Modalities, Cohesion, and Information Flow

G. A. KAVVOS, Wesleyan University, United States of America

It is informally understood that the purpose of modal type constructors in programming calculi is to control the flow of information between types. In order to lend rigorous support to this idea, we study the category of classified sets, a variant of a denotational semantics for information flow proposed by Abadi et al. We use classified sets to prove multiple noninterference theorems for modalities of a monadic and comonadic flavour. The common machinery behind our theorems stems from the the fact that classified sets are a (weak) model of Lawvere's theory of axiomatic cohesion. In the process, we show how cohesion can be used for reasoning about multi-modal settings. This leads to the conclusion that cohesion is a particularly useful setting for the study of both information flow, but also modalities in type theory and programming languages at large.

CCS Concepts: • **Theory of computation → Modal and temporal logics**; **Type theory**; **Denotational semantics**; **Categorical semantics**; *Type structures*; • **Software and its engineering → Functional languages**; *General programming languages*;

Additional Key Words and Phrases: information flow, information flow control, type systems, modal type systems, cohesion, modal type theory, modalities, noninterference, category theory

ACM Reference Format:
G. A. Kavvos. 2019. Modalities, Cohesion, and Information Flow. *Proc. ACM Program. Lang.* 3, POPL, Article 20 (January 2019), 29 pages. https://doi.org/10.1145/3290333

1 INTRODUCTION

Taming the flow of information within a computer system has been a problem of significant interest since the early days of Computer Science; see, for example, the models of Bell and LaPadula [LaPadula and Bell 1996; Rushby 1986], and the influential work of Denning [1976], which was the first to introduce the use of *lattices* for modelling secure information flow. The objective of these models is usually to express the property that data cannot flow in or out of certain regions of a computer system, thus achieving a certain form of confidentiality or integrity.

A modern way of achieving the above objective is to make it *language-based*. That is: to enrich a programming language with features that specify or control the flow of data, so that the programs we write are correct—or rather, secure—by design. One way of doing so is through the use of *type systems* which annotate variables or program expressions with security levels. It then suffices to prove that the type system ensures some form of *noninterference property*, which invariably states that, in a well-typed program, data cannot flow contrary to our wishes, e.g. from a type labelled as being of high security to one of low security. Several type systems of this form have been proposed, but all seem to be in some sense 'equivalent;' see e.g. the recent work of Rajani and Garg [2018].

Many of theses type systems feature some form of *modality*, which is broadly construed as a unary type constructor of some sort: see e.g. [Abadi et al. 1999; Miyamoto and Igarashi 2004; Shikuma and Igarashi 2008]. The rules that govern the behaviour of a modality almost always

Author's address: G. A. Kavvos, Department of Mathematics and Computer Science, Wesleyan University, 265 Church Street, Middletown, Connecticut, 06459, United States of America, gkavvos@wesleyan.edu.

2475-1421/2019/1-ART20
https://doi.org/10.1145/3290333

silently endow it with certain information flow properties. This is intuitively well-known both by modal type theorists, as well as practitioners who use modal type systems in programming calculi.

Nevertheless, the implicit properties of these systems have not been subjected to a detailed treatment. Given the recent resurgence of interest in modal type theory—as exemplified by cohesive homotopy type theory [Shulman 2018], and guarded type theory [Clouston et al. 2016] on the theoretical side, but also calculi for functional reactive programming [Krishnaswami 2013], effects [Curien et al. 2016] or coeffects [Petricek et al. 2014] on the more application-driven end—we believe that there is a need for a more universal approach to modal type theory. There have been major recent advances on the syntactical side, particularly through the fibrational framework of Licata et al. [2017]. However, we are still lacking a 'Swiss-army-knife model' that can help us understand and prove properties about the information flow of these modalities. This is the subject of the present paper.

For this purpose we introduce a refinement of the *dependency category* of Abadi et al. [1999], which we call *the category of classified sets*. These are sets equipped with indexed logical relations that encode *indistinguishability*, and thus a form of *data hiding*. Abadi et al. [1999] used a variant of this model to prove noninterference theorems for various information flow calculi that they translated to their *dependency core calculus*,

We will present a different, significantly more 'high tech' approach. First, we notice that the relations of indistinguishability that classified sets carry must be respected: that is, if $x \ R \ y$ then $f(x) \ R \ f(y)$ for any morphism $f : X \to Y$ of our model. Thus, if $x \ R \ y$, the 'points' x and y of X can be considered to be 'arbitrarily close' to each other in a 'space.' So close, in fact, as to not be distinguishable. This analogy is rife with topological intuition. We will see that these relations describe a sort of *cohesion* between points. It so happens that an axiomatic approach to this idea has been developed by Lawvere [2007], and that classified sets form a weak model of this theory.

Once this fact is established, we can show that many of the information flow properties that we wish to establish follow directly from this abstract framework of cohesion. From that point, we want to (a) identify appropriate structure in the category of classified sets for modelling a host of modal type theories that are used in information flow control, and (b) use this structure to prove noninterference theorems for these type theories.

This paper proceeds as follows. In §2 we introduce the category of classified sets, and show that it is a finitely complete and cocomplete, bicartesian closed category, and hence a model of the simply typed λ-calculus. Then, in §3 we will present the rudiments of *axiomatic cohesion*, and show that classified sets are pre-cohesive relative to ordinary sets. This will introduce some modal operations, and lead us to prove some basic noninterference theorems in §4.

Then, in §5, we present a levelled view of cohesion. Classified sets are defined parametrically in a set of security levels $\mathcal{L}$. If we add a new set π of security levels, then the new category of sets—which is classified over $\mathcal{L} \cup \pi$—is pre-cohesive over the category of sets classified over $\mathcal{L}$. This generates some more modal operators, which are now 'level-sensitive.' These are used in §6 to prove another set of noninterference theorems. We make some concluding remarks in §7.

Category theory is used extensively throughout the paper. The main tool is that of *adjunctions*, and their close relationship to (co)monads, both of which are beautifully covered in [Awodey 2010, §9-10]. We only use one advanced concept, namely that of (co)reflective subcategories, which correspond to *idempotent* monads and comonads; this is covered in [Mac Lane 1978, §IV.3], [Borceux 1994, Vol. II, §4.2.4], or the nLab wiki.[1]

[1] https://ncatlab.org/nlab/show/reflective+subcategory

2 CLASSIFIED SETS

> Reynolds integrated these two strands of thought and formulated a general principle of relational parametricity that is applicable to a wide range of contexts for capturing the notion of "information hiding" or "abstraction." Unfortunately, we believe that the magnitude of this achievement has not been sufficiently recognized.
>
> C. Hermida, U. S. Reddy, and E. P. Robinson [Hermida et al. 2014]

Let $\mathcal{L}$ be a set of labels, which we call *security levels*. We assume precisely nothing about $\mathcal{L}$, so our theory is curiously independent of its structure (finite, infinite, partial order, lattice, etc.).

Definition 1. *A* classified set S *over* $\mathcal{L}$ *(or: a set* S *classified over* $\mathcal{L}$*) consists of*

(1) *an ordinary carrier set* $|S|$*, and*
(2) *a family of reflexive relations* $(R_\ell)_{\ell \in \mathcal{L}}$ *on* $|S|$*, one for each level* $\ell \in \mathcal{L}$*.*

We will—more often than not—write $x \in S$ to mean $x \in |S|$. The underlying intuition pertaining to a classified set is that each level $\ell \in \mathcal{L}$ is to be understood as a *security clearance*, and the relation R_ℓ models *indistinguishability* for users at that clearance. That is: if $x \; R_\ell \; y$, then a user with clearance ℓ must *not* be able to distinguish between x and y. Reflexivity models the simple fact that x should be indistinguishable to itself. In logical relations, reflexivity is a theorem; but since not everything is defined inductively here, it must become an explicit requirement.[2]

The kind of *functions* admissible in our mathematical universe shall be precisely those that map indistinguishable inputs to indistinguishable outputs.

Definition 2. *Let* S *and* S' *be sets classified over* $\mathcal{L}$*. A* morphism of classified sets $f : S \to S'$ *is a function* $f : |S| \to |S'|$ *such that* $x \; R_\ell \; y$ *implies* $f(x) \; R_\ell \; f(y)$ *for all* $\ell \in \mathcal{L}$*.*

Classified sets over $\mathcal{L}$ and their morphisms constitute a category, which we denote as $\mathbf{CSet}_{\mathcal{L}}$.

2.1 Limits and Colimits

We move on to the examination of what kind of limits and colimits exist in classified sets, which tells us which kinds of data type we are able to construct.

We let $\mathbf{1}$ be the classified set with a singleton carrier set $\{*\}$, and $* \; R_\ell \; *$ for all $\ell \in \mathcal{L}$.

Proposition 1. $\mathbf{1}$ *is a terminal object in* $\boldsymbol{CSet}_{\mathcal{L}}$*.*

That is: there is a unique function from a classified set X to $\mathbf{1}$; it maps everything to $*$, hence collapsing all related pairs to one element, which cannot be distinguished from itself.

Similarly, we let the classified set $\mathbf{0}$ be the set whose carrier is the empty set, and all of whose relations are empty. Since there are no relations to preserve, it is evident that the unique empty function from $\mathbf{0}$ to any $|A|$ is a morphism, so

Proposition 2. $\mathbf{0}$ *is an initial object in* $\boldsymbol{CSet}_{\mathcal{L}}$*.*

For classified sets A and B, we let $|A \times B| \stackrel{\text{def}}{=} |A| \times |B|$. Given any $\ell \in \mathcal{L}$, we define

$$(a_1, b_1) \; R_\ell \; (a_2, b_2) \iff a_1 \; R_\ell \; a_2 \wedge b_1 \; R_\ell \; b_2$$

Hence, two pairs are indistinguishable exactly when they are so componentwise. If, for example, the second components are distinguishable, then so are the pairs; but this does not entail that we can distinguish the first components! In this way, we can 'classify' pairs without resorting to more complicated sets of labels, e.g. $\mathcal{L} \times \mathcal{L}$.

[2] Abadi et al. [1999] did not require their relations to be reflexive, which is a key property in showing pre-cohesion in §3.

The standard set-theoretic projections preserve R_ℓ, as does the standard set-theoretic product morphism $\langle f, g\rangle : |C| \to |A| \times |B|$ for any $f : C \to A$ and $g : C \to B$. Hence,

Proposition 3. *The classified set $A \times B$ is the categorical product of A and B in $\boldsymbol{\mathcal{CSet}}_{\mathcal{L}}$.*

A similar story applies to coproducts: given A and B, we define

$$|A + B| \stackrel{\text{def}}{=} |A| + |B| = \{\, (0, a) \mid a \in |A| \,\} \cup \{\, (1, b) \mid b \in |B| \,\}$$

and, naturally, we define R_ℓ to be

$$(0, a)\ R_\ell\ (0, a') \Leftrightarrow a\ R_\ell\ a', \quad (1, b)\ R_\ell\ (1, b') \Leftrightarrow b\ R_\ell\ b'$$

and $\neg\,((i, x)\ R_\ell\ (j, y))$ for $i \neq j$. The injections $|A| \to |A| + |B|$ and $|B| \to |A| + |B|$ clearly preserve R_ℓ, as does the set-theoretic coproduct morphism, so

Proposition 4. *The classified set $A + B$ is the categorical coproduct of A and B in $\boldsymbol{\mathcal{CSet}}_{\mathcal{L}}$.*

In a similar fashion, if we are given two parallel arrows $A \overset{f}{\underset{g}{\rightrightarrows}} B$, we can see that the set $E \stackrel{\text{def}}{=} \{\, a \in A \mid f(a) = g(a) \,\}$ equipped with $R_\ell\restriction_E$, the relations R_ℓ restricted to E, is a classified set, that the inclusion $E \hookrightarrow A$ is trivially a morphism, and that

Proposition 5. *E is the equaliser of $f : A \to B$ and $g : A \to B$.*

Constructing coequalisers is slightly more complicated. Recall that given two ordinary functions $A \overset{f}{\underset{g}{\rightrightarrows}} B$, their coequaliser is the set B quotiented by the equivalence relation $\sim_{f,g}$, which is the least equivalence relation such that $(f(a), g(a)) \in \sim_{f,g}$. The elements of $B/\!\sim_{f,g}$ are then equivalence classes $[b]$ of elements we wish to 'lump together.' Suppose now that A and B are classified; how should we classify $B/\!\sim_{f,g}$? We may consider two of its equivalence classes indistinguishable whenever it happens that two elements, one from each equivalence class, are 'lumped together' by R_ℓ. So we define

$$[b]\ R_\ell\ [b'] \Longleftrightarrow \exists x \in [b].\ \exists y \in [b'].\ x\ R_\ell\ y$$

and thus turn $B/\!\sim_{f,g}$ into a classified set. The quotient map $B \to B/\!\sim_{f,g}$ is then automatically a morphism of classified sets, and it is not hard to show that

Proposition 6. *$B/\!\sim_{f,g}$ is the coequaliser of $A \overset{f}{\underset{g}{\rightrightarrows}} B$ $\mathbf{CSet}_{\mathcal{L}}$.*

In short, we have the following theorem:

Theorem 1. *$\mathbf{CSet}_{\mathcal{L}}$ is finitely complete and finitely cocomplete.*

2.2 Exponentials

In a manner identical to that of logical relations, we are able endow the set of morphisms from a classified set to another with an indistinguishability relation. The idea is that two morphisms are indistinguishable if they map indistinguishable inputs to indistinguishable outputs. Note that this furnishes our theory with an *extensional* view of functions, where they are understood to be indistinguishable precisely when their input-output behaviour is.

Given classified sets A and B over $\mathcal{L}$, we define the classified set B^A by

$$\left|B^A\right| \stackrel{\text{def}}{=} \mathrm{Hom}_{\mathbf{CSet}_{\mathcal{L}}}(A, B)$$
$$f\ R_\ell\ g \Longleftrightarrow \forall a\ R_\ell\ a'.\ f(a)\ R_\ell\ g(a')$$

This is the usual definition of logical relations at function types. We can then define a function $\mathrm{ev} : B^A \times A \to B$ by $(f, a) \mapsto f(a)$, and the definition of B^A makes it a morphism. From that point, it is trivial to show that

Proposition 7. *B^A is the exponential of A and B.*

Hence,

Theorem 2. *$\mathbf{CSet}_{\mathcal{L}}$ is a bicartesian closed category.*

Thus the category $\mathbf{CSet}_{\mathcal{L}}$ is rich enough to model the simply typed λ-calculus with coproducts. In the following sections we will also show that there is enough structure to model a multitude of *modal operators* on types.

