Two Perspectives on Change and Institutions

Tim Fernando
Trinity College Dublin, Ireland

Abstract

The contrast between internal and external perspectives on change, discussed within applied ontology in recent years by Galton, is linked to the difference between finite automata and their runs. The link is based on a finite approximability hypothesis, under which granularities are bounded by signatures in institutions, as defined by Goguen and Burstall. How abstract types, described internally, are realized externally as concrete particulars is complicated by differences in signatures and by competing processes with related signatures.

1 Introduction

One of many ways perspectives can differ, dubbed SNAP/SPAN in Grenon and Smith 2004, is that between synchronic *snap*shots of continuants (or endurants) at fixed times and diachronic accounts of occurrents (or perdurants) *span*ning temporal stretches. A twist on SNAP/SPAN proposed by Antony Galton pulls processes away from events towards objects

SNAP	SPAN
objects	events
	processes

EXP	HIST
objects	events
processes	

(Galton 2008, page 332) for a contrast described in Lyons' semantics textbook between, on the one hand, the "experential" (EXP)

> given by someone who is personally involved in what he is describing . . . dynamic, deictic, subjective

and, on the other hand, the "historical" (HIST)

> presented dispassionately with the minimum of subjective involvement . . . static, non-deictic, objective

(Lyons 1977, page 688). Galton elevates this contrast to

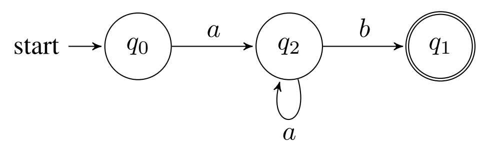
(†) a fundamental ontological distinction between EXP, the dynamic experiential world of objects and processes as they exist at one time, and HIST, the static historical overview populated by events that are generated by the ongoing processes in EXP

(Galton 2008, page 323). Two descriptions of time in McTaggart 1908 illustrate the distinction: a tensed A-series of moments from the future turning to the present and then the past (constituting EXP), and a tenseless B-series ordered by a binary relation $<$, pronounced "earlier than" (constituting HIST). An A-series judgment Past(φ) is interpreted in temporal logic relative to a time x (marking *now*, if you will) that changes in the righthand side of the clause

$$x \models \text{Past}(\varphi) \quad \Longleftrightarrow \quad y \models \varphi \text{ for some } y < x$$

to a time y earlier than x (Prior 1967). Various choices for x, $\models$, φ and $<$ are explored in the present paper, under a finite approximability hypothesis, motivated by computational and cognitive considerations. That hypothesis links EXP/HIST to the distinction between finite automata and their runs, up to bounded granularities, formulated as signatures in institutions (Goguen and Burstall 1992).

For orientation, consider the language a^+b accepted by the finite automaton $\mathcal{A}$ with three transitions



over the initial state q_0 and final (or accepting) state q_1. A run of $\mathcal{A}$ is a sequence of transitions that $\mathcal{A}$ makes, such as $q_0 \xrightarrow{a} q_2 \xrightarrow{b} q_1$, in the course of which, the finite automaton $\mathcal{A}$ changes its current state from q_0 to q_2 to q_1. How does this picture fit with Galton's proposal (†)? The automaton $\mathcal{A}$ coupled with its current state (initialized to q_0) is an EXP-process; a run is a HIST-event which describes change but does not itself change. Understood as accurate records of a past that is settled, HIST-events cannot change.[1] As continuants, EXP-processes may change and indeed, as described by HIST-events, do.

The relationship between EXP and HIST is complicated by processes that need not run to completion; sentences (1) to (4) are from Dowty 1979 (page 133).

(1) John was drawing a circle.

(2) John drew a circle.

(3) John was pushing a cart.

[1] In practice, of course, history *does* change, falling far short of an accurate record of the past. Revisions of HIST lie just outside the scope of the present paper.

(4) John pushed a cart.

(3) implies (4), but (1) may hold without (2) holding. How (1) can be true even if no circle was ever drawn by John is what Dowty calls the *Imperfective Paradox*, for which he appeals to *inertia worlds* that need not be actual. Parsons 1990 eschews non-actual worlds, attributing the difference between (2) and (4) to a notion of culmination over and above holding. While the debate between intensional and extensional accounts of the progressive continues (e.g., Klinedinst 2012), there is no shortage of linguistic constructions that have invited calls for non-actual worlds, such as counterfactuals (5), and non-veridical uses of *before* (6).

(5) John would have drawn a circle, had Al not stopped him.

(6) Al interfered before John drew a circle.

The moral for EXP/HIST, analyzed in terms of finite automata and their runs, is clear: any number of processes may run concurrently, not all of which may get completed.

The contrast between (3) implying (4) and (1) failing to imply (2) has often been adopted as the litmus test for processes versus events (e.g., Parsons 1990, page 183). (2) describes an event that culminates, whereas (4) describes a process that is dissective (Galton and Mizoguchi 2009, page 74), making the event-process contrast analogous to count-mass (e.g., Bach 1986). This count/mass analogy is orthogonal to the event/process distinction drawn in Galton 2012, under which some processes culminate (the *repeatables*) and other processes are dissective (the *continuables*). The thrust of Galton 2012 is instead to

> regard processes as abstract patterns of behaviour which may be realised in concrete form as actually occurring states or events

(page 35). It is this contrast that is targeted in the characterization below of EXP-processes as abstract types, and HIST-events as concrete particulars.

EXP-processes/automata	HIST-events/runs
abstract types	concrete particulars

The complication raised by (1)-(6) above is that while EXP-processes are typically conceived in isolation, they can be run, as HIST-events, alongside other EXP-processes that may interfere with them. What abstract types are manifested in concrete particulars can be tricky, making it tempting to focus on the concrete particulars and set aside abstract types to the extent that is possible.

