

Getting to Grips with Sweedler Runners for Systems

A conversational tutorial, Part V: certified runners
Predicates, quantitative values, and assume–guarantee structure

September 2026

Aim. Part IV explained what must be checked before changing the ambient Sweedler universe. Part V takes the first proposed change seriously. We begin with ordinary predicates, where certification simply means preservation of an invariant. We then pass to predicates valued in a commutative quantale V , describe lifted endofunctors and monads, and formulate what a *certified residual runner* should be. The guiding example is the Moore theory: assumptions live on inputs, invariants on states, and guarantees on outputs. We end by comparing this proposal with categorical assume–guarantee reasoning and by making explicit what remains conjectural about the full duoidal lift.

Contents

1	Why put predicates into the runner itself?	2
2	The first base category: ordinary predicates	3
2.1	The smallest example: a certified endofunction	4
3	Products, function spaces, and why predicates fit the old formulas	4
4	From 2 to a quantitative value object V	5
4.1	Tensor and internal hom	5
5	Two routes: internalise everything, or lift Set-based theories	6
5.1	Route A: build endofunctors directly on $V\text{-Pred}$	6
5.2	Route B: lift a Set-based endofunctor along the predicate fibration	6
6	Lifted algebras: handlers that preserve certification	7
6.1	Unary operations again	7
7	The general certified-runner formula	7
8	Back to $V = 2$: certified unary runners	8
9	Certified Moore machines	9
9.1	A tiny example	10
9.2	A good regulator as a certified Moore–Mealy feedback loop	11
9.3	The good-regulator condition as an adjunction	14
9.4	Assume–guarantee refinement as parameter change	15

10 Quantitative Moore certification and the cartesianity obstruction	16
11 Handler induction after lifting	19
11.1 Unary example	19
11.2 Moore example	20
12 Residual effects: safety through nondeterminism	20
13 Residual probability: several quantitative lifts	21
14 Liftings must respect the monad laws and the tensor	23
15 Assume–guarantee contracts as boundary predicates	23
16 Comparison with categorical assume–guarantee reasoning	24
17 Certified tight maps and certified squares	25
18 What about the left adjoint to handler induction?	26
19 A useful hierarchy of certification strength	26
20 Why the full duoidal question has not gone away	27
21 Where Lyapunov-style values might enter	27
22 A worked summary: one Moore machine, three certification semantics	28
23 What Part V has added	29
24 Where to go next	30
References and orientation	30

1 Why put predicates into the runner itself?

Up to now a runner has been an operational object. In the ordinary Set-based setting it has the form

$$\theta_X : TX \times S \longrightarrow R(X \times S),$$

subject to the runner laws. It says how a T -computation is realised through residual theory R while private state S evolves.

But systems questions are often not merely operational. We also want to know things such as

- whether a safe region of state space is invariant;
- whether an assumption on inputs is sufficient to guarantee an output property;
- whether a quantitative bound decreases under transition;
- whether a probabilistic transition maintains a desired lower bound on safety;

- whether one implementation refines another with respect to a certification order.

There are two rather different ways to add such information.

The first is to build an ordinary runner in **Set** and then attach a theorem to it afterwards:

$$\text{runner} \quad + \quad \text{external proof of safety.}$$

The second is to change the ambient category so that a state already comes equipped with a predicate or value, and every structural map is required to respect it:

$$\text{certification is part of being a morphism.}$$

It is the second route that interests us here.

The target is not merely “a runner together with a predicate.” It is a setting in which theories, handlers, runners, and transformations are themselves *lifted* to predicate-bearing objects, so that preservation of certification is structural.

This is ambitious. We will therefore begin with the simplest possible case and only generalise when the elementary meaning is clear.