3 COHESION

The modal structure on classified sets is closely related to—or, more precisely, induced by—Lawvere's *axiomatic cohesion*. But what is axiomatic cohesion? It is a theory developed by Lawvere [2007] as an attempt to capture the very broad idea of *mathematical spaces* that are endowed with some kind of *cohesion*, i.e. the idea that some points are 'very close to each other' or 'stuck together.' The prototypical example is that of *topological spaces*, which are sets of points equipped with a *topology*, i.e. a set of subsets of these points. These are called the *open sets*, and the choice of which subsets are open endows the underlying set with notions of continuity, connectedness, convergence etc.

If we write **Top** for the category of topological spaces, there is an obvious forgetful functor $U : \mathbf{Top} \longrightarrow \mathbf{Set}$ which 'forgets' the cohesive structure of a topological space, and returns the underlying set of points. Conversely, given any set there are two canonical ways to construct a topological space. The first is to specify that *no points are stuck together*. This is achieved by the *discrete topology*, where every subset is an open set. It forms a functor,

$$\Delta : \mathbf{Set} \longrightarrow \mathbf{Top}$$

that is the identity on functions: every function on sets is trivially a continuous function between the same sets seen as discrete spaces, so Δ is full and faithful.

The other way is to specify that *all points are stuck together*, and is achieved by endowing the set with the *codiscrete topology*, where the only open sets are the empty set and the entire space. It forms another functor,

$$\nabla : \mathbf{Set} \longrightarrow \mathbf{Top}$$

that is the identity on functions: every function on sets is trivially a continuous function between the same sets seen as codiscrete spaces.

The relationship between these functors is simple: they form a *string of adjoints*:

$$\Delta \dashv U \dashv \nabla$$

This categorifies the following two simple observations: every function $X \to \nabla Y$ from a topological space into a codiscrete space is continuous (as everything is collapsed into a single block of points); and every function $\Delta X \to Y$ from a discrete space into a topological space Y is continuous (as there is no cohesion to preserve in a discrete space).

But this is not the whole story. Consider a continuous function $X \to \Delta Y$ into the discrete space with points Y. Since it is continuous, it must preserve cohesion. Namely, it must map points of X that are 'stuck together' to points that are 'stuck together' in ΔY. But points are only 'stuck' to themselves in the discrete space ΔY, so in fact it must map all points 'stuck' to each other in X to a single point of Y. Thus, if we could somehow reduce X down to a set $C(X)$ where points 'stuck' together are collapsed to a single point, the continuous function $X \to \Delta Y$ would define an

ordinary function $C(X) \to Y$. Such a functor C *does* exist, and maps the topological space X to its set $C(X)$ of *connected components*. It is evidently left adjoint to Δ.

We are very close to showing that topological spaces are *cohesive relative to sets*. We will, however, not use this full notion in this paper, as the weaker notion of *pre-cohesion*, due to Lawvere and Menni [2015], is more than sufficient for our purposes:

Definition 3. *In a situation of the form*

$$\mathcal{E} \quad C \downarrow \dashv \uparrow \Delta \dashv \downarrow U \dashv \uparrow \nabla \quad \mathcal{S}$$

where $\mathcal{E}$ and $\mathcal{S}$ are extensive categories, we call $\mathcal{E}$ pre-cohesive relative to $\mathcal{S}$ *if*

(1) $\Delta, \nabla : \mathcal{S} \longrightarrow \mathcal{E}$ *are full and faithful;*
(2) $C : \mathcal{E} \longrightarrow \mathcal{S}$ *preserves finite products; and*
(3) *the* Nullstellensatz *holds: the counit $Id_{\mathcal{E}} \Rightarrow \Delta C$ is an epimorphism; or, equivalently, the unit $\Delta U \Rightarrow Id_{\mathcal{E}}$ is a monomorphism.*

The second requirement essentially expresses that a connected component of a product space is exactly a connected component in each of the two components. The third requirement, the *nullstellensatz*, has many equivalent forms, and essentially requires that the 'quotient' map that maps a point to its connected component is a epimorphic, i.e. a kind of abstract surjection (in other words: no connected component is empty).

The reason that a setting of pre-cohesion is of direct interest to modal type theory is that it automatically induces *three modalities* on the category $\mathcal{E}$ by composing each pair of adjoints. The first one,

$$\Box \stackrel{\text{def}}{=} \Delta U : \mathcal{E} \longrightarrow \mathcal{E}$$

is a comonad. Intuitively, $\Box$ takes a cohesive space, strips its points of their cohesive structure, and gives them the discrete structure: it 'unsticks' all points. The second one,

$$\blacklozenge \stackrel{\text{def}}{=} \nabla U : \mathcal{E} \longrightarrow \mathcal{E}$$

is a monad, which does the opposite: it 'sticks' all the points of a cohesive space together. Finally,

$$\int \stackrel{\text{def}}{=} \Delta C : \mathcal{E} \longrightarrow \mathcal{E}$$

is a monad. Intuitively, $\int$ collapses each connected component into a single point, and then presents that set of connected components as a discrete space.

Summarising, the above setting endows these functors with the following properties:

Corollary 1 (Fundamental Corollary of Pre-Cohesion).

(1) $U : \mathcal{E} \longrightarrow \mathcal{S}$ *preserves limits and colimits.*
(2) $\Delta : \mathcal{S} \longrightarrow \mathcal{E}$ *preserves limits and colimits.*
(3) $\nabla : \mathcal{S} \longrightarrow \mathcal{E}$ *preserves limits.*
(4) $C : \mathcal{E} \longrightarrow \mathcal{S}$ *preserves products and colimits.*
(5) $U\Delta \cong Id_{\mathcal{S}} : \mathcal{S} \longrightarrow \mathcal{S}$
(6) $U\nabla \cong Id_{\mathcal{S}} : \mathcal{S} \longrightarrow \mathcal{S}$
(7) $\Box \stackrel{def}{=} \Delta U : \mathcal{E} \longrightarrow \mathcal{E}$ *is an idempotent comonad. It is* exact, *i.e. preserves finite limits and colimits.*
(8) $\blacklozenge \stackrel{def}{=} \nabla U : \mathcal{E} \longrightarrow \mathcal{E}$ *is an idempotent monad. It is* left exact, *i.e. preserves finite limits.*
(9) $\int \stackrel{def}{=} \Delta C : \mathcal{E} \longrightarrow \mathcal{E}$ *is an idempotent monad. It preserves products and colimits.*

(10) $\int \dashv \Box \dashv \blacklozenge$

Proof. (1)-(4) follow by the fact each of these functors is a left or right adjoint, or by some assumption. (5) and (6) follow by the Yoneda lemma and the fact Δ/∇ are f.f.; e.g. for any $D \in \mathcal{S}$

$$\mathrm{Hom}_{\mathcal{S}}(D, U\Delta A) \cong \mathrm{Hom}_{\mathcal{E}}(\Delta D, \Delta A) \cong \mathrm{Hom}_{\mathcal{S}}(D, A)$$

(7)-(9) follow from (1)-(4) and the fact Δ/∇ are full and faithful and thus generate idempotent (co)monads (see e.g. [Borceux 1994, §4.3.2]). (10) follows from $C \dashv \Delta \dashv U \dashv \nabla$. □

The rest of the section is devoted to showing that

Theorem 3. $\mathbf{CSet}_{\mathcal{L}}$ *is pre-cohesive relative to* **Set**.

A variant of this fact (namely that reversible reflexive graphs are cohesive relative to sets) is already present in [Lawvere 2007]. There is an evident forgetful functor from classified sets to sets:

$$U : \mathbf{CSet}_{\mathcal{L}} \longrightarrow \mathbf{Set}$$
$$X \longmapsto |X|$$

which forget all the relations R_ℓ. We can then return to classified sets using the functor

$$\Delta : \mathbf{Set} \longrightarrow \mathbf{CSet}_{\mathcal{L}}$$

which adds to the set X the diagonal relation $x\ R_\ell\ x$ at each level $\ell \in \mathcal{L}$. This is the *finest* equality that can be supported by this setting, in that each element is only indistinguishable to itself. In that sense, ΔX is a classified set with carrier X that is *completely transparent*, in a manner reminiscent of the discrete topology on a set. Any function $f : X \to Y$ is trivially a function $f : \Delta X \to \Delta Y$, as the diagonal relation is trivially preserved; thus Δ is indeed a functor. Moreover, it is easy to see that it is full and faithful, and that

Proposition 8. $\Delta : \boldsymbol{Set} \longrightarrow \boldsymbol{CSet}_{\mathcal{L}}$ *is left adjoint to the forgetful functor.*

Proof. Any morphism $f : \Delta X \to Y$ is a function $f : X \to |Y| = UY$. Conversely, any function $f : X \to UY$ can be seen as a morphism $f : \Delta X \to Y$ of classified sets, as it trivially preserves the diagonal relation. Naturality is trivial. □

We then define

$$\nabla : \mathbf{Set} \longrightarrow \mathbf{CSet}_{\mathcal{L}}$$

to map a set X to itself, but equipped with the *complete relation* $R_\ell \overset{\text{def}}{=} |X| \times |X|$ at each $\ell \in \mathcal{L}$. That is: no element of ∇X is distinguishable from any other. Thus ∇X is the classified set with carrier X that is *maximally opaque*. This reminds us of the codiscrete topology on X. Again, rather trivially, any function $f : X \to Y$ can be seen as a morphism $f : \nabla X \to \nabla Y$, as it evidently preserves the complete relation on X. It is not hard to see that ∇ is a full and faithful functor, and that

Proposition 9. $\nabla : \boldsymbol{Set} \longrightarrow \boldsymbol{CSet}_{\mathcal{L}}$ *is right adjoint to the forgetful functor.*

Proof. Any function $f : |X| \to Y$ can be seen as a morphism $f : X \to \nabla Y$: it trivially preserves all the related pairs in X—no matter what they are—for ∇Y relates all elements of Y. Conversely, any morphism $f : X \to \nabla Y$ is simply a function $f : |X| \to |\nabla Y| = Y$. Naturality is again trivial. □

We now move on to connected components. Suppose we have a morphism $f : X \to \Delta Y$. Then, as each R_ℓ in ΔY is simply reflexivity, we have that for any $\ell \in \mathcal{L}$,

$$x\ R_\ell\ x' \text{ (in } X) \implies f(x)\ R_\ell\ f(x') \text{ (in } \Delta Y) \implies f(x) = f(x')$$

That is: f collapses related elements of X that are related at some—any!—level to a single element in Y. We cannot phrase this in terms of quotients yet, for R_ℓ need not be an equivalence relation.

So, let us define the relation $R^\star \subseteq |X| \times |X|$ to be the reflexive, symmetric, transitive closure of $\bigcup_{\ell \in \mathcal{L}} R_\ell$, i.e. the *least* equivalence relation containing all the R_ℓ's. We can now define $|X|/R^\star$. This extends to a functor

$$C : \mathbf{CSet}_{\mathcal{L}} \longrightarrow \mathbf{Set}$$

by letting $C(X) \stackrel{\text{def}}{=} |X|/R^\star$, and defining $Cf \stackrel{\text{def}}{=} f^\star : |X|/R^\star \to |Y|/R^\star$, where

$$f^\star([x]) \stackrel{\text{def}}{=} [f(x)]$$

$f^\star$ is well-defined: if $x\ R^\star\ x'$, then there is a (possibly empty) sequence $x_0, \dots, x_n$ of elements of X and a sequence of levels $\ell_0, \dots, \ell_{n+1}$ such that

$$x\ R_{\ell_0}\ x_0\ R^{-1}_{\ell_1}\ x_1 \cdots R^{-1}_{\ell_{n-1}}\ x_{n-1}\ R_{\ell_n}\ x_n\ R_{\ell_{n+1}}\ x'$$

But f preserves all of these, mapping inverses to inverses, so

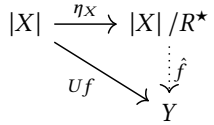
$$f(x)\ R_{\ell_0}\ f(x_0)\ R^{-1}_{\ell_1}\ f(x_1) \cdots R^{-1}_{\ell_{n-1}}\ f(x_{n-1})\ R^{-1}_{\ell_n}\ f(x_n)\ R_{\ell_{n+1}}\ f(x')$$

and hence $[f(x)] = [f(x')]$, so the choice of equivalence class representative is immaterial. We hence obtain a functor that gives us for each classified set X its set of connected components, i.e. the set of equivalence classes $|X|/R^\star$. Any two elements in the same equivalence class are indistinguishable at some level $\ell \in \mathcal{L}$.

Let us return to the function $f : X \to \Delta Y$. Define the relation

$$x \sim_f x' \quad \stackrel{\text{def}}{=} \quad f(x) = f(x')$$

This is clearly an equivalence relation, and we know that $\bigcup_{\ell \in \mathcal{L}} R_\ell \subseteq \sim_f$. But $R^\star$ is the least equivalence relation such that the above is true, so $R^\star \subseteq \sim_f$. Hence, $Uf : |X| \to Y$, which respects $\bigcup_\ell R_\ell$, can be uniquely factored as



where η_X is the quotient map, and $\hat{f}([x]) \stackrel{\text{def}}{=} f(x)$. So we can uniquely take $f : X \to \Delta Y$ to $\hat{f} : CX \to Y$, and

Proposition 10. *$C : \mathbf{CSet}_{\mathcal{L}} \to \mathbf{Set}$ is left adjoint to $\Delta : \mathbf{Set} \to \mathbf{CSet}_{\mathcal{L}}$.*

It is easy to see that the canonical map $C(X \times Y) \to CX \times CY$ is $[(x, y)] \mapsto ([x], [y])$, and that, due to the behaviour of the relations on the product, it has an inverse. Moreover, in the case of classified sets, the *nullstellensatz* is trivial: applying the hom-set isomorphism of $\Delta \dashv U$ to the identity $id_{UX} : UX \to UX$, which is really the identity function $id_{|X|} : |X| \to |X|$, yields the unit $\Delta UX \to X$, which is again the identity function on $\Delta UX = \Delta\,|X|$, which is injective and hence monic.

Thus, Theorem 3 holds.

4 NONINTERFERENCE I: MONADS AND COMONADS

The cohesive structure of classified sets that we have developed so far is enough to show some basic information flow properties for monadic and comonadic calculi. In this section, we will state and prove noninterference properties for (a) Moggi's monadic metalanguage, and (b) the Davies-Pfenning calculus. Both of these properties follow from an axiom of cohesion, which we call *codiscrete contractibility*, and which we discuss first.