Concentrating on concrete particulars, we will nevertheless avail ourselves of rudimentary forms of abstract types expressed by temporal propositions, called *fluents* for short. The particulars are analyzed at bounded granularities given by finite sets Σ of fluents. We keep the sets Σ finite in order to represent the runs by finite strings, enlarging Σ to lengthen the strings (refining the grain). The idea is familiar from the representations of a calendar year at various granularities. If the set $\Sigma = \{$Jan, Feb, …, Dec$\}$ of months suggests the string

$$s_\Sigma \;:=\; \fbox{Jan}\fbox{Feb}\cdots\fbox{Dec}$$

of length 12, enlarging Σ with days d1,d2,…,d31 refines s_Σ to the string

$$\fbox{Jan,d1}\fbox{Jan,d2}\cdots\fbox{Jan,d31}\fbox{Feb,d1}\cdots\fbox{Dec,d31}$$

of length 366 for a leap year. We draw boxes (instead of the usual curly braces $\{$ and $\}$) around sets *qua* symbols for a film strip. While it is irresistible to call the boxes SNAPshots, a change in Σ can cause a box to split, as $\fbox{Jan}$ in s_Σ does (30 times) with the addition of d1,d2…,d31

$$\fbox{Jan} \;\leadsto\; \fbox{Jan,d1}\fbox{Jan,d2}\cdots\fbox{Jan,d31}$$

SPANning 31 boxes. Similarly, a common Reichenbachian account of the progressive puts a *reference time* R inside the event time E,[2] splitting $\fbox{E}$ into 3 boxes

$$\fbox{E} \;\leadsto\; \fbox{E}\fbox{E,R}\fbox{E}$$

(one before, one simultaneous, and one after R). We can also encode runs of an automaton as strings; for example, $q_0 \xrightarrow{a} q_2 \xrightarrow{b} q_1$ above as $\fbox{$a, q_2$}\fbox{$b, q_1$}$, leaving out the automaton's initial state q_0.

Encoding runs as strings is useful for expressing the languages accepted by finite automata in Monadic Second-Order logic (MSO), one half of Büchi's theorem (e.g., Libkin 2010, Theorem 7.21, pp 124-126). Strings are construed as models of predicate logic, associating a finite set Σ with a signature Σ_S specifying a unary relation symbol P_a, for each $a \in \Sigma$, alongside a binary relation symbol S. The intent is that S express the successor relation between string positions, and P_a pick out the positions where a occurs. For instance,

$$\exists x \exists y (S(x, y) \wedge P_a(x))$$

says a occurs before the end of the string. Confining our attention to finite models of size $n \geq 0$,[3] we write $[n]$ for the set of integers from 1 to n

$$[n] \;:=\; \{1, 2, \ldots, n\}$$

(with $[0] = \emptyset$) and S_n for the successor relation on $[n]$

$$S_n \;:=\; \{(i, i+1) \mid i \in [n] \text{ and } i < n\}$$

(with $S_1 = S_0 = \emptyset$). A Σ_S-*model* $M = \langle [n], S_n \{P_a^M\}_{a \in \Sigma} \rangle$ interprets S as S_n and each P_a as a subset P_a^M of its domain $[n]$ (for some $n \geq 0$). In the terminology of Kripke semantics, $\langle [n], S_n \rangle$ is a frame, while $\{P_a^M\}_{a \in \Sigma}$ defines the Σ-valuation $v \subseteq [n] \times \Sigma$ such that

$$v(i, a) \iff i \in P_a^M$$

for all $i \in [n]$ and $a \in \Sigma$. We can also view M as the finite automaton $\mathcal{A}_M$ with initial state 1, final state n, and set of transitions

$$\{(i, S, j) \mid (i, j) \in S_n\} \;\cup\; \{(i, a, i) \mid a \in \Sigma \text{ and } i \in P_a^M\}$$

[2] See Moens and Steedman (1988), pp. 22 and 28 (footnote 3).

[3] We follow Libkin 2010 in allowing a model to have the empty set as its domain (universe).

over the alphabet $\Sigma \cup \{S\}$. For each $i \in [n]$, let us collect all $a \in \Sigma$ such that $i \in P_a^M$ in

$$\alpha_i \ := \ \{a \in \Sigma \mid i \in P_a^M\}.$$

Then $\mathcal{A}_M$ accepts the language

$$\alpha_1^* S \alpha_2^* S \cdots S \alpha_n^*$$

of strings $s_1 S s_2 S \cdots S s_n \in (\Sigma \cup \{S\})^*$ where $s_i \in \alpha_i^*$ for $i \in [n]$. What's more, there is a bijection between Σ_S-models M and strings $\alpha_1 \cdots \alpha_n$ over the powerset 2^Σ of Σ, as the equivalence

$$i \in P_a^M \iff a \in \alpha_i$$

goes from M to $\alpha_1 \cdots \alpha_n$ and back. Thus, a string $\alpha_1 \cdots \alpha_n \in (2^\Sigma)^*$ can serve as both a Σ_S-model against which to interpret predicate logic formulas such as $\exists x \exists y (S(x,y) \wedge P_a(x))$ and a Kripke model against which to interpret, in combination with a position $i \in [n]$, modal logic propositions such as $\Diamond a$ (where $a \in \Sigma$). The following table describes the idea.[4]

	predicate logic	modal logic
McTaggart	B-series	A-series
Lyons/Galton	HIST	EXP
perspective	external	internal
model	$\alpha_1 \cdots \alpha_n$	$\alpha_1 \cdots \alpha_n, i$

Table 1

Table 1 makes no mention of Σ (on the understanding Σ is fixed in the background) and derives EXP-automata $\mathcal{A}_M$ from HIST-strings M. But variations in Σ are useful to cope with the open-endedness of ontologies, and many finite automata do not have the form $\mathcal{A}_M$. In order to analyze variations in Σ and conceptions of EXP that are not reducible to the purely temporal realm HIST (automata being arguably prior to their runs), we adopt the framework of institutions from Goguen and Burstall 1992. We form institutions around strings $\alpha_1 \cdots \alpha_n$ based on MSO in the next section, section 2, and institutions around languages (as opposed to strings) for automata in section 3. Relations between the various institutions are explored in section 4.

2 HIST-events as strings

The *formulas* φ *of Monadic Second-Order Logic* (MSO) are generated through seven clauses

$$\varphi \ ::= \ S(x,y) \mid P_a(x) \mid X(x) \mid \varphi \wedge \varphi' \mid \neg\varphi \mid \exists x \varphi \mid \exists X \varphi$$

from three disjoint infinite sets Var_1, Var_2 and Θ of first-order variables $x, y \in Var_1$, second-order variables $X \in Var_2$, and fluents $a \in \Theta$, respectively. The *vocabulary* $voc(\varphi)$ of φ is the finite subset of Θ occurring in φ

$$voc(S(x,y)) = voc(X(x)) = \emptyset$$
$$voc(P_a(x)) = \{a\}$$
$$voc(\varphi \wedge \varphi') = voc(\varphi) \cup voc(\varphi')$$
$$voc(\neg\varphi) = voc(\exists x \varphi) = voc(\exists X \varphi) = voc(\varphi).$$

[4] The external/internal divide in Priorean tense logic is discussed at length in Blackburn 2006.

