; \{ u \otimes v \}$$

Hence by Proposition 14,

$$\text{SUCC}(t, \{ p_i \otimes (u \otimes v) \}_{p_i \in \mathcal{F}}) = \{ p_i \otimes (u \otimes v) \}_{p_i \in \mathcal{F}}$$

By Proposition 12, each element of $\mathcal{F}$ has a unique start time. Thus $\text{SUCC}(t, \mathcal{F}) = \mathcal{F}$ $\square$