Central to our noninterference proofs will be two canonically constructed objects; they will be classified sets over the carrier

$$\mathbb{B} \stackrel{\text{def}}{=} \{\mathtt{tt}, \mathtt{ff}\}$$

of booleans. The first one is the 'discrete' booleans $\Delta\mathbb{B}$, in which if $b\ R_\ell\ b'$ it follows that $b = b'$: these are the booleans that are visible to everyone.[3] The second one will be the 'codiscrete' booleans $\nabla\mathbb{B}$, in which $b\ R_\ell\ b'$ for all pairs of booleans (b, b'), which are invisible at all levels. Note that

$$\begin{aligned}\mathbb{B} &\cong \mathbf{1} + \mathbf{1}\\ \Delta\mathbb{B} &\cong \Delta\mathbf{1} + \Delta\mathbf{1} \cong \mathbf{1} + \mathbf{1}\\ \nabla\mathbb{B} &\cong \nabla U\Delta\mathbb{B} \cong \blacklozenge(\mathbf{1} + \mathbf{1})\end{aligned}$$

These all follow from the fundamental corollary (Corollary 1). The first isomorphism is by definition. The second holds because Δ preserves isomorphisms, colimits (+) and limits ($\mathbf{1}$). The final one is obtained by using $U\Delta \cong \mathsf{Id}$, then the second one, and then the definition $\blacklozenge \stackrel{\text{def}}{=} \nabla U$.

4.1 Contractible Codiscreteness and Information Flow

We are now ready to discuss a certain axiom that pre-cohesions may or may not satisfy. This axiom expresses a very intuitive property: it says that when all points are stuck together they constitute at most one connected component. Surprisingly, this single axiom is the main point of connection between cohesion and information flow: we will use it to prove results which are at the core of our noninterference theorems, both here and in §6.

We have carefully described the string of adjoints $C \dashv \Delta \dashv U \dashv \nabla$ that expresses the idea of pre-cohesion. In particular, ∇X is intuitively understood to be the object with point-set X equipped with maximal cohesion, i.e. 'everything stuck together.' A property that is should follow from this intuition is that if one were to look at the connected components of ∇X, one would find *at most one*—or none, if X is the empty set. Unfortunately, this is not something that readily follows from pre-cohesion, but we can explicitly ask for it.

Definition 4. *If $\mathcal{E}$ is pre-cohesive relative to $\mathcal{S}$ with $C \dashv \Delta \dashv U \dashv \nabla$, then this pre-cohesion satisfies* contractible codiscreteness *if codiscrete objects have at most a single connected component. That is: if for any $X \in \mathcal{S}$, $C(\nabla X)$ is a subobject of the terminal object $\mathbf{1}$, i.e. the unique morphism*

$$C(\nabla X) \to \mathbf{1}$$

is monic.

Lawvere and Menni [2015] consider this concept in toposes, and call it *connected codiscreteness*. The name for our more general setting is due to Shulman [2018].

It is easy to see that

Proposition 11. *The pre-cohesion of $\boldsymbol{CSet}_{\mathcal{L}}$ relative to $\boldsymbol{Set}$ satisfies contractible codiscreteness.*

This is a fancy way of stating the following obvious fact: if we start with a set X, equip it with the complete relation at each level $\ell \in \mathcal{L}$, and we then take the connected components of that, there will be at most one. In fact, if $X = \emptyset$ there will be none, and otherwise there will be exactly one.

The axiom of contractible codiscreteness is very useful, as it allows one to prove *abstract noninterference properties*. We will prove such a property presently, but we first have to discuss its slightly thorny interaction with non-emptiness in categorical terms.

Suppose the object X is *non-empty*, i.e. there is a point $x : \mathbf{1} \to X$. This is a sufficient condition to show that $C(\nabla X)$ is actually 'contractible,' i.e. isomorphic to the terminal object.

[3] They are referred to as the 'low security booleans' by Abadi et al. [1999], and denoted *boolL*.

Proposition 12 (Contractibility of non-empty codiscrete spaces). *Let $\mathcal{E}$ be pre-cohesive over $\mathcal{S}$ in a way that satisfies contractible codiscreteness. If $X \in \mathcal{S}$ is non-empty, then*

$$C(\nabla X) \cong \mathbf{1}$$

Proof. Let $x : \mathbf{1} \to X$. By applying $C\nabla$ to it, and then using preservation of products (which we have by Lemma 1), we obtain a point $\mathbf{1} \to C\nabla X$, which is to say that $C\nabla X$ is non-empty too. The composite $\mathbf{1} \to C\nabla X \to \mathbf{1}$ is trivially the identity. Moreover, the composite $C\nabla X \to \mathbf{1} \to C\nabla X$ is the identity on $C\nabla X$: as $C\nabla X \to \mathbf{1}$ is monic, any two morphisms into $C\nabla X$ are equal. □

We can now state and prove the following.

Proposition 13. *Let $\mathcal{E}$ be pre-cohesive over $\mathcal{S}$, in a way that satisfies contractible codiscreteness. Then*

(1) *Morphisms $\blacklozenge A \to \Delta B$ for non-empty $A \in \mathcal{E}$ and $B \in \mathcal{S}$ correspond to points $\mathbf{1} \to B$.*
(2) *Morphisms $\nabla A \to \square B$ for non-empty $A \in \mathcal{S}$ and $B \in \mathcal{E}$ correspond to points $\mathbf{1} \to UB$.*

Proof.

(1) The following isomorphisms hold naturally:

$$\begin{aligned} \mathcal{E}(\blacklozenge A, \Delta B) &\cong \mathcal{E}(\nabla UA, \Delta B) && \text{by definition of } \blacklozenge \\ &\cong \mathcal{S}(C\nabla UA, B) && \text{as } C \dashv \Delta \\ &\cong \mathcal{S}(\mathbf{1}, B) && \text{by Prop. 12} \end{aligned}$$

(2) The following isomorphisms hold naturally:

$$\begin{aligned} \mathcal{E}(\nabla A, \square B) &\cong \mathcal{E}(\nabla A, \Delta UB) && \text{by definition of } \square \\ &\cong \mathcal{S}(C\nabla A, UB) && \text{as } C \dashv \Delta \\ &\cong \mathcal{S}(\mathbf{1}, UB) && \text{by Prop. 12} \end{aligned}$$

□

In the concrete case of $\mathbf{CSet}_{\mathcal{L}}$ the above lemma says that if A is non-empty then morphisms of type $\blacklozenge A \to \Delta B$ and $\Delta A \to \square B$ are *constant functions*. Indeed, the first isomorphism in either proof is an identity, and the second one collapses all elements of ∇UA or ∇A to a single element.

A special case of (1), 'manually' proven for the particular case of $\mathbf{CSet}_{\mathcal{L}}$ and $B = \mathbb{B}$, forms the central reasoning involved in the noninterference proofs of Abadi et al. [1999]. Our result generalises this to any setting of pre-cohesion, and also yields the heretofore unnoticed dual (2).

We will now apply (1) and (2) to construct two noninterference proofs.

4.2 Moggi's Monadic Metalanguage

Moggi [1991] introduced the *monadic metalanguage*, a typed λ-calculus which, for each type A features a type of *computations* TA. The 'result' of a computation $M : TA$ is a 'value' of type A. Indeed, each 'value' is a trivial computation, as can be witnessed by the introduction rule:

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash [M] : TA}$$

The idea is that types of the form TA encapsulate various 'notions of computations,' nowadays referred to as *effects* (recursion, nondeterminism, operations on the store, etc.). Very little is assumed of the type constructor. To quote Moggi:

> "Rather than focusing on a specific T, we want to find the general properties common to all notions of computation."

Moggi then showed in [Moggi 1991, §3] that there is a categorical semantics of this language in which T is interpreted by a *strong monad* on a CCC $\mathcal{C}$. The category $\mathcal{C}$ is then the *category of values*, whereas the Kleisli category $\mathcal{C}_T$ of the monad $T : \mathcal{C} \longrightarrow \mathcal{C}$ is the *category of programs*.

The key information flow property enjoyed by monads is that once something is 'inserted' into the monad, then it cannot flow out. The monad identifies a region of the language which is *impure*, in that evaluating terms within the region causes *effects*. If the results of those effects were to flow 'outside' the monad, the outcome would be the loss of referential transparency. This information flow property is evident if one looks at the elimination rule for T:

$$\frac{\Gamma \vdash M : TA \quad \Gamma, x : A \vdash N : TB}{\Gamma \vdash \mathsf{let}\ x = M\ \mathsf{in}\ N : TB}$$

If we have a computation $M : TA$ that yields a result of type A, the let construct allows us to substitute it for a variable of type A, but only as long as this to be used within *another* computation $N : TB$. Thus values of type TA cannot 'escape' the scope of T.

It is not hard to show that $\mathbf{CSet}_{\mathcal{L}}$ is a model of Moggi's monadic metalanguage (with booleans), with T being interpreted by $\blacklozenge$, and Bool by $\Delta\mathbb{B} \cong \mathbf{1} + \mathbf{1}$: the only thing that remains to be shown is that $\blacklozenge$ is a strong monad, which is true by a more general result that we cover later (Prop. 18). If we use the standard categorical interpretation, as defined by [Moggi 1991, §3], a term $x : TA \vdash M : \mathsf{Bool}$ is interpreted as a morphism $\blacklozenge [\![A]\!] \to \Delta\mathbb{B}$ in $\mathbf{CSet}_{\mathcal{L}}$. Hence, by Prop. 13 we know it is interpreted by a constant function in the model.

Nevertheless, this is not a noninterference result yet, for it tells us something about a particular model, viz. $\mathbf{CSet}_{\mathcal{L}}$, but nothing about the equational theory of Moggi's calculus. To infer something about it from one of its models, some *completeness property* of that model must be established. We will use the simplest one of all, which is known as *adequacy*. It is the most basic form of completeness, and is commonly used in the study of calculi with recursion, e.g. PCF [Plotkin 1977; Streicher 2006]. A model of a calculus is adequate exactly when it is complete at *ground types*. What we take to be a ground type is up to us, but the usual choices are simple base types (booleans, flat naturals, etc.) For this paper, we use the term to refer to types defined by a collection of constants, along with elimination and computation rules. For example, Bool is a ground type: it may be presented by the constants

$$\Gamma \vdash \mathtt{tt} : \mathsf{Bool} \qquad \Gamma \vdash \mathtt{ff} : \mathsf{Bool}$$

along with the the elimination rule

$$\frac{\Gamma \vdash M : \mathsf{Bool} \quad \Gamma \vdash E_0 : C \quad \Gamma \vdash E_1 : C}{\Gamma \vdash \mathsf{if}\ M\ \mathsf{then}\ E_1\ \mathsf{else}\ E_0 : C}$$

and the two computation rules

$$\mathsf{if}\ \mathtt{tt}\ \mathsf{then}\ E_1\ \mathsf{else}\ E_0 = E_1$$
$$\mathsf{if}\ \mathtt{ff}\ \mathsf{then}\ E_1\ \mathsf{else}\ E_0 = E_0$$

Suppose that a (non-dependent, possibly modal) type theory satisfies *canonicity at all ground types*, i.e. for every such ground type G and every *closed* term $\vdash M : G$ there is a unique constant $\mathsf{c} : G$ such that $\vdash M = c : G$. For example, canonicity for Bool requires that every closed term $\vdash M : \mathsf{Bool}$ is equal to either $\mathtt{tt}$ or $\mathtt{ff}$. In strongly normalising programming calculi, canonicity is a corollary of *progress and preservation theorems* [Pierce 2002, §8.3, §9]: the normalisation of a well-typed closed term will reach a *canonical form*. In type theories, canonicity is a corollary of *confluence and strong normalisation*: each closed term can be reduced to a unique normal form, and each closed normal form of type A can only correspond an introduction rules of type A, which for ground types are simply constants.

Suppose now that we interpret the type theory in a category in the standard way—e.g. as in [Abramsky and Tzevelekos 2011; Crole 1993]—so that (a) closed terms $\vdash M : G$ are interpreted as points $\mathbf{1} \to [\![G]\!]$, (b) the interpretation is sound, in that $\vdash M = N : A$ implies $[\![M]\!] = [\![N]\!]$, and (c) the interpretation of each ground type G is *injective*, in that distinct constants have distinct interpretations. Then, the interpretation is automatically adequate:

Lemma 1 (Adequacy). *Suppose that a type theory satisfies canonicity at ground types, and has a sound categorical interpretation which is injective at every ground type G, in the sense that*

$$[\![\vdash c_i : G]\!] = [\![\vdash c_j : G]\!] : \mathbf{1} \to [\![G]\!] \quad \Longrightarrow \quad c_i \equiv c_j$$

Then this interpretation is adequate *for G, in the sense that*

$$[\![\vdash M : G]\!] = [\![\vdash c_i : G]\!] \quad \Longrightarrow \quad \vdash M = c_i : G$$

Proof. By canonicity, for any $\vdash M : G$ we have $\vdash M = c_j : G$ for some constant c_j of G. But then $[\![\mathrm{c}_i]\!] = [\![M]\!] = [\![\mathrm{c}_j]\!]$ by soundness, so $\mathrm{c}_i \equiv \mathrm{c}_j$, and hence $\vdash M = \mathrm{c}_i : G$. □

This lemma applies to the standard interpretation of Moggi's monadic metalanguage into any CCC with a strong monad. It is known that the straightforward extension of Moggi's calculus with coproducts and a unit type (which together subsume Bool) is confluent, strongly normalising, and has a sound interpretation into any biCCC with a strong monad: this was shown by Benton et al. [1998]. Thus, it satisfies canonicity at Bool. Hence, the above lemma still applies, and we obtain

Theorem 4 (Noninterference for Moggi). *Let A be a non-empty type, which is to say there exists a closed term of type A. If $x : TA \vdash M : \mathit{Bool}$ then for any $\vdash E, E' : TA$ we have*

$$\vdash M[E/x] = M[E'/x] : \mathit{Bool}$$

Proof. The interpretation of Bool as $\Delta\mathbb{B} \cong \mathbf{1} + \mathbf{1}$ in $\mathbf{CSet}_{\mathcal{L}}$ satisfies the assumptions of Lemma 1. Hence, the interpretation is adequate for it. We have that

$$[\![\vdash M[F/x] : \mathrm{Bool}]\!] = [\![x : TA \vdash M : \mathrm{Bool}]\!] \circ [\![\vdash F : TA]\!] : \mathbf{1} \to \Delta\mathbb{B}$$

for any $\vdash F : TA$. But as A is non-empty, we have that $[\![A]\!]$ is non-empty, so by Proposition 13(1), $[\![x : TA \vdash M : \mathrm{Bool}]\!] : \blacklozenge\, [\![A]\!] \to \Delta\mathbb{B}$ is a constant function, so

$$[\![M[E/x]]\!] = [\![x : TA \vdash M : \mathrm{Bool}]\!] \circ [\![E]\!] = [\![x : TA \vdash M : \mathrm{Bool}]\!] \circ [\![E']\!] = [\![M[E'/x]]\!]$$

for any $\vdash E, E' : TA$. By adequacy, it follows that $\vdash M[E/x] = M[E'/x] : \mathrm{Bool}$. □

4.3 The Davies-Pfenning Calculus

Davies and Pfenning introduced a comonadic modal type theory as a type system for binding-time analysis in [Davies and Pfenning 1996, 2001]. The idea was that data could not arbitrarily flow from type A to type $\Box A$. This is clearly reflected in the structure of the type system: each typing judgement comes with two contexts, and has the shape

$$\Delta \mid \Gamma \vdash M : A$$

where Δ are the 'modal' variables, and Γ are normal variables. A 'modal' variable can be used as a normal variable, as witnessed by the rule $\Delta, u : A \mid \Gamma \vdash u : A$. However, when introducing terms of type $\Box A$, we can only use 'modal' variables from Δ:

$$\frac{\Delta \mid \cdot \vdash M : A}{\Delta \mid \Gamma \vdash \mathrm{box}\ M : \Box A}$$

It is not hard to define a non-trivial morphism $\mathrm{Bool} \to \Box\mathrm{Bool}$:

$$b : \mathrm{Bool} \vdash \ \mathrm{if}\ b\ \mathrm{then\ box\ tt\ else\ box\ ff} : \Box\mathrm{Bool}$$

However, it is impossible to pass from $A \to B$ to $\Box(A \to B)$ in general. This property was used by Davies and Pfenning to separate things that were available *statically*, by isolating them under the box, from things that are available *dynamically* (and thus cannot always be used for metaprogramming). For example, one can always use a boolean constant for metaprogramming, as shown above, but one cannot use 'live' piece of code, a function $A \to B$. However, if one has prudently arranged to have a copy of the source code of a function $A \to B$ available, that would be of type $\Box(A \to B)$, and one can then use it for metaprogramming.