An MSO-*sentence* is understood to be an MSO-formula in which all variable occurrences are bound. We let *Fin*(Θ) be the set of finite subsets of Θ, and for every $\Sigma \in Fin(\Theta)$, put every MSO sentence with vocabulary contained in Σ into the set $MSO(\Sigma)$

$$MSO(\Sigma) := \{\varphi \mid \varphi \text{ is an MSO-sentence and } voc(\varphi) \subseteq \Sigma\}.$$

The notion of a Σ_S-model M *satisfying* a sentence $\varphi \in MSO(\Sigma)$, written $M \models_\Sigma \varphi$, is defined in the usual Tarskian manner. The Σ_S-model $\langle [n], S_n, \{P_a^M\}_{a \in \Sigma} \rangle$ is identified with the string $\alpha_1 \cdots \alpha_n$ where α_i is $\{a \in \Sigma \mid i \in P_a^M\}$ for each $i \in [n]$. Thus, for each $\Sigma \in Fin(\Theta)$, Σ-satisfaction is a binary relation

$$\models_\Sigma \ \subseteq \ (2^\Sigma)^* \times MSO(\Sigma)$$

between $(2^\Sigma)^*$ and $MSO(\Sigma)$. To describe how $\models_\Sigma$ varies as Σ ranges over finite subsets of Θ, we define the function $\rho_\Sigma : (2^\Theta)^* \to (2^\Sigma)^*$ that intersects a string in $(2^\Theta)^*$ componentwise with Σ for its Σ-*reduct*

$$\rho_\Sigma(\alpha_1 \cdots \alpha_n) \ := \ (\alpha_1 \cap \Sigma) \cdots (\alpha_n \cap \Sigma)$$

in $(2^\Sigma)^*$. For example, $\rho_{\{\mathrm{E}\}}(\boxed{\mathrm{E}}\boxed{\mathrm{E,R}}\boxed{\mathrm{E}}) = \boxed{\mathrm{E}}\boxed{\mathrm{E}}\boxed{\mathrm{E}}$.

Proposition 1 *For all* $\Sigma \in Fin(\Theta)$, $\varphi \in MSO(\Sigma)$ *and* $s \in (2^\Sigma)^*$,

$$s \models_\Sigma \varphi \iff \rho_{voc(\varphi)}(s) \models_{voc(\varphi)} \varphi .$$

With Proposition 1, the relations $\{\models_\Sigma\}_{\Sigma \in Fin(\Theta)}$ become an *institution* (Goguen and Burstall 1992) provided we

(i) construe *Fin*(Θ) as a category of signatures with morphisms (Σ, Σ') whenever $\Sigma \subseteq \Sigma' \in Fin(\Theta)$

(ii) extend the map $\Sigma \mapsto MSO(\Sigma)$ to pairs (Σ, Σ') such that $\Sigma \subseteq \Sigma' \in Fin(\Theta)$, setting $MSO(\Sigma, \Sigma')$ to the inclusion $MSO(\Sigma) \hookrightarrow MSO(\Sigma')$ mapping $\varphi \in MSO(\Sigma) \subseteq MSO(\Sigma')$ to itself

(iii) define a contravariant functor *Mod* from *Fin*(Θ) so that whenever $\Sigma \subseteq \Sigma' \in Fin(\Theta)$, $Mod(\Sigma)$ is $(2^\Sigma)^*$, while $Mod(\Sigma', \Sigma) : MSO(\Sigma') \to MSO(\Sigma)$ is the restriction of ρ_Σ to $Mod(\Sigma')$

$$Mod(\Sigma', \Sigma)(s) \ = \ \rho_\Sigma(s) \qquad \text{for all } s \in (2^{\Sigma'})^*.$$

One may expect that for every sentence $\varphi \in MSO(\Sigma)$, the set $\{s \in (2^\Sigma)^* \mid s \models_\Sigma \varphi\}$ is a regular language, in view of Büchi's theorem, BT, mentioned in the introduction. The problem, however, is that BT interprets $\varphi \in MSO(\Sigma)$ relative to strings over Σ, not 2^Σ as above. To establish the regularity of $\{s \in (2^\Sigma)^* \mid s \models_\Sigma \varphi\}$ via BT, we can translate $\varphi \in MSO(\Sigma)$ to $\varphi^\sharp \in MSO(2^\Sigma)$ homomorphically, treating P_a as the union of $P_{\Sigma'}$'s for $a \in \Sigma' \subseteq \Sigma$

$$P_a(x)^\sharp \ := \ \bigvee \{P_{\Sigma'}(x) \mid \Sigma' \subseteq \Sigma \text{ and } a \in \Sigma'\} \quad \text{for } a \in \Sigma.$$

To invert the translation, we map $\psi \in MSO(2^\Sigma)$ to $\psi_\sharp \in MSO(\Sigma)$ with

$$P_{\Sigma'}(x)_\sharp \ := \ \bigwedge \{P_a(x) \mid a \in \Sigma'\} \wedge \bigwedge \{\neg P_a(x) \mid a \in \Sigma - \Sigma'\} \qquad \text{for } \Sigma' \subseteq \Sigma.$$

An advantage in interpreting $\mathit{MSO}(\Sigma)$ relative to strings over 2^{Σ} is Proposition 1 above, the *satisfaction condition* for an institution (Goguen and Burstall 1992).