2 The first base category: ordinary predicates

Let $2 = \{\text{false} \leq \text{true}\}$. An object of $2\text{-}\mathbf{Pred}$ is a set X equipped with a predicate

$$p : X \longrightarrow 2.$$

Equivalently, it is a subset

$$P \subseteq X.$$

A morphism

$$f : (X, P) \longrightarrow (Y, Q)$$

is a function $f : X \rightarrow Y$ satisfying

$$x \in P \implies f(x) \in Q.$$

Equivalently,

$$p(x) \leq q(fx).$$

Notice that this is *not* the strict slice category $\mathbf{Set}/2$, where we would require equality $p = qf$. We only require truth to be preserved. A certified state may be sent to a certified state; what happens to uncertified points is irrelevant to this particular condition.

There is an evident forgetful functor

$$U : 2\text{-}\mathbf{Pred} \longrightarrow \mathbf{Set}, \quad U(X, P) = X.$$

The fibres of U are posets of predicates on a fixed set. Thus U is naturally viewed as a predicate fibration.

2.1 The smallest example: a certified endofunction

Take our old theory

$$T = \mathbb{N}y,$$

the free monad on one unary operation. A $\{T, \text{Id}\}$ -runner on a set S was simply an endofunction

$$f : S \rightarrow S.$$

Now put a predicate $P \subseteq S$ on the state space and ask that f be a morphism in **2-Pred**:

$$f : (S, P) \rightarrow (S, P).$$

This means exactly

$$\boxed{s \in P \implies f(s) \in P.}$$

So the very first certified runner is nothing exotic. It is simply a dynamical system together with an invariant.

This is the right sanity check for the whole chapter. If our eventual categorical machinery does not reduce to “the designated safe states are closed under transition” in this example, we have chosen the wrong lift.

3 Products, function spaces, and why predicates fit the old formulas

The Set-based runner formulas involved products $X \times S$ and function spaces A^S . We therefore need predicate-bearing versions of these constructions.

For ordinary predicates, define

$$(X, P) \otimes (Y, Q) = (X \times Y, P \times Q),$$

where

$$(x, y) \in P \times Q \iff x \in P \text{ and } y \in Q.$$

The corresponding function-space predicate on Y^X is

$$[P, Q](h) \iff \forall x \in X, x \in P \implies h(x) \in Q.$$

Thus

$$(Y, Q)^{(X, P)}$$

has the ordinary function set Y^X as carrier, but only those functions that send certified inputs to certified outputs satisfy the function-space predicate.

The usual tensor–hom adjunction survives:

$$\mathbf{2-Pred}((Z, R) \otimes (X, P), (Y, Q)) \cong \mathbf{2-Pred}((Z, R), (Y, Q)^{(X, P)}).$$

This is already suggestive for handler induction. In Parts I–II the runner-induced handler lived on A^S . Here the same carrier should acquire the predicate

$$\boxed{h : S \rightarrow A \text{ is certified} \iff h(P) \subseteq Q.}$$

So the move from Set to predicates does not destroy the old formulas. It refines them.

4 From 2 to a quantitative value object V

Now let V be a commutative unital quantale. Concretely, we need:

- a complete lattice $(V, \leq)$;
- a commutative monoidal product $\otimes : V \times V \rightarrow V$ with unit k ;
- preservation of arbitrary joins by $u \otimes -$;
- hence a residual or internal hom $[u, v]$ characterised by

$$u \otimes w \leq v \iff w \leq [u, v].$$

An object of $V\text{-}\mathbf{Pred}$ is a pair

$$(X, p), \quad p : X \rightarrow V.$$

A morphism

$$f : (X, p) \rightarrow (Y, q)$$

is a function satisfying

$$\boxed{p(x) \leq q(fx) \text{ for every } x.}$$

The intended reading depends on V . For $V = 2$, larger means “truer.” For a quantitative resource or cost order, one may reverse the order or choose a different convention. We will not try to make one order orientation serve every application.

4.1 Tensor and internal hom

The natural tensor is

$$(X, p) \otimes (Y, q) = (X \times Y, (x, y) \mapsto p(x) \otimes q(y)).$$

The internal hom has underlying set Y^X and value

$$\boxed{[p, q](h) = \bigwedge_{x \in X} [p(x), q(hx)].}$$

For $V = 2$, this says precisely

$$[p, q](h) = \mathbf{true} \iff \forall x, p(x) \Rightarrow q(hx).$$

Thus $V\text{-}\mathbf{Pred}$ is the natural quantitative generalisation of the predicate category we just used.