Terms of type $\Box A$ are used by substituting for a variable in the 'modal' context. This yields the following elimination rule:

$$\frac{\Delta \mid \Gamma \vdash M : \Box A \quad \Delta, u : A \mid \Gamma \vdash N : B}{\Delta \mid \Gamma \vdash \mathsf{let\ box}\ u = M\ \mathsf{in}\ N : B}$$

along with the reduction $\mathsf{let\ box}\ u = \mathsf{box}\ M\ \mathsf{in}\ N \to N[M/u]$.

Even though the information flow properties of the Davies-Pfenning calculus are intuitive, it is not at all evident how to express them as a noninterference theorem: whilst in Moggi's calculus the monad T blocked all information flow out of it, the Davies-Pfenning calculus allows *some* flow into the modal types, as witnessed by the non-trivial morphism $\mathsf{Bool} \to \Box\mathsf{Bool}$ constructed above.

The way out of this impasse is to consider the dual of the statement used to prove noninterference for the metalanguage, namely Proposition 13(2). In the place of booleans, we will use the *codiscrete booleans* Bool_∇. These are introduced explicitly by

$$\Delta \mid \Gamma \vdash \mathtt{tt} : \mathsf{Bool}_\nabla \qquad \Delta \mid \Gamma \vdash \mathtt{ff} : \mathsf{Bool}_\nabla$$

and the elimination rule

$$\frac{\Gamma \vdash M : \mathsf{Bool}_\nabla \quad \Gamma \vdash E_0 : C \quad \Gamma \vdash E_1 : C \quad C \text{ is codiscrete}}{\Gamma \vdash \mathsf{if}\ M\ \mathsf{then}\ E_1\ \mathsf{else}\ E_0 : C}$$

with the same computation rules as before. The side condition that C is codiscrete is defined by (I) unit (if we have it) and Bool_∇ are codiscrete; (II) if A and B are codiscrete, then so is $A \times B$; (III) if B is codiscrete, then so is $A \to B$. We will discover the roots of this definition in §5.1. For now, let us say that the intended meaning is that, when interpreted in $\mathsf{CSet}_{\mathcal{L}}$, the object $[\![C]\!]$ will be isomorphic to $\blacklozenge [\![C]\!]$.

Now, let us note that the Davies-Pfenning calculus satisfies confluence, and strong normalisation [Kavvos 2017a,b], and hence canonicity, with canonical forms at type $\Box A$ of the form box M, with M is a canonical form at type A. It is straightforward to re-establish canonicity after the addition of codiscrete booleans. The Davies-Pfenning calculus also has a standard interpretation in any CCC with a product-preserving comonad on it: see e.g. [Hofmann 1999; Kavvos 2017a]. we may again show adequacy:

Lemma 2 (Davies-Pfenning Adequacy). *Suppose we have a categorical model for the Davies-Pfenning calculus, as well as a ground type G that satisfies canonicity along with an injective interpretation in that model, so that*

$$[\![\vdash c_i : G]\!] = [\![\vdash c_j : G]\!] : \mathbf{1} \to [\![G]\!] \quad \Longrightarrow \quad c_i \equiv c_j$$

Then this interpretation is adequate for $\Box G$, *in the sense that*

$$[\![\vdash M : \Box G]\!] = [\![\vdash \mathsf{box}\ c_i : \Box G]\!] \quad \Longrightarrow \quad \vdash M = \mathsf{box}\ c_i : \Box G$$

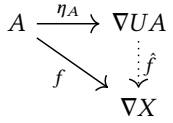
Proof. The canonical forms of type $\Box G$ are precisely those of the form box c_i. We therefore have that $\vdash M = \mathsf{box}\ \mathrm{c}_i : \Box G$ for some c_i. But then, if $[\![M]\!] = [\![\mathsf{box}\ \mathrm{c}_j]\!]$ we have—writing $(-)^*$ for

the co-Kleisli extension and using soundness—that

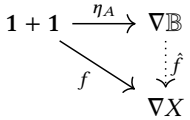
$$\left[\!\left[\mathrm{c}_j\right]\!\right]^* = \left[\!\left[\mathrm{box}\ \mathrm{c}_j\right]\!\right] = [\![M]\!] = [\![\mathrm{box}\ \mathrm{c}_i]\!] = [\![\mathrm{c}_i]\!]^*$$

Post-composing with the counit (see [Kavvos 2017b, Prop. 4, §7.3.2]) we obtain $\left[\!\left[\mathrm{c}_j\right]\!\right] = [\![\mathrm{c}_i]\!]$, and hence $\mathrm{c}_i \equiv \mathrm{c}_j$. □

By Corollary 1, we know that $\Box : \mathbf{CSet}_{\mathcal{L}} \longrightarrow \mathbf{CSet}_{\mathcal{L}}$ is a product-preserving comonad, and thus a model of the Davies-Pfenning calculus. Moreover, we can interpret Bool_∇ by $\nabla\mathbb{B}$. This is a little harder to see: the reason is that the adjunction $U \dashv \nabla$ is a *reflection*, as ∇ is full and faithful. Thus, writing $\eta_A : A \to \nabla UA$ for the unit, we have a universal property;[4] it is that any $f : A \to \nabla X$ can be uniquely factorised through η_A:



Specialising this to $A = \Delta\mathbb{B} \cong \mathbf{1} + \mathbf{1}$, we have that any $f : \mathbf{1} + \mathbf{1} \to \nabla X$ can be uniquely factorised through $\nabla U\Delta\mathbb{B} \cong \nabla\mathbb{B}$:



This is exactly the elimination rule for Bool_∇: to define a function out of it one must tell what it does on the two constants (the components of $\mathbf{1} + \mathbf{1}$) by giving a term for each in a codiscrete type C, which is then of the right form ∇X, as $[\![C]\!] \cong \blacklozenge [\![C]\!] = \nabla(UC)$.

It is straightforward to show that the above interpretation is sound, and satisfies the assumptions of Lemma 2. We then have

Theorem 5 (Noninterference for Davies-Pfenning). *Let* $\cdot \mid x : \mathit{Bool}_\nabla \vdash M : \Box G$ *for any ground type* G. *Then, for any* $\vdash E, E' : \mathit{Bool}_\nabla$, *we have*

$$\vdash M[E/x] = M[E'/x] : \Box G$$

Proof. As $\mathbb{B}$ is non-empty, we have by Proposition 13(2) that

$$f \stackrel{\text{def}}{=} \left[\!\left[\cdot \mid x : \mathsf{Bool}_\nabla \vdash M : \Box G\right]\!\right] : \nabla\mathbb{B} \to \Box\,[\![G]\!]$$

is a constant function in $\mathbf{CSet}_{\mathcal{L}}$, so

$$[\![M[E/x]]\!] = f \circ [\![E]\!] = f \circ [\![E']\!] = [\![M[E'/x]]\!]$$

for any $\vdash E, E' : \mathsf{Bool}_\nabla$. By adequacy, it follows that $\vdash M[E/x] = M[E'/x] : \Box G$. □

5 LEVELLED COHESION

Whilst useful for discussing the properties of monadic and comonadic modalities, the results presented above are not particularly interesting in terms of information flow. For each classified set X, the set $\Box X$ is a version of it where everything is visible, whereas $\blacklozenge X$ is a version of it where nothing is. Might we be able to use the same ideas about pre-cohesion to make something similar, yet far more expressive? The key lies in relating pre-cohesion to the set of labels $\mathcal{L}$, of which we have not yet made any use.

[4] This is derived from the fact η_A is a universal arrow and ∇ is full and faithful.

Given a subset $\pi \subseteq \mathcal{L}$ of labels, there is a 'partially forgetful' functor,

$$U_\pi : \mathbf{CSet}_{\mathcal{L}} \longrightarrow \mathbf{CSet}_{\mathcal{L}-\pi}$$

which maps classified sets over $\mathcal{L}$ to classified sets over $\mathcal{L} - \pi$ by forgetting R_ℓ for $\ell \in \pi$. As before, there are two ways to define a set classified over $\mathcal{L}$ when given a set classified over $\mathcal{L} - \pi$.

The first one is given by the functor

$$\Delta_\pi : \mathbf{CSet}_{\mathcal{L}-\pi} \longrightarrow \mathbf{CSet}_{\mathcal{L}}$$

$\Delta_\pi X$ has the same carrier as X, and the same relations R_ℓ for $\ell \notin \pi$. But if $\ell \in \pi$, then R_ℓ of $\Delta_\pi X$ is the diagonal relation. We let Δ_π be the identity on morphisms: all the relations not in π are preserved by $\Delta_\pi f$, and the rest are diagonal so they are also trivially preserved. So for each classified set X over $\mathcal{L} - \pi$, we have a classified set $\Delta_\pi X$ over $\mathcal{L}$ which is *transparent at* π. It is easy to see that Δ_π is full and faithful, and that

Proposition 14. $\Delta_\pi \dashv U_\pi$

The second one is given by the functor

$$\nabla_\pi : \mathbf{CSet}_{\mathcal{L}-\pi} \longrightarrow \mathbf{CSet}_{\mathcal{L}}$$

which, this time, adds the complete relation as R_ℓ for each $\ell \in \pi$, and is also the identity on morphisms. Thus, for each classified set X over $\mathcal{L} - \pi$, we have a classified set $\nabla_\pi X$ over $\mathcal{L}$ which is *opaque at* π. It is also easy to see that it ∇_π is full and faithful, and that

Proposition 15. $U_\pi \dashv \nabla_\pi$

It remains to consider connected components. Suppose we have a morphism $f : X \to \Delta_\pi Y$. For each $\ell \in \pi$, the relation $\Delta_\pi Y$ is reflexivity. Hence, for $\ell \in \pi$,

$$x \; R_\ell \; x' \;\; (\text{in } X) \implies f(x) \; R_\ell \; f(x') \;\; (\text{in } \Delta_\pi Y) \implies f(x) = f(x')$$

Thus, if x and x' are indistinguishable at level $\ell \in \pi$, then f collapses them to a single element. As before, we would also like to phrase this in terms of quotients. Let $R^\star_\pi$ to be the reflexive, symmetric, transitive closure of $\bigcup_{\ell \in \pi} R_\ell$. The construction of the quotient set $|X| / R^\star_\pi$ extends to a functor

$$C_\pi : \mathbf{CSet}_{\mathcal{L}} \longrightarrow \mathbf{CSet}_{\mathcal{L}-\pi}$$

by letting $C_\pi(X)$ be the classified set with carrier $|X| / R^\star_\pi$, and, for $\ell \in \mathcal{L} - \pi$,

$$[b] \; R_\ell \; [b'] \;\; \text{in } C_\pi(X) \qquad \iff \qquad \exists x \in [b]. \; \exists y \in [b']. \; x \; R_\ell \; y \;\; \text{in } X$$

We let $C_\pi f \stackrel{\text{def}}{=} f^\star_\pi : |X| / R^\star_\pi \to |Y| / R^\star_\pi$, where

$$f^\star_\pi([x]) \stackrel{\text{def}}{=} [f(x)]$$

$f^\star_\pi$ is well-defined, as—by the same argument as in §3—f preserves $R^\star_\pi$. If $[b] \;\; R_\ell \;\; [b']$, there exist x and y such that $x \;\; R^\star_\pi \;\; b$, $y \;\; R^\star_\pi \;\; b'$, and $x \;\; R_\ell \;\; y$. But f preserves all of these relations, so $[f(b)] \; R_\ell \; [f(b')]$. So $f^\star_\pi$ is actually a morphism $C_\pi(X) \to C_\pi(Y)$. A similar argument to the one in §3 takes $f : X \to \Delta_\pi Y$ to a unique $\hat{f} : C_\pi X \to Y$, and hence

Proposition 16. $C_\pi : \mathbf{CSet}_{\mathcal{L}} \to \mathbf{CSet}_{\mathcal{L}-\pi}$ *is left adjoint to* $\Delta_\pi : \mathbf{CSet}_{\mathcal{L}-\pi} \to \mathbf{CSet}_{\mathcal{L}}$.

It is easy to see that this preserves products, and that the nullstellensatz holds as before (the natural isomorphisms showing $\Delta_\pi \dashv U_\pi \dashv \nabla_\pi$ are identities on the hom-sets). In total:

Theorem 6. $\mathbf{CSet}_{\mathcal{L}}$ *is pre-cohesive relative to* $\mathbf{CSet}_{\mathcal{L}-\pi}$.