A string s is understood to come with a fixed granularity given by a signature Σ such that $s \in (2^{\Sigma})^*$. Σ-reducts preserve string length. But as hinted by the discussion in the introduction of

$$s_{\Sigma} \ := \ \boxed{\text{Jan}}\boxed{\text{Feb}} \cdots \boxed{\text{Dec}}$$

and

$$\boxed{\text{Jan,d1}}\boxed{\text{Jan,d2}} \cdots \boxed{\text{Jan,d31}}\boxed{\text{Feb,d1}} \cdots \boxed{\text{Dec,d31}}$$

we should expect the length of a string to grow as we enlarge Σ. To accommodate this growth, we implement "McTaggart's dictum that 'there could be no time if nothing changed'" (Prior 1967, page 85) by working with strings $\alpha_1\alpha_2 \cdots \alpha_n$ that are *stutterless* in that $\alpha_i \neq \alpha_{i+1}$ for i from 1 to $n-1$. Given a string s, let $bc(s)$ be the stutterless string obtained from s by compressing blocks α^n of $n > 1$ consecutive occurrences in s of the same symbol α to a single α, leaving s otherwise unchanged

$$bc(s) \ := \ \begin{cases} bc(\alpha s') & \text{if } s = \alpha\alpha s' \\ \alpha \ bc(\beta s') & \text{if } s = \alpha\beta s' \text{ with } \alpha \neq \beta \\ s & \text{otherwise.} \end{cases}$$

The restriction of bc to any finite alphabet is computable by a finite-state transducer, as are, for all $\Sigma' \in \mathit{Fin}(\Theta)$ and $\Sigma \subseteq \Sigma'$, the composition $\rho_{\Sigma}; bc$ for bc_{Σ}

$$bc_{\Sigma}(s) \ := \ bc(\rho_{\Sigma}(s)) \qquad \text{for } s \in (2^{\Sigma'})^*.$$

For example, if Σ is {Jan,Feb,. . .,Dec}, bc_{Σ} maps

$$\boxed{\text{Jan,d1}}\boxed{\text{Jan,d2}} \cdots \boxed{\text{Jan,d31}}\boxed{\text{Feb,d1}} \cdots \boxed{\text{Dec,d31}}$$

to s_{Σ}. Without the compression bc in bc_{Σ}, we are left with the map ρ_{Σ} that leaves the ontology intact (insofar as the domain of an MSO-model is given by the string length), whilst restricting the vocabulary (for Σ-reducts). The institution described by Proposition 1 can be adjusted to another institution in which

- the models are stutterless strings[5]
- the reducts ρ_{Σ} are replaced by bc_{Σ}, and
- the satisfaction relations $\models'_{\Sigma}$ are given by explicitly referring to the sentence's vocabulary

$$s \models'_{\Sigma} \varphi \iff bc_{voc(\varphi)}(s) \models_{voc(\varphi)} \varphi.$$

Compressing strings via bc_{Σ} can be inverted to lengthen strings. The *inverse limit* $\mathfrak{IL}(\Theta, bc)$ *of* Θ, bc consists of functions $\mathbf{a} : \mathit{Fin}(\Theta) \to \mathit{Fin}(\Theta)^*$ that respect the projections bc_{Σ}

$$\mathbf{a}(\Sigma) = bc_{\Sigma}(\mathbf{a}(\Sigma')) \text{ whenever } \Sigma \subseteq \Sigma' \in \mathit{Fin}(\Theta).$$

The prefix relation on strings

$$s \ \mathit{prefix} \ s' \iff s' = s\hat{s} \text{ for some } \hat{s}$$

lifts to maps **a** and **a′** in $\mathfrak{IL}(\Theta, bc)$ by universal quantification for an irreflexive relation

$$\mathbf{a} \prec \mathbf{a}' \iff \mathbf{a} \neq \mathbf{a}' \text{ and } (\forall \Sigma \in \mathit{Fin}(\Theta)) \ \mathbf{a}(\Sigma) \ \mathit{prefix} \ \mathbf{a}'(\Sigma)$$

that is *tree-like* on $\mathfrak{IL}(\Theta, bc)$ — i.e., transitive and left linear: for every $\mathbf{a} \in \mathfrak{IL}(\Theta, bc)$, and all $\mathbf{a}_1 \prec \mathbf{a}$ and $\mathbf{a}_2 \prec \mathbf{a}$,

$$\mathbf{a}_1 \prec \mathbf{a}_2 \ \text{ or } \ \mathbf{a}_2 \prec \mathbf{a}_1 \ \text{ or } \ \mathbf{a}_2 = \mathbf{a}_1.$$

In other words, time branches at the inverse limit $\mathfrak{IL}(\Theta, bc)$.

Working with the functions ρ_{Σ}, we can decompose a string $s \in (2^{\Sigma \cup \Sigma'})^*$ as

$$s \ = \ \rho_{\Sigma}(s) \ \& \ \rho_{\Sigma'}(s)$$

where *superposition* & forms the componentwise union of strings of sets with the same length

$$(\alpha_1 \cdots \alpha_n) \ \& \ (\beta_1 \cdots \beta_n) \ := \ (\alpha_1 \cup \beta_1) \ \cdots \ (\alpha_n \cup \beta_n)$$

inducing a relation of *subsumption* $\trianglerighteq$

$$s \trianglerighteq s' \iff s \& s' = s$$

(Fernando 2004). Subsumption is componentwise containment $\supseteq$ between equally long strings of sets α_i and β_i

$$\alpha_1 \cdots \alpha_n \trianglerighteq \beta_1 \cdots \beta_m \iff n = m \text{ and } \beta_i \subseteq \alpha_i \text{ for } i \in [n]$$

and extends naturally to a relation with languages L, understood as disjunctions and interpreted by existential quantification

$$s \trianglerighteq L \iff (\exists s' \in L) \ s \trianglerighteq s'.$$

For example, a string s has length ≥ 2 iff $s \trianglerighteq \boxed{}\boxed{}^{+}$. We can reconstruct the Vendler classes described in Dowty 1979 and variants thereof by representing an event e at granularity Σ as a string $str_{\Sigma}(e) \in (2^{\Sigma})^+$, relative to which we define e to be

- Σ-*dynamic* if $bc(str_{\Sigma}(e)) \trianglerighteq \boxed{}\boxed{}^{+}$
- Σ-*durative* if $bc(str_{\Sigma}(e)) \trianglerighteq \boxed{}\boxed{}\boxed{}^{+}$
- Σ-*telic* if $str_{\Sigma}(e) \trianglerighteq \boxed{\neg\varphi}^{+}\boxed{\varphi}$ for some fluent $\varphi \in \Sigma$ (marking the culmination of e).

The choice of vocabulary Σ determines what a string can represent; it is linked in Fernando 2015 to, among other things, the *event nucleus* in Moens and Steedman 1988 (associating a preparatory phase, a culmination and a consequent state with an event, *not* all of which may be represented in a string). Up to granularity Σ, the string $str_{\Sigma}(e)$ gives a completely deterministic picture of the event e. For non-determinism at Σ, we must look to a language (over the alphabet 2^{Σ}) with more than the one string $str_{\Sigma}(e)$.