The function-space value

$$[p, q](h) = \bigwedge_x [p(x), q(hx)]$$

is going to be the quantitative certificate carried by the handler space A^S . It measures how well the function h transports the state certificate p into the handler certificate q .

5 Two routes: internalise everything, or lift Set-based theories

Part IV ended with an important fork. There are at least two ways to proceed.

5.1 Route A: build endofunctors directly on $V\text{-Pred}$

One may try to work in

$$[V\text{-Pred}, V\text{-Pred}]_{\text{acc}}$$

and reproduce the full duoidal construction there: composition, Day convolution, interchange, monads, comonads, Sweedler homs.

If all the hypotheses hold, this is the cleanest internal story. A theory is literally a monad on the predicate category.

But this is also the most ambitious route. Local presentability, accessibility, closed convolution, and existence of measuring objects all need to be checked for the chosen V . Moreover, the relevant endofunctors cannot in general be arbitrary accessible endofunctors. The monoidal runner calculus requires endofunctors carrying coherent strength for the tensor of $V\text{-Pred}$, and monads whose unit and multiplication respect that strength.

Thus the natural candidate is not simply

$$[V\text{-Pred}, V\text{-Pred}]_{\text{acc}},$$

but an appropriate category of accessible strong endofunctors

$$[V\text{-Pred}, V\text{-Pred}]_{\text{acc}}^{\text{str}},$$

or an equivalent enriched formulation when the required hypotheses identify enrichment with tensorial strength. The precise choice is part of the structure to be established rather than a freely adjustable technical convenience.

5.2 Route B: lift a Set-based endofunctor along the predicate fibration

Suppose instead that we begin with an ordinary endofunctor

$$F : \mathbf{Set} \rightarrow \mathbf{Set}.$$

A *lifting* of F is an endofunctor

$$\bar{F} : V\text{-Pred} \rightarrow V\text{-Pred}$$

such that, ideally strictly or up to a specified isomorphism,

$$U\bar{F} = FU.$$

Thus $\bar{F}$ does not change the underlying Set-level behaviour. It tells us what predicate or quantitative value should be placed on FX when X carries the value p .

Likewise, a monad lifting $\bar{T}$ of a Set-monad T must lift not only the endofunctor but also its unit and multiplication.

This route has two advantages for a tutorial:

1. it keeps the effect theory T from Parts I–III visible underneath the certification layer;
2. it makes the non-uniqueness of certification semantics impossible to ignore.

Established constructions such as TT-lifting and codensity lifting provide systematic ways to construct monad liftings along suitable fibrations. They do not, by themselves, prove that all such liftings assemble into the full duoidal Sweedler universe we ultimately want.

6 Lifted algebras: handlers that preserve certification

Suppose T is a **Set**-monad and $\bar{T}$ a lifting to $V\text{-}\mathbf{Pred}$.

A $\bar{T}$ -algebra is a morphism in $V\text{-}\mathbf{Pred}$

$$a : \bar{T}(X, p) \longrightarrow (X, p)$$

whose underlying **Set**-map is a T -algebra

$$a : TX \rightarrow X.$$

Thus the ordinary handler equations are unchanged, but the algebra map must also respect the predicate or valuation.

6.1 Unary operations again

For $T = \mathbb{N}y$, a **Set**-algebra is an endofunction

$$g : X \rightarrow X.$$

Under the obvious invariant-preserving lift to $2\text{-}\mathbf{Pred}$, a certified handler is simply

$$g : (X, P) \rightarrow (X, P),$$

so

$$x \in P \Rightarrow g(x) \in P.$$

Both the model and comodel sides now carry preservation conditions. This is exactly what we want: certification does not belong exclusively to “states” or exclusively to “handlers.” It belongs to the ambient semantics.