It is worth reiterating some of the things that we automatically learn from this fact:

Corollary 2. *For a given $\pi \subseteq \mathcal{L}$,*

(1) $U_\pi : \mathbf{CSet}_{\mathcal{L}} \longrightarrow \mathbf{CSet}_{\mathcal{L}-\pi}$ *preserves (co)limits.*
(2) $\Delta_\pi : \mathbf{CSet}_{\mathcal{L}-\pi} \longrightarrow \mathbf{CSet}_{\mathcal{L}}$ *preserves (co)limits.*
(3) $\nabla_\pi : \mathbf{CSet}_{\mathcal{L}-\pi} \longrightarrow \mathbf{CSet}_{\mathcal{L}}$ *preserves limits.*
(4) $\Box_\pi \overset{def}{=} \Delta_\pi U_\pi : \boldsymbol{CSet}_{\mathcal{L}} \longrightarrow \boldsymbol{CSet}_{\mathcal{L}}$ *is an idempotent comonad that preserves finite (co)limits.*
(5) $\blacklozenge_\pi \overset{def}{=} \nabla_\pi U_\pi : \boldsymbol{CSet}_{\mathcal{L}} \longrightarrow \boldsymbol{CSet}_{\mathcal{L}}$ *is an idempotent monad that preserves finite limits.*
(6) $\int_\pi \overset{def}{=} \Delta_\pi C_\pi : \boldsymbol{CSet}_{\mathcal{L}} \longrightarrow \boldsymbol{CSet}_{\mathcal{L}}$ *is an idempotent monad. preserves products and colimits.*
(7) $\int_\pi \dashv \Box_\pi \dashv \blacklozenge_\pi$

As before, we can use this structure to prove abstract noninterference theorems for $\mathbf{CSet}_{\mathcal{L}}$, which will now be more expressive than before. Before we take on that task, however, we want to discuss two aspects of $\mathbf{CSet}_{\mathcal{L}}$. First, we want to identify the reflective and (co)reflective subcategories identified by $\Box_\pi$ and $\blacklozenge_\pi$. And, second, we want to prove once and or all that $\blacklozenge_\pi$ is a strong monad.

5.1 Reflective and Coreflective Subcategories

In cohesive settings we have a string of adjoints $\Delta \dashv U \dashv \nabla$, where Δ and ∇ are full and faithful. This is to say that both Δ and ∇ are a kind of 'inclusion,' and that—up to equivalence—they exhibit certain subcategories of $\mathcal{E}$. In particular, ∇ exhibits a *reflective subcategory* of objects Y such that $\blacklozenge Y \cong Y$, and Δ exhibits a *coreflective subcategory* of $\mathcal{E}$, viz. the full subcategory consisting of objects X such that $\Box X \cong X$. Moreover, such adjunctions generate *idempotent (co)monads*, in that their (co)multiplications $\Box \Rightarrow \Box^2$ and $\blacklozenge^2 \Rightarrow \blacklozenge$ are isomorphisms [Borceux 1994, Prop. 4.3.2].

It is illuminating to look at each of these cases for the pre-cohesion of $\mathbf{CSet}_{\mathcal{L}}$ relative to $\mathbf{CSet}_{\mathcal{L}-\pi}$.

Reflection. If X is a classified set such that $X \cong \blacklozenge_\pi X$, then R_ℓ is the complete relation at all $\ell \in \pi$. This leads us to the following definition:

Definition 5 (Protection). *Let $\pi \subseteq \mathcal{L}$ be a set of labels. A classified set X is* protected at π *just if $R_\ell = |X| \times |X|$ for all $\ell \in \pi$.*

Thus the monad $\blacklozenge_\pi$ induces a reflective subcategory $\mathbf{CSet}_{\mathcal{L},\mathrm{p},\pi}$ of *classified sets protected at π*. In other words, there is a functor $\mathcal{R}_\pi : \mathbf{CSet}_{\mathcal{L}} \longrightarrow \mathbf{CSet}_{\mathcal{L},\mathrm{p},\pi}$ that is left adjoint to the inclusion:

$$\mathbf{CSet}_{\mathcal{L},\mathrm{p},\pi} \underset{i}{\overset{\mathcal{R}_\pi}{\rightleftarrows}} \mathbf{CSet}_{\mathcal{L}} \quad (\perp)$$

$\mathcal{R}_\pi$ replaces R_ℓ with the complete relation for every $\ell \in \pi$. In addition, we have that $\blacklozenge_\pi = i \circ \mathcal{R}_\pi$. Finally, it is easy to see that $\blacklozenge_\pi$ is a *strictly idempotent* monad: the multiplication is the identity, as $X = \blacklozenge_\pi X$ whenever X is protected at π.

Interestingly,

Proposition 17. $\mathcal{R}_\pi : \boldsymbol{CSet}_{\mathcal{L}} \longrightarrow \boldsymbol{CSet}_{\mathcal{L},\mathrm{p},\pi}$ *preserves finite products.*

This is because $\mathbf{CSet}_{\mathcal{L},\mathrm{p},\pi} \cong \mathbf{CSet}_{\mathcal{L}-\pi}$, and up to that equivalence $\mathcal{R}_\pi$ is just U_π, and i is just ∇_π. By a theorem of category theory—see e.g. [Johnstone 2003, §A4.3.1]—if a reflector preserves finite products, as $\mathcal{R}_\pi$ does, then the reflective subcategory is an *exponential ideal*. That is,

Corollary 3. *If A is a classified set, and B is protected at π, then B^A is protected at π.*

This is what underlies the definition of *codiscrete types* we gave in §4.3, and which was introduced as a unmotivated definition in [Abadi et al. 1999].

Coreflection. Dually, the second subcategory consists of classified sets X over $\mathcal{L}$ such that $X \cong \Box_\pi X$. Each R_ℓ in X for $\ell \in \pi$ is the diagonal relation. The corresponding definition is:

Definition 6 (Visibility). *Let $\pi \subseteq \mathcal{L}$ be a set of labels. A classified set X is* visible at π *just if $R_\ell = \{ (x, x) \mid x \in |X| \}$ for every $\ell \in \pi$.*

This full coreflective subcategory induced $\Box_\pi$ is notated $\mathbf{CSet}_{\mathcal{L},\mathrm{v},\pi}$, and its objects are those *classified sets that are visible at π*. There is a functor

$$\mathcal{D}_\pi : \mathbf{CSet}_{\mathcal{L}} \longrightarrow \mathbf{CSet}_{\mathcal{L},\mathrm{v},\pi}$$

which returns the classified set $\mathcal{D}_\pi X$ with the same carrier, but if $\ell \in \pi$ then R_ℓ is the diagonal relation. It is also the identity on morphisms. This functor is right adjoint to the inclusion:

$$\mathbf{CSet}_{\mathcal{L}} \underset{D_\pi}{\overset{i}{\rightleftarrows}} \mathbf{CSet}_{\mathcal{L},\mathrm{v},\pi} \qquad (i \dashv D_\pi)$$

and of course $\Box_\pi = i \circ \mathcal{D}_\pi$. In a manner similar to previous one, $\Box_\pi$ is also a *strictly idempotent comonad*, as $X = \Box_\pi X$ whenever X is visible at π.

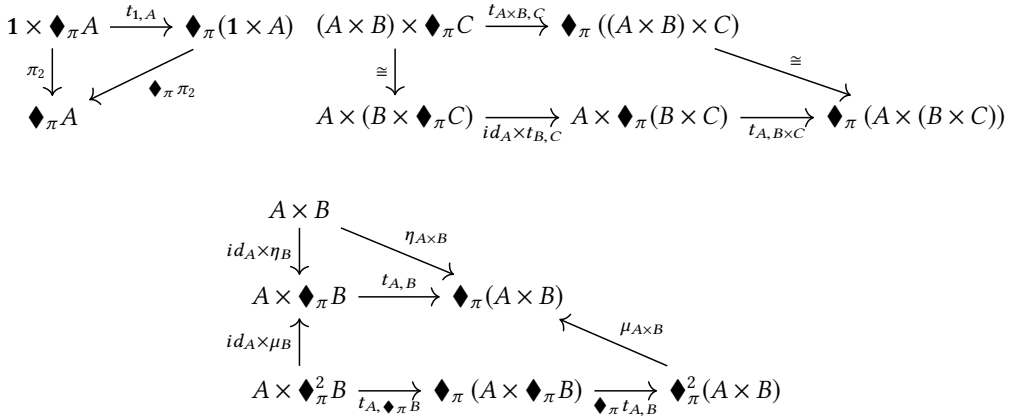
5.2 Redaction Is a Strong Monad

We record here a fact that we have already used and will use again, namely that

Proposition 18. *$\blacklozenge_\pi$ is a* strong monad*: that is, there exists a natural transformation*

$$t_{A,B} : A \times \blacklozenge_\pi B \to \blacklozenge_\pi (A \times B)$$

such that the following diagrams commute:



Proof. The components $t_{A,B}$ are identity functions on $|A \times \blacklozenge_\pi B| = |A| \times |B| = |\blacklozenge_\pi (A \times B)|$. These preserve all the relations at $\ell \in \pi$, as $\blacklozenge_\pi (A \times B)$ is protected at π. If $\ell \notin \pi$, then $(a, b)\ R_\ell\ (a', b')$ in $A \times \blacklozenge_\pi B$ means that $a\ R_\ell\ a'$ in A and $b\ R_\ell\ b'$ in B, so $(a, b)\ R_\ell\ (a', b')$ in $A \times B$ and hence in $\blacklozenge_\pi (A \times B)$. The diagrams commute as all the arrows excluding projections and associativities (but including components of t, η and μ and products thereof) are identities on carrier sets. □

5.3 Stacking Pre-Cohesions and Inter-Level Reasoning

In fact, it is not hard to generalise the results in the above section to the fact that we can 'stack' such pre-cohesions on top of one another. For example, by applying Theorem 6 twice, if $\pi' \subseteq \pi \subseteq \mathcal{L}$, we can obtain two pre-cohesions:

$$\begin{array}{ccccccc} & & \mathbf{CSet}_{\mathcal{L}} & & & & \\ & C_{\pi\subseteq\mathcal{L}} \big\downarrow & \dashv & \big\uparrow \Delta_{\pi\subseteq\mathcal{L}} & \dashv \big\downarrow U_{\pi\subseteq\mathcal{L}} & \dashv \big\uparrow \nabla_{\pi\subseteq\mathcal{L}} & \\ & & \mathbf{CSet}_{\pi} & & & & \\ & C_{\pi'\subseteq\pi} \big\downarrow & \dashv & \big\uparrow \Delta_{\pi'\subseteq\pi} & \dashv \big\downarrow U_{\pi'\subseteq\pi} & \dashv \big\uparrow \nabla_{\pi'\subseteq\pi} & \\ & & \mathbf{CSet}_{\pi'} & & & & \end{array}$$

which compose to what we previously denoted $C_{\mathcal{L}-\pi'} \dashv \Delta_{\mathcal{L}-\pi'} \dashv U_{\mathcal{L}-\pi'} \dashv \nabla_{\mathcal{L}-\pi'}$. For example,

$$U_{\mathcal{L}-\pi'} = U_{\pi'\subseteq\pi} \circ U_{\pi'\subseteq\mathcal{L}} : \mathbf{CSet}_{\mathcal{L}} \longrightarrow \mathbf{CSet}_{\pi'}$$
$$\nabla_{\mathcal{L}-\pi'} = \nabla_{\pi\subseteq\mathcal{L}} \circ \nabla_{\pi'\subseteq\pi} : \mathbf{CSet}_{\pi'} \longrightarrow \mathbf{CSet}_{\mathcal{L}}$$

and so forth. This forms a functor

$$\mathcal{P}(\mathcal{L})^{\mathrm{op}} \longrightarrow \mathbf{Precoh}$$

from the opposite powerset lattice of $\mathcal{L}$ to **Precoh**, whose morphisms are strings of adjoints $(C, \Delta, U, \nabla) : \mathcal{E} \to \mathcal{S}$ that exhibit $\mathcal{E}$ to be pre-cohesive over $\mathcal{S}$. The functor maps the unique arrow $\alpha : \pi \subseteq \pi'$ to a pre-cohesion $(C_\alpha, \Delta_\alpha, U_\alpha, \nabla_\alpha) : \mathbf{CSet}_{\pi'} \to \mathbf{CSet}_{\pi}$. Thus, each $\alpha : \pi \subseteq \pi'$ induces modalities $\Box_\alpha, \blacklozenge_\alpha : \mathbf{CSet}_{\pi'} \longrightarrow \mathbf{CSet}_{\pi}$, and we previously wrote $\Box_\pi$ for $\Box_{\alpha:\mathcal{L}-\pi\subseteq\mathcal{L}}$.

This functorial structure satisfies a number of strange-looking—yet very intuitive in terms of information flow—properties. To begin, we recall the isomorphism $U_\alpha\ \Delta_\alpha \cong U_\alpha\ \nabla_\alpha \cong \mathsf{Id}$, which was a consequence of pre-cohesion (Proposition 1), and holds *on-the-nose* in classified sets. This has the following consequences regarding the induced modalities.

Proposition 19. *If $\gamma : \pi'' \subseteq \pi'$ and $\alpha : \pi' \subseteq \pi$, then:*

(1) $\blacklozenge_{\alpha\circ\gamma}\ \Box_\alpha = \blacklozenge_{\alpha\circ\gamma} : \mathbf{CSet}_{\pi} \longrightarrow \mathbf{CSet}_{\pi}$
(2) $\Box_{\alpha\circ\gamma}\ \blacklozenge_\alpha = \Box_{\alpha\circ\gamma} : \mathbf{CSet}_{\pi} \longrightarrow \mathbf{CSet}_{\pi}$

Proof.

(1)

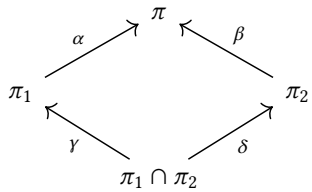
$$\begin{aligned} \blacklozenge_{\alpha\circ\gamma}\ \circ \Box_\alpha &= \nabla_{\alpha\circ\gamma}\ U_{\alpha\circ\gamma}\ \Delta_\alpha\ U_\alpha && \text{by definition} \\ &= \nabla_{\alpha\circ\gamma}\ U_\gamma\ U_\alpha\ \Delta_\alpha\ U_\alpha && \text{by functoriality and contravariance of } U \\ &= \nabla_{\alpha\circ\gamma}\ U_\gamma\ U_\alpha && \text{by the fundamental corollary} \\ &= \nabla_{\alpha\circ\gamma}\ U_{\alpha\circ\gamma} && \text{by functoriality and contravariance of } U \\ &= \blacklozenge_{\alpha\circ\gamma} && \text{by definition} \end{aligned}$$

(2) Similar to (1).

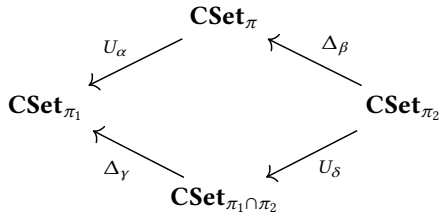
□

Intuitively, the first of these results says that declassifying things at some levels, and then redacting more than what was declassified is exactly the same as redacting all at once. Note that we used no particular properties about the model of classified sets, apart from the strictness of $U\Delta \cong U\nabla \cong \mathsf{Id}$.