[5] Apart from applying bc, a string can be made stutterless also by adding a fluent, turning, for instance, $\boxed{a}\boxed{a}\boxed{a}$ into $\boxed{a\text{,tic}}\boxed{a}\boxed{a\text{,tic}}$. The crucial point is that stutterlessness ensures the vocabulary is large enough to express the distinctions of interest (insofar as they lengthen a string).

3 EXP-processes as sets of strings

Moving from strings $\alpha_1\alpha_2\cdots\alpha_n$ to finite automata, let us recall from the introduction the deterministic automaton $\mathcal{A}_M$ that accepts the language $\alpha_1^* S\alpha_2^* S\cdots S\alpha_n^*$ over the alphabet $\Sigma\cup\{S\}$ and note that a transition in $\mathcal{A}_M$ labeled by $a\in\Sigma$ does not move time forward (unlike one labeled by S). Whereas the previous section decomposes $\alpha_1\cdots\alpha_n$ through subsets of Σ, the present section decomposes a state also through the transitions from it, treating some of the transitions as fields that make up a record, and other transitions as specifications of types (including singletons for tokens). Take the famous example (7) of event modification/predication in Davidson 1967 (page 81).

(7) Jones did it slowly, deliberately, with a knife

We associate (7) with the record

(8)
$$\begin{bmatrix} \textit{who} = \begin{bmatrix}\textit{jones}\end{bmatrix} \\ \textit{how} = \begin{bmatrix} \textit{slow} \\ \textit{deliberate} \\ \textit{with} = \begin{bmatrix}\textit{knife}\end{bmatrix} \end{bmatrix} \end{bmatrix}$$

which we analyze presently as a minimal deterministic automaton accepting the four strings *who jones, how slow, how deliberate* and *how with knife*.

Automata can be formed from languages using a notion of derivative connected with the Myhill-Nerode theorem (e.g., Hopcroft and Ullman 1979). Given a language L and a string s, the *s-derivative of L* is the set

$$L_s \ := \ \{s' \mid ss' \in L\}$$

of strings that put after s belong to L (Brzozowski 1964). The Myhill-Nerode equivalence $\sim_L$ between strings that have the same continuations in L is equality of derivatives

$$s \sim_L s' \iff L_s = L_{s'}.$$

The chain of equivalences

$$a_1a_2\cdots a_n \in L \iff a_2\cdots a_n \in L_{a_1} \iff \cdots \iff \epsilon \in L_{a_1\cdots a_n}$$

(from $a_1\cdots a_n$ to the empty/null string ϵ) means that L is accepted by the deterministic automaton with

- s-derivatives L_s as states
- initial state $L = L_\epsilon$
- a-transitions from L_s to L_{sa} (for every symbol a)
- final states L_s such that $\epsilon \in L_s$.

The Myhill-Nerode theorem says that a language L over a finite alphabet A is regular iff the set $\{L_s \mid s\in A^*\}$ of derivatives of L is finite. Note that L_s is non-empty precisely if s is the prefix of some string in L. Moreover, if L_s is empty then so is L_{sa} for every symbol a. That is, $\emptyset$ is a sink state that we may safely exclude from the states of the automaton above, at the cost of making the transition function partial. Let us define an *A-state* to be a non-empty subset q of A^* that is prefix-closed (i.e., for all $sa\in q$, $s\in q$). An A-state q can then make an a-transition to its a-derivative q_a precisely if $a\in q$.

Now, let us fix an infinite set *Act* of symbols that will play a role analogous to Θ in section 2 . For $A\in \textit{Fin}(\textit{Act})$, let $\textit{sen}(A)$ be the set of formulas generated from $a\in A$ according to

$$\varphi \ ::= \ \top \mid \neg\varphi \mid \varphi\wedge\varphi' \mid \langle a\rangle\varphi$$

(Hennessy and Milner 1985). We interpret these formulas over A-states q, treating $\top$ as a tautology, $\neg$ as negation, $\wedge$ as conjunction, and $\langle a\rangle$ as a diamond modal operator for a-transitions

$$q \models \langle a\rangle\varphi \iff a\in q \text{ and } q_a \models \varphi.$$

We extend $\langle a\rangle\varphi$ from $a\in A$ to strings $s\in A^*$, defining $\langle\epsilon\rangle\varphi := \varphi$ and $\langle as\rangle\varphi := \langle a\rangle\langle s\rangle\varphi$ so that

$$\langle a_1\cdots a_n\rangle\varphi \ = \ \langle a_1\rangle\cdots\langle a_n\rangle\varphi.$$

and

$$q \models \langle s\rangle\varphi \iff s\in q \text{ and } q_s \models \varphi.$$

Next, given a string $s\in \textit{Act}^*$ and a set $A\in\textit{Fin}(\textit{Act})$, we compute the longest prefix of s that belongs to A^* by the function $\pi_A : \textit{Act}^* \to A^*$ defined by $\pi_A(\epsilon) := \epsilon$ and

$$\pi_A(as) \ := \ \begin{cases} a\pi_A(s) & \text{if } a\in A \\ \epsilon & \text{otherwise.}\end{cases}$$

The *A-restriction of* a language $q\subseteq \textit{Act}^*$ is the image of q under π_A

$$q\restriction A \ := \ \{\pi_A(s) \mid s\in q\}\ .$$

If q is an *Act*-state, then its A-restriction, $q\restriction A$, is an A-state and is just the intersection $q\cap A^*$ with A^*. A-restrictions are interesting because satisfaction $\models$ of formulas in $\textit{sen}(A)$ can be reduced to them.

Proposition 2 *For every $A\in\textit{Fin}(\textit{Act})$, $\varphi\in\textit{sen}(A)$ and Act-state q,*

$$q\models\varphi \iff q\restriction A\models\varphi$$

and if, moreover, $s\in q\restriction A$, then

$$q\models\langle s\rangle\varphi \iff (q\restriction A)_s\models\varphi.$$

Proposition 2 is proved by a routine induction on $\varphi\in\textit{sen}(A)$.