7 The general certified-runner formula

Now suppose $\bar{T}$ and $\bar{R}$ are suitable liftings of **Set**-monads T and R . There is an additional structural requirement hidden in the word “suitable.” The liftings used in the runner calculus must interact coherently with the tensor on $V\text{-}\mathbf{Pred}$. In particular, for the residual theory we need a tensorial strength

$$\mathbf{st}_{(X,p),(Y,q)} : (X, p) \otimes \bar{R}(Y, q) \longrightarrow \bar{R}((X, p) \otimes (Y, q)),$$

natural in both variables and satisfying the usual unit and associativity coherence conditions. For a lifted monad, the strength must also be compatible with the monad unit and multiplication.

This condition is automatic over **Set**, where every endofunctor carries the canonical cartesian strength used throughout Parts I–III. It is not automatic after passing to a genuinely quantitative monoidal category such as $V\text{-}\mathbf{Pred}$.

Thus a monad lifting

$$\bar{R} : V\text{-}\mathbf{Pred} \rightarrow V\text{-}\mathbf{Pred}$$

is not yet sufficient for the full certified-runner calculus. What we require is a suitably strong monad lifting. Let the private runner state be

$$(S, p) \in V\text{-}\mathbf{Pred}.$$

The Set-level runner had natural maps

$$\theta_X : TX \times S \longrightarrow R(X \times S).$$

The lifted version should be a natural family of morphisms in $V\text{-}\mathbf{Pred}$

$$\boxed{\bar{\theta}_{(X,q)} : \bar{T}(X, q) \otimes (S, p) \longrightarrow \bar{R}((X, q) \otimes (S, p)).}$$

Its underlying Set-map is the old θ_X .

This is the basic candidate definition of a certified residual runner once the chosen monad liftings carry the required tensorial strengths.

The morphism condition says, schematically,

$$\begin{aligned} & \text{incoming computation certificate} \otimes \text{runner-state certificate} \\ & \leq \text{residual-computation certificate.} \end{aligned}$$

The exact inequality depends on the chosen liftings $\bar{T}$ and $\bar{R}$. That dependence is a feature, not a bug.

At Set level a runner specifies *how execution proceeds*. At predicate level we additionally choose what it means for a T -computation and an R -computation to be certified. A certified runner is then an execution map that is monotone with respect to those chosen certification semantics.

8 Back to $V = 2$: certified unary runners

Let us verify that the formula says something familiar.

Take

$$T = \mathbb{N}y, \quad R = \text{Id},$$

and use the straightforward safety lift. A runner is an endofunction

$$f : S \rightarrow S.$$

The lifted state object is (S, P) , and the certified-runner condition reduces to

$$f : (S, P) \rightarrow (S, P).$$

Hence

$$\boxed{P(s) \Rightarrow P(f(s)).}$$

After n steps,

$$P(s) \Rightarrow P(f^n(s))$$

follows automatically by composition in $2\text{-}\mathbf{Pred}$.

This trivial-looking fact illustrates the advantage of changing the ambient category: closure under iteration is inherited from categorical composition rather than reproved for every system.

9 Certified Moore machines

Now take our Moore theory

$$T_{\text{Moore}}(A, O).$$

At Set level a comodel/runner into Id is

$$\delta : A \times S \rightarrow S, \quad o : S \rightarrow O.$$

To add assume–guarantee data, equip the interfaces and state with predicates

$$\alpha : A \rightarrow 2, \quad p : S \rightarrow 2, \quad \gamma : O \rightarrow 2.$$

Think of them as

$$\alpha = \text{input assumption}, \quad p = \text{state invariant}, \quad \gamma = \text{output guarantee}.$$

Demand that transition and observation are morphisms in **2-Pred**:

$$\begin{aligned} \delta : (A, \alpha) \otimes (S, p) &\longrightarrow (S, p), \\ o : (S, p) &\longrightarrow (O, \gamma). \end{aligned}$$

Unpacked, these are

$$\boxed{\alpha(a) \wedge p(s) \Rightarrow p(\delta(a, s))},$$

and

$$\boxed{p(s) \Rightarrow \gamma(o(s))}.$$

This is already a recognisable assume–guarantee pattern:

if the current state satisfies the invariant and the incoming value satisfies the assumption,
then the next state again satisfies the invariant; every invariant state produces an
output satisfying the guarantee.