Next, the forgetful functor U and the discretisation functor Δ sometimes commute. More specifically, if we have a pullback diagram in $\mathcal{P}(\mathcal{L})$



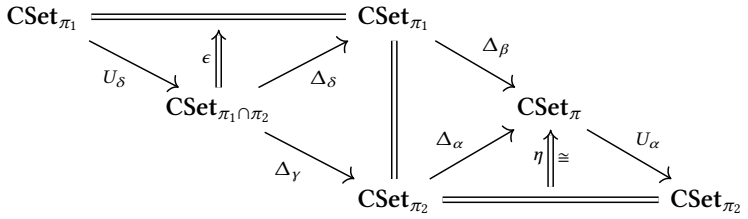
then it is easy to check that the following functor diagram commutes on-the-nose:



That is: if, starting from π_2, we equip the extra labels with discrete cohesion, and then forget everything down to π_1, we have not changed any of the labels of $\pi_1 \cap \pi_2$. Thus, we might first forget, and then discretise up to π_1. The same holds of codiscretisation, so we obtain

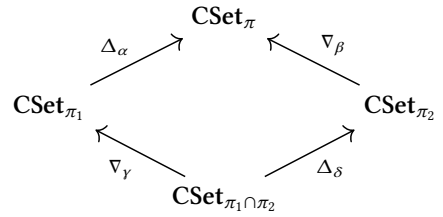
$$U_\alpha \nabla_\beta = \nabla_\gamma U_\delta$$

A sufficient condition for the above is that the components of the pasting diagram

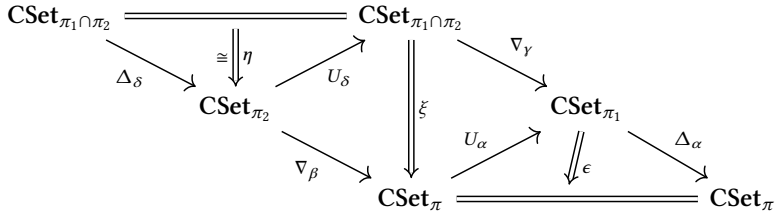


are identities—and they indeed are in our case: the components of η and ϵ are identity functions, so this reduces to simply checking $\Delta_\gamma\ U_\delta = U_\alpha\ \Delta_\beta$ again.

Secondly, if we discretise and codiscretise in disjoint regions, then these operations can be swapped. Namely, given a pullback diagram like above, we also have that

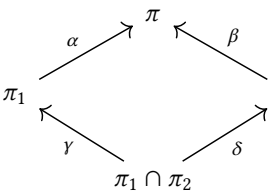


commutes. Again, it suffices that the following pasting diagram composes to an identity:



where ξ is a vertion of the preceding pasting diagram for ∇.

These 'basic laws' allow us to prove many more that relate the different cohesive structures.



Proposition 20. *Let* π_1 *be a pullback diagram. Then*

(1) $\Box_\gamma\, U_\alpha = U_\alpha\, \Box_\beta$
(2) $\blacklozenge_\gamma\, U_\alpha = U_\alpha\, \blacklozenge_\beta$
(3) $\Box_{\alpha\circ\gamma} = \Box_\alpha\, \Box_\beta$
(4) $\blacklozenge_{\alpha\circ\gamma} = \blacklozenge_\alpha\, \blacklozenge_\beta$
(5) $\Box_\alpha\, \Box_\beta = \Box_\beta\, \Box_\alpha$
(6) $\blacklozenge_\alpha\, \blacklozenge_\beta = \blacklozenge_\beta\, \blacklozenge_\alpha$

Proof.

(1)

$$\begin{aligned}
\Box_\gamma\, U_\alpha &= \Delta_\gamma\, U_\gamma\, U_\alpha && \text{by definition} \\
&= \Delta_\gamma\, U_\delta\, U_\beta && \text{by functoriality of } U \\
&= U_\alpha\, \Delta_\beta U_\beta && \text{by the first basic law above} \\
&= U_\alpha\, \Box_\beta && \text{by definition}
\end{aligned}$$

(2) Similar to (1), but with ∇.

(3)

$$\begin{aligned}
\Box_{\alpha\circ\gamma} &= \Delta_\alpha\, \Delta_\gamma\, U_\gamma\, U_\alpha && \text{by definition } (U \text{ contravariant}) \\
&= \Delta_\alpha\, \Box_\gamma\, U_\alpha && \text{by definition} \\
&= \Delta_\alpha\, U_\alpha\, \Box_\beta && \text{by (1)} \\
&= \Box_\alpha\, \Box_\beta && \text{by definition}
\end{aligned}$$

(4) Similar to (3), but with ∇.

(5) By applying (3) twice: $\Box_\alpha\, \Box_\beta = \Box_{\alpha\circ\gamma} = \Box_{\beta\circ\delta} = \Box_\beta\, \Box_\alpha$

(6) Similar to (5).

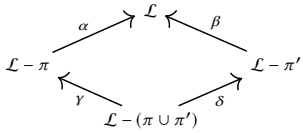
□

All this structure allows us to show things about the original modalities $\Box_\pi, \blacklozenge_\pi : \mathbf{CSet}_{\mathcal{L}} \longrightarrow \mathbf{CSet}_{\mathcal{L}}$.

Proposition 21.

(1) *If* $\pi \cap \pi' = \emptyset$, *then* $\Box_\pi \Box_{\pi'} = \Box_{\pi\cup\pi'}$.

(2) *If* $\pi \cap \pi' = \emptyset$, *then* $\blacklozenge_\pi \blacklozenge_{\pi'} = \blacklozenge_{\pi \cup \pi'}$.
(3) $\square_\pi \square_{\pi'} = \square_{\pi \cup \pi'}$
(4) $\blacklozenge_\pi \blacklozenge_{\pi'} = \blacklozenge_{\pi \cup \pi'}$
(5) *If* $\pi \subseteq \pi'$, *then* $\square_{\pi'} \blacklozenge_\pi = \square_{\pi'}$.
(6) *If* $\pi \subseteq \pi'$, *then* $\blacklozenge_{\pi'} \square_\pi = \blacklozenge_{\pi'}$.
(7) *If* $\pi \cap \pi' = \emptyset$, *then* $\square_\pi \blacklozenge_{\pi'} = \blacklozenge_{\pi'} \square_\pi$.
(8) $\square_\pi \blacklozenge_{\pi'} = \blacklozenge_{\pi' - \pi} \square_\pi$.
(9) $\blacklozenge_\pi \square_{\pi'} = \square_{\pi' - \pi} \blacklozenge_\pi$.

Proof. Notice that $\pi \cap \pi' = \emptyset$ implies that 

is a pullback diagram.

We can hence use Prop. 20: for (1), we have

$$\square_{\pi \cup \pi'} = \square_{\alpha \circ \gamma} = \square_\alpha \square_\beta = \square_\pi \square_{\pi'}$$

by definition and Prop. 20(3), and very similarly for (2). (3) follows by writing $\pi = \pi_1 \uplus (\pi \cap \pi')$ and $\pi' = \pi_2 \uplus (\pi \cap \pi')$ as disjoint unions, and then using (1) and strict idempotence to compute

$$\square_\pi \square_{\pi'} = \square_{\pi_1} \square_{\pi \cap \pi'} \square_{\pi \cap \pi'} \square_{\pi_2} = \square_{\pi_1} \square_{\pi \cap \pi'} \square_{\pi_2} = \square_{\pi_1 \cup (\pi \cap \pi') \cup \pi_2}$$

which is by definition equal to $\square_{\pi \cup \pi'}$. Again, a similar story for (4).

(5) and (6) follow from Prop. 19 (with $\gamma : \mathcal{L} - \pi' \subseteq \mathcal{L} - \pi$ and $\alpha : \mathcal{L} - \pi \subseteq \mathcal{L}$).

For (7), we calculate:

$$\begin{aligned}
\square_\pi \blacklozenge_{\pi'} &= \square_\alpha \blacklozenge_\beta && \text{by definition} \\
&= \Delta_\alpha U_\alpha \nabla_\beta U_\beta && \text{by definition} \\
&= \Delta_\alpha \nabla_\gamma U_\delta U_\beta && \text{by the first basic law} \\
&= \nabla_\beta \Delta_\delta U_\delta U_\beta && \text{by the second basic law} \\
&= \nabla_\beta \square_\delta U_\alpha && \text{by definition} \\
&= \nabla_\beta U_\beta \square_\alpha && \text{by Prop. 20(1)} \\
&= \blacklozenge_\beta \square_\alpha && \text{by definition}
\end{aligned}$$

Finally, (8) follows easily by writing $\pi = (\pi \cap \pi') \uplus (\pi - \pi')$ and using (4), (7) and (5); and similarly for (9). □

All of the above equational laws express very intuitive properties. For example, (3) says that if we declassify everything at security levels $\pi \cup \pi'$, we could have done that in two steps, with either π or π' first. (5) can be understood to mean that protecting everything at levels π and then declassifying everything at a larger set of levels π' is exactly the same as declassifying π' in one go. (7) allows us to switch redaction and declassification if they act on disjoint sets of labels. Finally, (8) and (9) show how to switch them even when there is overlap.

We have therefore developed an armoury of results about information flow. But notice that we have used no relational reasoning at all! We have only relied on a functor $C^{\mathrm{op}} \longrightarrow \mathbf{Precoh}$, and the three equations

$$U_\alpha \Delta_\beta = \Delta_\gamma U_\delta \tag{1}$$
$$U_\alpha \nabla_\beta = \nabla_\gamma U_\delta \tag{2}$$
$$\Delta_\alpha \nabla_\gamma = \nabla_\beta \Delta_\delta \tag{3}$$

for each pullback diagram $\begin{array}{ccc} \cdot & \xrightarrow{\gamma} & \cdot \\ {\scriptstyle\delta}\downarrow & & \downarrow{\scriptstyle\alpha} \\ \cdot & \xrightarrow[\beta]{} & \cdot \end{array}$ in C. It is conceivable that these equations could have a more general standing, especially if we replace equality with natural isomorphism. All the results in this section would then hold, but only up to natural isomorphism.

6 NONINTERFERENCE II: MULTI-MODAL INFORMATION FLOW

The theory developed in the previous section enables us to model *multi-modal* information flow calculi, which feature modalities that are *indexed* in some way. The usual way to do so is to index them over a poset $(\mathcal{L}, \sqsubseteq)$ of security levels, and which is often (but not always) a lattice. We will focus on two main examples: the *dependency core calculus* (DCC) of Abadi et al. [1999], and the *sealing calculus* of Shikuma and Igarashi [2008].

We have now moved on to levelled cohesion over an arbitrary set $\mathcal{L}$ of labels. The first thing we want to note is that codiscrete contractibility is still satisfied, as long as $\pi \neq \emptyset$: if X is non-empty, we redact it at some levels π, and then take the 'view from π,' we end up with almost nothing, namely a single connected component, i.e. $C_\pi(\nabla_\pi X) \cong \mathbf{1}$ For that reason, a levelled version of Prop. 13 from §4 holds:

Proposition 22. *If $\pi \neq \emptyset$ and A is non-empty, then morphisms $\blacklozenge_\pi A \to \Delta_\pi B$ naturally correspond to points $\mathbf{1} \to B$, and are hence constant functions in $\mathbf{CSet}_\mathcal{L}$.*

The proof is the same as before.

6.1 Dependency Core Calculus

The *dependency core calculus* of Abadi et al. [1999] is at its core a version of Moggi's computational metalanguage with multiple monads T_ℓ indexed over $\ell \in \mathcal{L}$, where $(\mathcal{L}, \sqsubseteq)$ is a *lattice*, also called an *information flow lattice*.[5] The introduction rule is exactly that of Moggi:

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash [M]_\ell : T_\ell A}$$

The elimination rule is modified slightly:

$$\frac{\Gamma \vdash M : T_\ell A \quad \Gamma, x : A \vdash N : B \quad \text{B is protected at } \ell}{\Gamma \vdash \mathsf{let}\ x = M\ \mathsf{in}\ N : B}$$

Following Abadi et al. [1999], we say that the type B is protected at ℓ whenever: (I) if $\ell' \sqsubseteq \ell$ then $T_\ell(A)$ is protected at ℓ'; (II) if A is protected at ℓ, then so is $T_{\ell'}(A)$ for any ℓ'; and (III) if A, B are protected at ℓ, then so are $A \times B$ and $C \to A$ for any type C.

To interpret the DCC[6] all we need is *strong, strictly idempotent monad* T_ℓ on a CCC C for each $\ell \in \mathcal{L}$, such that if B is a protected type, then $[\![B]\!] = T_\ell\, [\![B]\!]$ strictly, i.e. $[\![B]\!]$ is in the reflective subcategory induced by T_ℓ. The elimination rule then reduces to Moggi's interpretation, and we can straightforwardly adapt the soundness proof, as well as the canonicity proof for Bool.

It is now easy to see that the levelled structure on $\mathbf{CSet}_\mathcal{L}$ from §5 is a model of the DCC, with

$$T_\ell \overset{\text{def}}{=} \blacklozenge_{\downarrow \ell}$$

[5] It is worth noting that, curiously, the meet and join operations are never used in the study of the DCC.
[6] More specifically: a version of the DCC without fixpoints, which the original included.

where $\downarrow \ell \stackrel{\text{def}}{=} \{ \ell' \in \mathcal{L} \mid \ell' \sqsubseteq \ell \}$ is the *principal lower set* of ℓ. It is not hard to show that if B is protected at ℓ (as a type) then $[\![B]\!]$ is protected at $\downarrow \ell$ (as a classified set). As shown in §5.1–5.2, $\blacklozenge_\pi$ is a strong, strictly idempotent monad, so that $[\![B]\!] = \blacklozenge_{\downarrow \ell} [\![B]\!]$.

There is no explicit noninterference theorem stated for DCC itself in [Abadi et al. 1999]. Instead, there are six translations from various calculi into DCC, which are used to prove noninterference for each of these 'source' calculi. The technique is always the same: first, show that the translation $(-)^\dagger$ is *adequate*, in the sense that E has a canonical form in the source calculus if and only if the semantics of the translation satisfies $[\![E^\dagger]\!] \neq \bot$. Then, some argument similar to our Prop. 22 is used to show constancy of the 'DCC-induced' semantics $[\![E^\dagger]\!]$ of a term $E : \mathsf{Bool}$ with a single free variable of an appropriately 'secure' type. Then $[\![(E[M/x])^\dagger]\!] = [\![(E[M'/x])^\dagger]\!]$ for all M, M', and a single use of adequacy suffices to complete the argument.

We will attempt to capture the essence common to these proofs by the following proposition.

Proposition 23. *If $\pi - \pi' \neq \emptyset$, A is non-empty, and B is visible at $\pi - \pi'$, then all morphisms*

$$\blacklozenge_\pi A \to \blacklozenge_{\pi'} B$$

are constant functions.