There is structure lurking around Proposition 2 that is most conveniently described in category-theoretic terms. For $A\in\textit{Fin}(\textit{Act})$, let $Q(A)$ be the category with

- A-states q as objects
- pairs (q,s) such that $s\in q$ as morphisms from q to q_s, with identities (q,ϵ) and composition as concatenation

$$(q,s)\ ;\ (q_s,s') \ := \ (q,ss').$$

To turn Q into a functor from $\mathit{Fin}(\mathit{Act})^{op}$ (with morphisms (A', A) such that $A \subseteq A' \in \mathit{Fin}(\mathit{Act})$) to the category **Cat** of small categories, we map a $\mathit{Fin}(\mathit{Act})^{op}$-morphism (A', A) to the functor $Q(A', A) : Q(A') \to Q(A)$ sending an A'-state q' to the A-state $q' \restriction A$, and the $Q(A')$-morphism (q', s') to the $Q(A)$-morphism $(q' \restriction A, \pi_A(s'))$. The *Grothendieck construction for* Q is the category $\int Q$ where

- objects are pairs (A, q) such that $A \in \mathit{Fin}(\mathit{Act})$ and q is an A-state
- morphisms from (A', q') to (A, q) are pairs

$$((A', A),\ (q' \restriction A, s))$$

of $\mathit{Fin}(\mathit{Act})^{op}$-morphisms (A', A) and $Q(A)$-morphisms $(q' \restriction A, s)$ such that $(q' \restriction A)_s = q$.

(e.g., Tarlecki, Burstall and Goguen 1991). $\int Q$ integrates the different categories $Q(A)$ (for $A \in \mathit{Fin}(\mathit{Act})$), lifting a $Q(A)$-morphism (q, s) to a ($\int Q$)-morphism from (A', q') to (A, q_s) whenever $A \subseteq A'$ and $q' \restriction A = q$.

Given a small category C, let us write $|C|$ for the set of objects of C. Thus, for $A \in \mathit{Fin}(\mathit{Act})$, $|Q(A)|$ is the set

$$|Q(A)| \ = \ \{q \subseteq A^* \mid q \neq \emptyset \text{ and } q \text{ is prefix-closed}\}$$

of A-states. Next, for $(A, q) \in |\int Q|$, let $\mathit{Mod}(A, q)$ be the full subcategory of $Q(A)$ with objects required to have q as a subset

$$|\mathit{Mod}(A, q)| \ := \ \{q' \in |Q(A)| \mid q \subseteq q'\}.$$

That is, $|\mathit{Mod}(A, q)|$ is the set of A-states q' such that for all $s \in q$, $q' \models \langle s \rangle \top$. The intuition is that q is a form of record typing over A that allows us to simplify clauses such as

$$q' \models \langle s \rangle \varphi \iff s \in q' \text{ and } {q'}_s \models \varphi \qquad (\dagger)$$

when $s \in q \subseteq q'$ (making s a path through a record; Cooper 2012, page 284). The second conjunct in the righthand side of (†), $q'_s \models \varphi$, presupposes the first conjunct, $s \in q'$. We can lift that presupposition out of (†) by asserting that whenever $s \in q$ and $q \subseteq q'$,

$$q' \models \langle s \rangle \varphi \iff q'_s \models \varphi.$$

This comes close to the equivalence in Proposition 2, except that A-restrictions are missing. These reducts appear once we vary A, and step from $Q(A)$ to $\int Q$. Taking this step, we turn the categories $\mathit{Mod}(A, q)$ to a functor Mod from $\int Q$ to **Cat**, mapping a $\int Q$-morphism $\sigma = ((A', A), (q' \restriction A, s))$ from (A', q') to (A, q) to the functor

$$\mathit{Mod}(\sigma) : \mathit{Mod}(A', q') \to \mathit{Mod}(A, q)$$

sending $q'' \in |\mathit{Mod}(A', q')|$ to the s-derivative of its A-restriction, $(q'' \restriction A)_s$, and a $\mathit{Mod}(A', q')$-morphism (q'', s') to the $\mathit{Mod}(A, q)$-morphism $(q'' \restriction A, \pi_A(s'))$.

The syntactic counterpart of $Q(A)$ is $\mathit{sen}(A)$, which we turn into a functor sen matching Mod. A basic insight from Goguen and Burstall 1992 informing the present approach is the importance of a category **Sign** of *signatures* which the functor *sen* maps to the category **Set** of sets (and functions) and which *Mod* maps contravariantly to **Cat**. The definition of *Mod* above suggests that $\mathbf{Sign}^{op}$ is $\int Q$.[6] A $\int Q$-morphism from $\int Q$-objects (A', q') to (A, q) is determined uniquely by a string $s \in q' \restriction A$ such that

$$q = (q' \restriction A)_s \text{ and } A \subseteq A'. \qquad (\ddagger)$$

Let $(A, q) \xrightarrow{s} (A', q')$ abbreviate the conjunction (‡), which holds precisely if $((A', A), (q' \restriction A, s))$ is a $\int Q$-morphism from (A', q') to (A, q). Now for $(A, q) \in |\int Q|$, let $\mathit{sen}(A, q)$ be $\mathit{sen}(A)$ (ignoring q), and when $(A, q) \xrightarrow{s} (A', q')$, let

$$\mathit{sen}(\sigma) : \mathit{sen}(A) \to \mathit{sen}(A')$$

send $\varphi \in \mathit{sen}(A)$ to $\langle s \rangle \varphi \in \mathit{sen}(A')$. To see that an *institution* arises from restricting $\models$ to $|\mathit{Mod}(A, q)| \times \mathit{sen}(A)$, for $(A, q) \in |\int Q|$, it remains to check the *satisfaction condition*:

whenever $(A, q) \xrightarrow{s} (A', q')$ and $q'' \in \mathit{Mod}(A', q')$ and $\varphi \in \mathit{sen}(A)$,

$$q'' \models \langle s \rangle \varphi \iff (q'' \restriction A)_s \models \varphi.$$

This follows from Proposition 2, as s must be in $q' \restriction A$ and thus also in $q'' \restriction A$.

Returning to sentence (7), let *Act* contain the finite set

$$A \ := \ \{\textit{who, jones, how, slow, deliberate, with, knife}\}$$

so that if q is the A-state $\{\textit{who, how, how with,}, \epsilon\}$ then

$$q \ \cup \ \{\textit{who jones, how slow, how deliberate, how with knife}\}$$

is an (A, q)-model corresponding to the record (8) that (having a non-empty s-derivative for every $s \in q$) has record type q. Records and types are employed extensively in Cooper 2012 as linguistic resources that, as argued in Cooper and Ranta 2008, characterize natural language. The relevance of this to EXP/HIST is expressed concisely by

$$\frac{\text{EXP}}{\text{HIST}} \approx \frac{\text{automata}}{\text{runs}} \approx \frac{\text{resources}}{\text{uses}} \approx \frac{\text{types}}{\text{tokens}}$$

where, in the simplest case, a token is a string, while a type is a language (to which the string may or may not belong).