Proof. If B is visible at $\pi - \pi'$ then $B = \square_{\pi - \pi'} B$, so

$$\blacklozenge_{\pi'} B = \blacklozenge_{\pi'} \square_{\pi-\pi'} B = \square_{(\pi-\pi')-\pi'} \blacklozenge_{\pi'} B = \square_{\pi-\pi'} \blacklozenge_{\pi'} B = \Delta_{\pi-\pi'} U_{\pi-\pi'} \blacklozenge_{\pi'} B$$

by Prop. 21(9), set theory, and the definition of $\square_\pi$. But also

$$\blacklozenge_\pi A = \blacklozenge_{\pi-\pi'} \blacklozenge_{\pi \cap \pi'} A$$

by Prop. 21(4) and set theory. As $\pi - \pi' \neq \emptyset$ and A is non-empty, Prop. 22 applies. □

We also have the following analogue of Lemma 1.

Lemma 3 (DCC Adequacy). *Suppose we have a sound categorical interpretation for the DCC, as described before. Suppose that, as per the definition in [Moggi 1991], each monad T_ℓ satisfies the* mono requirement*, i.e. each component $A \to T_\ell A$ of the unit is mono. Let G be a ground type that satisfies canonicity, with an injective interpretation, so that*

$$[\![\vdash c_i : G]\!] = [\![\vdash c_j : G]\!] \quad \Longrightarrow \quad c_i \equiv c_j$$

Then this interpretation is adequate *for $T_\ell G$, in the sense that*

$$[\![\vdash M : T_\ell G]\!] = [\![\vdash [c_i]_\ell : T_\ell G]\!] \quad \Longrightarrow \quad \vdash M = [c_i]_\ell : T_\ell G$$

Proof. The canonical forms at $T_\ell G$ are exactly $[\mathsf{c}_i]_\ell$. Let M normalise to $[\mathsf{c}_j]_\ell$. Then

$$\eta_G \circ [\![\mathsf{c}_j]\!] = [\![[\mathsf{c}_j]_\ell]\!] = [\![M]\!] = [\![[\mathsf{c}_i]_\ell]\!] = \eta_G \circ [\![\mathsf{c}_i]\!]$$

As η_G is mono, $[\![\mathsf{c}_j]\!] = [\![\mathsf{c}_i]\!]$, and hence $\mathsf{c}_j \equiv \mathsf{c}_i$. □

We can now interpret DCC+booleans into $\mathsf{CSet}_{\mathcal{L}}$, with $[\![\mathsf{Bool}]\!] \stackrel{\text{def}}{=} \Delta\mathbb{B} \cong \mathbf{1} + \mathbf{1}$. This interpretation satisfies all the requirements of Lemma 3—as every component $A \to \blacklozenge_\pi A$ is a mono—so it is adequate.

Theorem 7 (Noninterference for DCC). *Let A be a non-empty type, and let $x : T_\ell A \vdash M : T_{\ell'} \mathsf{Bool}$ with $\ell \not\sqsubseteq \ell'$. Then, for any $\vdash E, E' : T_\ell$, we have*

$$\vdash M[E/x] = M[E'/x] : T_{\ell'} \mathsf{Bool}$$

Proof. We have that

$$[\![x : T_\ell A \vdash M : T_{\ell'}\mathrm{Bool}]\!] : \blacklozenge_{\downarrow\ell}[\![A]\!] \to \blacklozenge_{\downarrow\ell'}\Delta\mathbb{B}$$

But $\ell \not\sqsubseteq \ell'$ if and only if $\downarrow \ell \not\subseteq \downarrow \ell'$, so $\downarrow \ell - \downarrow \ell' \neq \emptyset$. As $\Delta\mathbb{B}$ is visible everywhere, it follows by Proposition 23 that $[\![M]\!] : \blacklozenge_{\downarrow\ell}[\![A]\!] \to \blacklozenge_{\downarrow\ell'}\Delta\mathbb{B}$ is a constant function, so

$$[\![M[E/x]]\!] = [\![M]\!] \circ [\![E]\!] = [\![M]\!] \circ [\![E']\!] = [\![M[E'/x]]\!]$$

for any $\vdash E, E' : T_\ell A$. By adequacy, it follows that $\vdash M[E/x] = M[E'/x] : T_{\ell'}\mathrm{Bool}$. □

6.2 The Sealing Calculus

The *sealing calculus* was introduced by Shikuma and Igarashi [2008]. Its history is complicated: it is a simplification of a calculus introduced by Tse and Zdancewic [2004] as a refinement of DCC. The authors originally hoped to prove noninterference for DCC not through denotational methods—as in [Abadi et al. 1999]—but by translating it to System F and using parametricity. However, there was a technical issue in their work. Shikuma and Igarashi [2008] carried out a similar programme by translating their sealing calculus to simple types, thus proving noninterference through parametricity for simple types. Subsequently, Bowman and Ahmed [2015] carried out the original programme to completion, by translating DCC itself to System F_ω.

The sealing calculus is also based on a partial order $(\mathcal{L}, \sqsubseteq)$ of security levels. It augments the context of the simply typed λ-calculus with a finite set π of *observer levels*. Typing judgements are of the form

$$\Gamma \mid \pi \vdash M : A$$

The idea is that an observer can only read data the classification of which is below their observer status. We write $\ell \sqsubseteq \pi$ to mean that ℓ is below some level in π.

The introduction rule specifies that a term obtained using observer access ℓ can be *sealed*, thus becoming a term that can be handled *possibly without* (but not necessarily without) access ℓ:

$$\frac{\Gamma \mid \pi \cup \{\ell\} \vdash M : A}{\Gamma \mid \pi \vdash [M]_\ell : [A]_\ell}$$

$[A]_\ell$ is a *type sealed at* ℓ. Conversely, if the observer level dominates ℓ, terms can be *unsealed*:

$$\frac{\Gamma \mid \pi \vdash M : [A]_\ell \quad \ell \sqsubseteq \pi}{\Gamma \mid \pi \vdash M^\ell : A}$$

and, naturally, $([M]_\ell)^\ell = M$. The rest of the system is just that of simple types.

Our levelled cohesion can be used quite directly to provide semantics for the sealing calculus. To do so, notice that the only rules that interact with the observer context π are the modal/sealing rules: if we forget those for a moment, everything else is simply-typed λ-calculus. Thus, we can interpret the sealing-free/'constant π' part of the calculus in any cartesian closed category. We will choose to do so in the *co-Kleisli category* $\mathrm{CoKl}(\square_{\downarrow\pi})$ of $\square_{\downarrow\pi}$, which has the same objects as $\mathbf{CSet}_\mathcal{L}$, but whose morphisms $A \to B$ are the morphisms $\square_{\downarrow\pi}A \to B$ of $\mathbf{CSet}_\mathcal{L}$. This is standard, see e.g. [Awodey 2010, Ex. 11, §10.6]. Also standard is the following theorem, which is considered 'folk' by Brookes and Geva [1992], and mentioned in passing by Uustalu and Vene [2008]:

Theorem 8. *If $Q : C \longrightarrow C$ is a product-preserving comonad on a cartesian closed category C, then its co-Kleisli category $\mathrm{CoKl}(Q)$ is also cartesian closed.*

Thus, a sequent $x_1 : A_1, \ldots, x_n : A_n \mid \pi \vdash M : A$ is interpreted as an arrow

$$\square_{\downarrow\pi}([\![A_1]\!] \times \cdots \times [\![A_n]\!]) \to [\![A]\!]$$

The idea is that the observer context π declassifies everything at levels below some level in π. But recall that $\Box_\pi$ is product-preserving, and that in the particular example of $\mathbf{CSet}_{\mathcal{L}}$ it is *strictly* product preserving. Thus, morphisms of the above type are of the form

$$\Box_{\downarrow\pi} [\![A_1]\!] \times \cdots \times \Box_{\downarrow\pi} [\![A_n]\!] \to [\![A]\!]$$

Sealing is uniformly interpreted on types by the redaction functors:

$$[\![[A]_\ell]\!] \stackrel{\text{def}}{=} \blacklozenge_{\downarrow\ell}\ [\![A]\!]$$

On terms, the sealing and unsealing rules will be interpreted by using the adjunction $\Box_{\downarrow\pi} \dashv \blacklozenge_{\downarrow\pi}$ to move between the different co-Kleisli categories of the comonads $\Box_{\downarrow\pi}$. For simplicity we explain the case $\ell \notin \pi$ which discards ℓ from the observer levels. Recall that, by Prop. 21, we have

$$\Box_{\downarrow(\pi\cup\{\ell\})} = \Box_{\downarrow\pi\cup\downarrow\ell} = \Box_{\downarrow\pi}\ \Box_{\downarrow\ell} = \Box_{\downarrow\ell}\ \Box_{\downarrow\pi}$$

So the interpretation of a term $\Gamma \mid \pi \cup \{\ell\} \vdash M : A$, which is a morphism $[\![\Gamma]\!] \to [\![A]\!]$ in the co-Kleisli category of $\Box_{\downarrow\pi\cup\{\ell\}}$, is, by the above, really a morphism of type

$$\Box_{\downarrow\ell} \left(\Box_{\downarrow\pi}\ [\![A_1]\!] \times \cdots \times \Box_{\downarrow\pi}\ [\![A_n]\!]\right) \to [\![A]\!]$$

in $\mathbf{CSet}_{\mathcal{L}}$. Moving across the adjunction yields a morphism

$$\Box_{\downarrow\pi}\ [\![A_1]\!] \times \cdots \times \Box_{\downarrow\pi}\ [\![A_n]\!] \to \blacklozenge_{\downarrow\ell}\, [\![A]\!]$$

which is now a morphism in the co-Kleisli category of $\Box_\pi$, and we take that to be the interpretation of $\Gamma \mid \pi \vdash M : [A]_\ell$. Unsealing is obtained by moving in the opposite direction: doing so, we obtain a morphism

$$\Box_{\downarrow\ell}\ \Box_{\downarrow\pi}\ [\![A_1]\!] \times \cdots \times \Box_{\downarrow\ell}\ \Box_{\downarrow\pi}\ [\![A_n]\!] \to [\![A]\!]$$

But $\ell \sqsubseteq \pi$ implies $\downarrow \pi = \downarrow \pi \cup \downarrow \ell$, and so $\Box_{\downarrow\ell}\ \Box_{\downarrow\pi}$ is just $\Box_{\downarrow\pi}$. Again, it would suffice for these equalities of modalities to be mere natural isomorphisms. Soundness of the equations of the sealing calculus then follow from the fact the adjunction induces a natural isomorphism between the appropriate hom-sets.

Shikuma and Igarashi [2008] prove the following noninterference theorem. Say that terms $\cdot \mid \pi \vdash M_1, M_2 : A$ at observer level π are *contextually equivalent* just if there is no term of ground type at the same observer level that can distinguish them: that is, if $x : A \mid \pi \vdash N : \mathrm{Bool}$, then $N[M_1/x] = N[M_2/x]$. This defines an equivalence relation $\approx_\pi$, which extends to substitutions $\cdot \mid \pi \vdash \sigma : \Gamma$. The main theorem states that if $\sigma \approx_\pi \sigma' : \Gamma$ and $\Gamma \mid \pi \vdash E : A$, then $E[\sigma] \approx_\pi E[\sigma']$. That is: substituting terms indistinguishable at π yields results indistinguishable at π.

This is a particularly strong noninterference theorem, which takes many pages of very beautiful—but also painfully elaborate!—work to show. We will content ourselves with using the model to provide the following direct corollary: [7]

Theorem 9 (Noninterference for Sealing Calculus). *Let A be a non-empty type. If $\ell \not\sqsubseteq \pi$, then for any $\cdot \mid \pi \vdash M, N : [A]_\ell$ and $x : [A]_\ell \mid \pi \vdash E : \mathit{Bool}$, we have $\cdot \mid \pi \vdash E[M/x] = E[N/x] : \mathit{Bool}$.*

First, we notice that Shikuma and Igarashi [2008] show confluence and strong normalisation for the sealing calculus (including unit and coproducts, which subsume Bool). Thus canonicity holds, and Lemma 1 applies to yield adequacy at Bool. The domain of $[\![x : [A]_\ell \mid \pi \vdash E : \mathrm{Bool}]\!]$ is $\blacklozenge_{\downarrow\ell} [\![A]\!]$ as an object of $\mathrm{CoKl}(\Box_{\downarrow\pi})$, and hence as an object of $\mathbf{CSet}_{\mathcal{L}}$ it is

$$\Box_{\downarrow\pi}\ \blacklozenge_{\downarrow\ell}\ [\![A]\!] = \blacklozenge_{\downarrow\ell-\downarrow\pi}\ \Box_{\downarrow\pi}\ [\![A]\!]$$

[7] It is easy to show that any $M, N : [A]_\ell$ are contextually equivalent at $\pi \not\sqsupseteq \ell$ by a logical relations argument: see [Shikuma and Igarashi 2008, Theorem 2.18]. Then this result is the special case of the noninterference theorem, once we observe that equality and observational equivalence coincide at the ground type Bool.

by Prop. 21(8). Its codomain is $\Delta\mathbb{B}$, which is visible everywhere. But, as $\ell \not\sqsubseteq \pi$, we have that $\downarrow \ell - \downarrow \pi \neq \emptyset$, so Prop. 23 applies (with $\pi' = \emptyset$) to show that it is a constant function. Then,

$$[\![E[M/x]]\!] = [\![E]\!] \circ_{\mathrm{CoKl}} [\![M]\!] = [\![E]\!] \circ_{\mathrm{CoKl}} [\![N]\!] = [\![E[N/x]]\!]$$

where $\circ_{\mathrm{CoKl}}$ is composition $\mathrm{CoKl}(\square_{\downarrow\pi})$, and by using adequacy the proof is complete.

7 CONCLUSION

To recapitulate: we have defined the model of classified sets, and shown that it forms a pre-cohesion. This led us to the generation of modalities $\int \dashv \square \dashv \blacklozenge$, and the proof of noninterference properties for Moggi's monadic metalanguage and the Davies-Pfenning comonadic calculus. Next, we took a levelled view of cohesion, and showed that this generates a multi-modal framework $\int_\pi \dashv \square_\pi \dashv \blacklozenge_\pi$. These modalities satisfy many algebraic laws, which we then used to prove noninterference for two multi-modal information flow calculi, the dependency core calculus, and the sealing calculus.

We discuss two aspects of our work that we believe may lead to interesting future developments.