4 EXP/HIST and connections

Having likened EXP/HIST to modal/predicate logic in Table 1 (from the introduction), we brought out notions of granularity in sections 2 and 3 through signatures Σ, organized into a category **Sign**, from which institutionally,

(i) a functor sen : **Sign** $\to$ **Set** assigns a set $\mathit{sen}(\Sigma)$ of Σ-sentences

(ii) a contravariant functor Mod : $\mathbf{Sign}^{op} \to$ **Cat** assigns a set $|\mathit{Mod}(\Sigma)|$ of Σ-models, and

[6] That said, we might refine **Sign**, requiring of a signature (A, q) that q be a regular language. For this, it suffices to replace $\int Q$ by $\int R$ where $R : \mathit{Fin}(\mathit{Act})^{op} \to$ **Cat** is the subfunctor of Q such that $R(A)$ is the full subcategory of $Q(A)$ with objects regular languages. We can make this refinement with*out* requiring that A-states in $\mathit{Mod}(A, q)$ be regular, forming $\mathit{Mod}(A, q)$ from Q (not R).

(iii) satisfaction relations $\models_\Sigma$ relate Σ-models and Σ-sentences smoothly across different signatures (made precise by the satisfaction condition).

Institutions were formed from a large set Θ of fluents in section 2 and a large set *Act* of symbols in section 3, as outlined in the table below.

	HIST	EXP
signature	$\Sigma \in \mathit{Fin}(\Theta)$	(A, q) where $q \subseteq A^*$
model	$(2^\Sigma)^*$-string	A-language $\supseteq q$
sentence	MSO_Σ	Hennessy-Milner over A

Table 2

In HIST, a Σ-model is a (2^Σ)-string (where an X-string is a string over the alphabet X), while in EXP, an (A, q)-model is an A-language that contains q (where an X-language is a set of X-strings). A comparison of Tables 1 and 2 raises the question: what has become of the clear difference between a HIST-model $\alpha_1 \cdots \alpha_n$ and an EXP-model $\alpha_1 \cdots \alpha_n, i$ in Table 1, consisting of a string position $i \in [n]$?

A string $\alpha_1 \cdots \alpha_n$ can be paired with a string position $i \in [n]$ within the institution built around MSO in section 2 with sentences expressive enough to capture Priorean tense logic (using second-order quantification for the transitive closure $<$ of S). Indeed, for any 2^Σ-string $\alpha_1 \cdots \alpha_n$ and set $I \subseteq [n]$ of string positions (including singletons $\{i\}$ for $i \in [n]$), we can encode the pair $\alpha_1 \cdots \alpha_n, I$ as the $2^{\Sigma \cup \{a\}}$-string $\alpha'_1 \cdots \alpha'_n$, for some fluent $a \in \Theta - \Sigma$, where

$$\alpha'_j = := \begin{cases} \alpha_j \cup \{a\} & \text{if } j \in I \\ \alpha_j & \text{otherwise} \end{cases}$$

for $j \in [n]$. The stutterless restriction on strings in section 2 (left out of Table 2 for simplicity) points to a conception of time as a container (containing what happens during it) that equates a stretch of time with the set of EXP-processes running over that stretch. Making a bounded granularity explicit reduces the implausibility of modeling an EXP-process as a finite automaton. But the complication discussed in the introduction relating to the Imperfective Paradox remains:

($\star$) over any stretch of time, any number of EXP-processes may run, some interfering with others.

Competition between processes deepens the EXP/HIST divide, moving away from the simple picture in Table 1 of EXP-processes reducible to HIST-runs. The primacy of timelines is challenged by a causal realm that provides *rules and regulations* over and above episodic instances recorded in a timeline (Carlson 1995, Steedman 2005).

If EXP is not reducible to HIST, might HIST be reducible to EXP? This depends on what EXP-processes are available for reducing HIST-strings to. For any string s, the singleton $\{s\}$ is a regular language, embedding HIST-models trivially into EXP-models. But implicit in the complication ($\star$) above is a view of the timeline as combining many separate EXP-processes, conceived largely in isolation and potentially clashing when run alongside other EXP-processes. That is, given a HIST-timeline s and an EXP-language m, we should ask not so much whether $s \in m$ (m being just one of the processes running in s, and thus too small to account for all of s) but rather whether there is a string $s' \in m$ that, for example, s subsumes (i.e., $s \unrhd s'$), or perhaps $s \unrhd \square^* s' \square^*$ (allowing s to extend before and/or after s'). Indeed, under ($\star$), the string $s' \in m$ may not run to completion in s, suggesting a further weakening of the condition $s \unrhd \square^* s' \square^*$. Let R be some such condition between s and s', on the basis of which we link s and m, defining

$$s' \ R\text{-}connects\ (s, m) \iff sRs' \text{ and } s' \in m.$$

We can explore different instantiations of R in HIST, by expanding a signature Σ with a copy Σ' that is disjoint, $\Sigma \cap \Sigma' = \emptyset$, and forming *model pairs* (e.g., Keisler 1977, page 71) in $\mathit{MSO}(\Sigma \cup \Sigma')$ via superposition & (as defined in section 2).

Proposition 3 *Given two disjoint finite sets Σ and Σ', $L \subseteq (2^{\Sigma \cup \Sigma'})^*$ is a regular language iff there is an ϵ-free finite-state transducer that computes the relation $\{(\rho_\Sigma(s), \rho_{\Sigma'}(s)) \mid s \in L\}$.*

An example of a relation described by Proposition 3 is subsumption $\unrhd$, with the symbols renamed for disjoint copies. String pairs (s, s') in which $s \unrhd s'$ pick out parts s' of s that are of interest, often leaving some string positions empty. In the terminology of *Carnap-Montague intensions*, the *index* s provides a context for the *denotation* s'. The pair may fall outside subsumption $\unrhd$, as in the case (1) and not (2) of the Imperfective Paradox (where the event s' of John drawing a circle may not be fully realized in the index s). The disjoint vocabularies for string pairs (s, s') distinguish what is actual according to the index s from what the denotation s' describes, allowing the strings to branch away from each other. Iterating the construction (and multiplying vocabularies), we can form any finite number of alternatives within HIST.