Cohesion as a theory of information flow. Our results demonstrate that the very general and abstract framework of pre-cohesion in fact has very concrete applications in analysing information flow. Our noninterference proofs rely on very general lemmas about pre-cohesion—in the case of §6, with some additional, slightly mysterious equations—which are then applied to each calculus by using some usually very simple form of adequacy. We believe these to be a simplification compared to previous work on noninterference, which required quite a bit of hard work in terms of fully abstract/fully complete translations, e.g. [Bowman and Ahmed 2015; Shikuma and Igarashi 2008].

However, we believe this to be the tip of the iceberg: cohesion can tell us much more about information flow. However, this cannot happen unless we replace adequacy, which is a very weak form of completeness, with something stronger. It would be very interesting to see whether there are information flow calculi for which classified sets are *fully abstract*, in the sense that M and N are contextually equivalent if and only if $[\![M]\!] = [\![N]\!]$ in $\mathbf{CSet}_{\mathcal{L}}$. That would grant us far more power to use ideas from and properties of cohesion to prove theorems about information flow.

Conversely, we can seek information flow calculi with likeness to cohesion-based models: we can look at what structure is available in the multi-modal setting of classified sets, and then try to formulate a calculus from that. This would most likely lead to a multi-modal version of the *spatial type theory* of Shulman [2018]. Such a calculus would also have much to offer in terms of resolving the debate between, for example, coarse-grained and fine-grained formulations, as discussed by Rajani and Garg [2018]. In a sense, its formulation would provide a mathematical justification for *canonical*, type-theoretic choices of modalities and constructs. (That is, if we of course accept classified sets as a canonical model of information flow.) Additionally, the calculus would include the *shape at* π modality ($\int_\pi$), which has never appeared before in papers on information flow type systems, and which might have interesting applications as a 'security quotient' or 'secure view' type constructor. Finally, there seem to be close connections between this work and *graded monads and comonads*, which also have applications in information flow: see [Gaboardi et al. 2016].

Cohesion as a basis for multi-modal type theories. The other side of the coin in the present development is that information flow—which is a garden-variety application for multi-modal types—can be quite eloquently spoken about in this language of cohesion. In particular, the formulation of a functor $\mathcal{P}(\mathcal{L})^{\mathrm{op}} \longrightarrow \mathbf{Precoh}$ enabled very short and conceptual proofs of results that would ordinarily require a lot of uninteresting relational reasoning. It is thus natural to ask whether there might be other functors $\mathcal{C}^{\mathrm{op}} \longrightarrow \mathbf{Precoh}$ of ***Precoh****-valued presheaves* that yield interesting analyses of multi-modal logical systems. In fact, some of the notation we developed in §5 bears a

striking similarity to the adjoint logic of Licata and Shulman [2016], and the framework of Licata et al. [2017]. Could there be a closer connection between these developments?

The combination of the parametricity-style reasoning and cohesion has also recently appeared in the multi-modal type theories of Nuyts et al. [2017] and Nuyts and Devriese [2018]. There is certainly potential for a very interesting connection to be made there, e.g. by devising a dependent type theory for information flow, for which the noninterference theorems can be proven *internally*.

Related work. As mentioned in the introduction, the problem of information flow is almost as old as Computer Science itself, dating at least as far back as Bell and LaPadula's report of 1973 [LaPadula and Bell 1996]. The notion of *noninterference* itself was introduced by Goguen and Meseguer [1982]. The use of types to guarantee information flow control, and hence some form of noninterference, appears to have begun in the 1990s, with the first work on higher-order functional programming being that of Heintze and Riecke [1998] on the SLam calculus; see *op. cit.* for a useful list of references to approaches that preceded it. Rajani and Garg [2018] provide a good overview and a largely complete list of references to the literature thereafter.

Even the first higher-order noninterference results, such as those for SLam [Heintze and Riecke 1998], use some form of logical relations, who directly state that they are "borrowing ideas from Reynolds." In particular, Heintze and Riecke [1998] use logical relations on top of a denotational model, which then led to the *dependency category* of Abadi et al. [1999], which we have refined into classified sets. Other than that, there appears to be very little other work on denotational models of information flow; even if situated outside the higher-order functional setting, we ought to mention the work of Sabelfeld and Sands [2001].

Furthermore, we should note that this paper is the first *categorical* approach to information flow. Even though we mostly use the motivating example of classified sets, all our theorems are rather general, and apply to all pre-cohesive settings indexed over subsets of $\mathcal{L}$ for which the equations 1, 2 and 3 apply. It is also worth noting that we did not use any particular structure on $\mathcal{L}$, even though most of the related work cited above asks for some kind of lattice—even if they do not use it either: we merely made use of indexing over $\mathcal{P}(\mathcal{L})$.

ACKNOWLEDGMENTS

I would like to thank Dan Licata for numerous observations that led to the material in this paper. The pasting diagrams in §5.3 are due to Amar Hadzihasanovic. Thanks are due to Mario Alvarez-Picallo, Mike Shulman, and the anonymous reviewers for their many useful suggestions, corrections, and careful reading. Finally, I would also like to thank Dan Licata's cat, Otto, for keeping me company during the writing of this paper.

This material is based upon work supported by the Air Force Office of Scientific Research under award number FA9550-16-1-0292. Any opinions, finding, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Air Force.

REFERENCES

Martín Abadi, Anindya Banerjee, Nevin Heintze, and Jon G Riecke. 1999. A core calculus of dependency. In *Proceedings of the 26th ACM SIGPLAN-SIGACT symposium on Principles of programming languages - POPL '99*. ACM Press, New York, New York, USA, 147–160. https://doi.org/10.1145/292540.292555

Samson Abramsky and Nikos Tzevelekos. 2011. Introduction to Categories and Categorical Logic. In *New Structures for Physics*, Bob Coecke (Ed.). Springer-Verlag, 3–94. https://doi.org/10.1007/978-3-642-12821-9_1 arXiv:1102.1313

Steve Awodey. 2010. *Category Theory*. Oxford University Press.

Nick Benton, Gavin M. Bierman, and Valeria de Paiva. 1998. Computational types from a logical perspective. *Journal of Functional Programming* 8, 2 (1998), 177–193. https://doi.org/10.1017/S0956796898002998

Francis Borceux. 1994. *Handbook of Categorical Algebra.* Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511525865

William J. Bowman and Amal Ahmed. 2015. Noninterference for free. In *Proceedings of the 20th ACM SIGPLAN International Conference on Functional Programming - ICFP 2015.* ACM Press, New York, New York, USA, 101–113. https://doi.org/10.1145/2784731.2784733

Stephen Brookes and Shai Geva. 1992. Computational comonads and intensional semantics. In *Applications of Categories in Computer Science*, M. P. Fourman, Peter T Johnstone, and Andrew M Pitts (Eds.). Vol. 177. Cambridge University Press, Cambridge, 1–44. https://doi.org/10.1017/CBO9780511525902.003

Ranald Clouston, Aleš Bizjak, Hans Bugge Grathwohl, and Lars Birkedal. 2016. The guarded lambda calculus: Programming and reasoning with guarded recursion for coinductive types. *Logical Methods in Computer Science* 12, 3 (2016), 1–39. https://doi.org/10.2168/LMCS-12(3:7)2016

Roy L. Crole. 1993. *Categories for Types.* Cambridge University Press.

Pierre-Louis Curien, Marcelo Fiore, and Guillaume Munch-Maccagnoni. 2016. A theory of effects and resources: adjunction models and polarised calculi. In *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages - POPL 2016.* ACM Press, New York, New York, USA, 44–56. https://doi.org/10.1145/2837614.2837652

Rowan Davies and Frank Pfenning. 1996. A modal analysis of staged computation. In *Proceedings of the 23rd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL'96).* 258–270. https://doi.org/10.1145/382780.382785

Rowan Davies and Frank Pfenning. 2001. A modal analysis of staged computation. *J. ACM* 48, 3 (2001), 555–604. https://doi.org/10.1145/382780.382785

Dorothy E Denning. 1976. A lattice model of secure information flow. *Commun. ACM* 19, 5 (1976), 236–243. https://doi.org/10.1145/360051.360056

Marco Gaboardi, Shin-ya Katsumata, Dominic Orchard, Flavien Breuvart, and Tarmo Uustalu. 2016. Combining effects and coeffects via grading. In *Proceedings of the 21st ACM SIGPLAN International Conference on Functional Programming - ICFP 2016.* ACM Press, New York, New York, USA, 476–489. https://doi.org/10.1145/2951913.2951939

J. A. Goguen and J. Meseguer. 1982. Security Policies and Security Models. In *1982 IEEE Symposium on Security and Privacy.* IEEE, 11–11. https://doi.org/10.1109/SP.1982.10014

Nevin Heintze and Jon G Riecke. 1998. The SLam calculus: programming with secrecy and integrity. In *Proceedings of the 25th ACM SIGPLAN-SIGACT symposium on Principles of programming languages - POPL '98.* ACM Press, New York, New York, USA, 365–377. https://doi.org/10.1145/268946.268976

Claudio Hermida, Uday S. Reddy, and Edmund P. Robinson. 2014. Logical relations and parametricity - A Reynolds Programme for category theory and programming languages. *Electronic Notes in Theoretical Computer Science* 303 (2014), 149–180. https://doi.org/10.1016/j.entcs.2014.02.008

Martin Hofmann. 1999. *Type Systems for Polynomial-Time Computation.* Habilitation thesis. Technischen Universität Darmstadt. http://www.lfcs.inf.ed.ac.uk/reports/99/ECS-LFCS-99-406/

Peter T. Johnstone. 2003. *Sketches of an Elephant: A Topos Theory Compendium.* Clarendon Press.

G. A. Kavvos. 2017a. Dual-context calculi for modal logic. In *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS).* IEEE. https://doi.org/10.1109/LICS.2017.8005089

G. A. Kavvos. 2017b. *Dual-context calculi for modal logic (technical report).* Technical Report. University of Oxford. http://www.lambdabetaeta.eu/papers/dualcalc.pdf

Neelakantan R. Krishnaswami. 2013. Higher-order functional reactive programming without spacetime leaks. In *Proceedings of the 18th ACM SIGPLAN international conference on Functional programming - ICFP '13.* ACM, ACM Press, New York, New York, USA, 221. https://doi.org/10.1145/2500365.2500588

Leonard J. LaPadula and D. Elliott Bell. 1996. Secure Computer Systems: Mathematical Foundations. *Journal of Computer Security* 4, 2-3 (1996), 239–263. https://doi.org/10.3233/JCS-1996-42-308

F. William Lawvere. 2007. Axiomatic cohesion. *Theory and Applications of Categories* 19, 3 (2007), 41–49. http://www.tac.mta.ca/tac/volumes/19/3/19-03.pdf

F. William Lawvere and M. Menni. 2015. Internal choice holds in the discrete part of any cohesive topos satisfying stable connected codiscreteness. *Theory and Applications of Categories* 30, 26 (2015), 909–932. http://www.tac.mta.ca/tac/volumes/30/26/30-26.pdf

Daniel R. Licata and Michael Shulman. 2016. Adjoint Logic with a 2-Category of Modes. In *Proceedings of LFCS 2016.* 219–235. https://doi.org/10.1007/978-3-319-27683-0_16

Daniel R. Licata, Michael Shulman, and Mitchell Riley. 2017. A Fibrational Framework for Substructural and Modal Logics. In *2nd International Conference on Formal Structures for Computation and Deduction (FSCD 2017) (Leibniz International Proceedings in Informatics (LIPIcs))*, Dale Miller (Ed.), Vol. 84. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 25:1—-25:22. https://doi.org/10.4230/LIPIcs.FSCD.2017.25

Saunders Mac Lane. 1978. *Categories for the Working Mathematician*. Graduate Texts in Mathematics, Vol. 5. Springer New York, New York, NY. https://doi.org/10.1007/978-1-4757-4721-8

Kenji Miyamoto and Atsushi Igarashi. 2004. A Modal Foundation for Secure Information Flow. In *Proceedings of the Workshop on Foundations of Computer Security (FCS'04)*. 187–203.

Eugenio Moggi. 1991. Notions of computation and monads. *Information and Computation* 93, 1 (1991), 55–92. https://doi.org/10.1016/0890-5401(91)90052-4

Andreas Nuyts and Dominique Devriese. 2018. Degrees of Relatedness. In *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science - LICS '18*. ACM Press, New York, New York, USA, 779–788. https://doi.org/10.1145/3209108.3209119

Andreas Nuyts, Andrea Vezzosi, and Dominique Devriese. 2017. Parametric quantifiers for dependent type theory. *Proceedings of the ACM on Programming Languages* 1, ICFP (2017). https://doi.org/10.1145/3110276

Tomas Petricek, Dominic Orchard, and Alan Mycroft. 2014. Coeffects: A calculus of context-dependent computation. *Proceedings of the 19th ACM SIGPLAN international conference on Functional programming - ICFP '14* (2014), 123–135. https://doi.org/10.1145/2628136.2628160

Benjamin C. Pierce. 2002. *Types and Programming Languages*. The MIT Press.

Gordon D. Plotkin. 1977. LCF considered as a programming language. *Theoretical Computer Science* 5, 3 (1977), 223–255. https://doi.org/10.1016/0304-3975(77)90044-5

Vineet Rajani and Deepak Garg. 2018. Types for Information Flow Control: Labeling Granularity and Semantic Models. In *31st IEEE Symposium on Computer Security Foundations (CSF 2018)*. arXiv:1805.00120

John Rushby. 1986. The Bell and La Padula Security Model. *Draft report, Computer Science Laboratory, SRI* (1986), 1–19. https://doi.org/10.1007/978-1-4419-5906-5_811

Andrei Sabelfeld and David Sands. 2001. A per model of secure information flow in sequential programs. *Higher-Order and Symbolic Computation* 14 (2001), 59–91. https://doi.org/10.1023/A:1011553200337

Naokata Shikuma and Atsushi Igarashi. 2008. Proving Noninterference by a Fully Complete Translation to the Simply Typed lambda-calculus. *Logical Methods in Computer Science* 4, 3 (2008), 10. https://doi.org/10.2168/LMCS-4(3:10)2008

Michael Shulman. 2018. Brouwer's fixed-point theorem in real-cohesive homotopy type theory. *Mathematical Structures in Computer Science* 28, 6 (2018), 856–941. https://doi.org/10.1017/S0960129517000147 arXiv:1509.07584

Thomas Streicher. 2006. *Domain-theoretic Foundations of Functional Programming*. World Scientific.

Stephen Tse and Steve Zdancewic. 2004. Translating dependency into parametricity. In *Proceedings of the ninth ACM SIGPLAN international conference on Functional programming - ICFP '04*. ACM Press, New York, New York, USA, 115. https://doi.org/10.1145/1016850.1016868

Tarmo Uustalu and Varmo Vene. 2008. Comonadic Notions of Computation. *Electronic Notes in Theoretical Computer Science* 203, 5 (2008), 263–284. https://doi.org/10.1016/j.entcs.2008.05.029