The sets of strings that serve as EXP-models in section 3 need not be finite. The strings in these sets may range over HIST-timelines if we take $\mathit{Fin}(\Theta)$ to be the set *Act* of symbols, finite subsets of which serve as the alphabets A from which EXP-signatures (A, q) are formed. Implicit in the simple membership $s' \in m$ in the definition of s' R-connects (s, m) is the assumption that *Act* is essentially $\mathit{Fin}(\Theta)$. But different choices of *Act* are suitable for different applications. To describe record types (which have proved useful in linguistic semantics; e.g., Cooper 2012), it is helpful to close the set $\mathit{sen}(A)$ of sentences φ under the construct $\Box_B \varphi$, for every $B \subseteq A$, with

$$q \models \Box_B \varphi \iff (\forall s \in q \cap B^*)\ q_s \models \varphi$$

for every A-state q. (The arguments in section 3 carry over with this modification.) Exactly what sentences we associate with EXP is crucial if we are to relate EXP and HIST in the manner ontologies are related in, for example, Kutz, Mossakowski and Lücke 2010. The notion of s' R-connecting (s, m) above takes a semantic approach that needs to be supplemented on the syntactic side. Much work remains to be carried out, not the least of which is an account within EXP of how to map choices of *Act* such as that made above for Davidson's (7) to the HIST-timelines represented in section 2 as $Fin(\Theta)$-strings. Are there, an anonymous referee asks, institution comorphisms between EXP and HIST?

5 Conclusion

Behind the institutions above are the correspondences

$$\frac{\text{EXP-process}}{\text{HIST-event}} \approx \frac{\text{internal mechanism}}{\text{external timeline}} \approx_{\Sigma} \frac{\text{automata}}{\text{string}}.$$

A HIST-timeline is where different EXP-processes, framed largely *in* isolation, go *out* to meet and be seen, as HIST-events. A finite approximability hypothesis attaches the subscript Σ on $\approx$, ranging over signatures that by bounding granularity, allow us to formulate the HIST-timelines as strings and the EXP-processes as finite automata. Structured as a category, the signatures provide a guide for exploring the forces that constitute the causal realm EXP, and their tortuous manifestations as events in the temporal realm HIST.

Acknowledgments I am grateful to my anonymous referees for comments, and to Science Foundation Ireland's ADAPT Centre for funding,

References

Emmon Bach 1986. The algebra of events. *Linguistics and Philosophy* 9: 5–16.

Patrick Blackburn 2006. Arthur Prior and Hybrid Logic. *Synthese* 150(3):329–372.

Janusz A. Brzozowski 1964. Derivatives of Regular Expressions. *J.ACM* 11:481–494.

Greg N. Carlson 1995. Truth conditions of generic sentences: two contrasting views. In *The Generic Book*, pp. 224–237. University of Chicago Press.

Robin Cooper 2012. Type theory and semantics in flux. In *Handbook of the Philosophy of Science. Volume 14: Philosophy of Linguistics*. pages 271–323

Robin Cooper and Aarne Ranta 2008. Natural languages as collections of resources. In *Language in Flux: Dialogue Coordination, Language Variation, Change and Evolution*, pp. 109–120. College Publications.

Donald Davidson 1967. The logical form of action sentences. In N. Rescher (ed.) *The Logic of Decision and Action*, pp 81–95, University of Pittsburgh Press.

David R. Dowty 1979. *Word Meaning and Montague Grammar*. Reidel.

Tim Fernando 2004. A finite-state approach to events in natural language semantics. *J. Logic and Computation* 14(1):79–92.

Tim Fernando 2015. The semantics of tense and aspect: a finite-state perspective. In S. Lappin and C. Fox, editors, *Handbook of Contemporary Semantic Theory*, second edition, Wiley-Blackwell.

Antony Galton 2008. Experience and history: processes and their relation to events. *J. Logic and Computation* 18(3):323–340.

Antony Galton 2012. The ontology of states, processes and events. In Mitsuhiro Okada and Barry Smith (editors), *Interdisciplinary Ontology, Vol. 5, Proceedings of the Fifth Interdisciplinary Ontology Meeting*, Tokyo. Keio University, Open Research Centre for Logic and Formal Ontology, pages 35–45.

Antony Galton and Riichiro Mizoguchi 2009. The Water Falls but the Waterfall does not Fall: New perspectives on Objects, Processes and Events. *Applied Ontology* 4(2):71–107

Joseph Goguen and Rod Burstall 1992. Institutions: abstract model theory for specification and programming. *J.ACM* 39(1):95–146.

Pierre Grenon and Barry Smith 2004. SNAP and SPAN: Towards dynamic spatial ontology. *Spatial Cognition and Computation* 4(1):69–103.

Matthew Hennessy and Robin Milner 1985. Algebraic laws for non-determinism and concurrency. *J.ACM* 32(1): 137–161.

John Hopcroft and Jeffrey Ullman 1979. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley.

H. Jerome Keisler 1977. Fundamentals of model theory, in *Handbook of Mathematical Logic*, pp 47–103 North-Holland.

Nathan Klinedinst 2012. Intensionality and the Progressive. UCL Working Papers in Linguistics 2012, pp 21–26.

Oliver Kutz, Till Mossakowski, and Dominik Lücke 2010. Carnap, Goguen, and the hyperontologies: Logical pluralism and heterogeneous structuring in ontology design. *Logica Universalis* 4(2):255–333.

Leonid Libkin 2010. *Elements of Finite Model Theory*, Springer.

John Lyons 1977. *Semantics*, Volume 2. Cambridge University Press.

John McTaggart 1908. The Unreality of Time. *Mind* 17:457–473.

Marc Moens and Mark Steedman 1988. Temporal ontology and temporal reference. *Computational Linguistics* 14(2):15–28.

Terence Parsons 1990. *Events in the Semantics of English: A Study in Subatomic Semantics*. MIT Press.

Arthur N. Prior 1967. *Past, Present and Future*. Clarendon Press, Oxford.

Mark Steedman 2005. *The Productions of Time*, Draft, http://homepages.inf.ed.ac.uk/steedman/papers.html.

Andrzej Tarlecki, Rod Burstall and Joseph Goguen 1991. Some fundamental algebraic tools for the semantics of computation: Part 3. Indexed categories. *Theoretical Computer Science* 91:239–264.