There is a small point here which becomes important as soon as we leave the Boolean case. Strictly speaking, the Moore **get**-cooperation does not merely observe

$$o \longrightarrow O.$$

It observes while retaining the current state:

$$\text{get}_S \longrightarrow O \times S, \quad s \longmapsto (o(s), s).$$

In **2-Pred** this distinction is invisible. Indeed, equipping $O \times S$ with the product predicate gives the condition

$$p(s) \Rightarrow \gamma(o(s)) \wedge p(s),$$

which is equivalent simply to

$$p(s) \Rightarrow \gamma(o(s)).$$

Thus in the Boolean setting we may harmlessly present observation by the map

$$o : (S, p) \longrightarrow (O, \gamma).$$

The equivalence uses the fact that conjunction copies predicates freely:

$$p = p \wedge p.$$

For a genuinely non-cartesian quantitative tensor this simplification need not survive. We return to this point in the next section.

9.1 A tiny example

Let

$$S = \mathbb{Z}, \quad A = \{-1, 0, 1\},$$

with transition

$$\delta(a, s) = s + a.$$

Suppose the certified region is

$$P = \{s \mid 0 \leq s \leq 10\}.$$

There is a state-independent assumption which makes the certification condition hold, namely

$$\alpha = \{0\}.$$

But this is degenerate from the systems point of view: it permits only the input which leaves the state unchanged. Thus the problem is not that no global assumption exists, but that the only global assumption strong enough to preserve the invariant freezes the dynamics.

To admit nontrivial evolution, the admissibility condition must depend on the current state. This is exactly what motivates replacing a predicate

$$\alpha : A \rightarrow 2$$

by a contextual assumption

$$\text{Ass} : A \times S \rightarrow 2.$$

The example therefore exhibits the limitation of state-independent contracts particularly sharply:

global assumptions can ensure safety here only by forbidding nontrivial motion.

This exposes a limitation of a predicate $\alpha : A \rightarrow 2$ that is independent of state. A useful assumption may instead be a predicate

$$\text{Ass} : A \times S \rightarrow 2,$$

for example

$$\text{Ass}(a, s) \iff 0 \leq s + a \leq 10.$$

Then the transition condition becomes

$$p(s) \wedge \text{Ass}(a, s) \Rightarrow p(\delta(a, s)).$$

Likewise, a guarantee may naturally depend on state and output:

$$\text{Gar} : S \times O \rightarrow 2.$$

So the separated predicates α, p, γ are a useful first case, not the final language of assume–guarantee contracts.

9.2 A good regulator as a certified Moore–Mealy feedback loop

We can now return to the Moore–Mealy feedback construction of Part III and see a particularly concrete reason for passing from **Set** to 2-valued predicates. Virgo, Biehl, Baltieri and Capucci [1] study a discrete-time embodied agent coupled to an environment. Their agent is a Moore machine with state space X ,

$$r : X \longrightarrow A, \quad u : X \times S \longrightarrow X,$$

where r produces an action and u updates the agent state after receiving a sensor value. The environment is a Mealy machine with state space Y ,

$$e : Y \times A \longrightarrow Y \times S.$$

Writing

$$e(y, r(x)) = (y', s),$$

the closed Moore–Mealy system has state space $X \times Y$ and update

$$h(x, y) = (u(x, s), y').$$

This is precisely the kind of closed loop constructed in Part III. There we described the wiring by a tight interpretation

$$W : \mathbb{N}y \longrightarrow T_{\text{Moore}}(S, A) + T_{\text{Mealy}}(A, S),$$

and restriction of the product comodel along W produced the autonomous endofunction

$$h : X \times Y \longrightarrow X \times Y.$$

We can therefore regard the present example as asking what happens when that same wired system is lifted from sets to predicates.

The regulation condition is an invariant predicate

Suppose first that

$$G \subseteq X \times Y$$

is the set of *good* joint states: the regulation task requires the coupled system to remain inside G . Virgo et al. call a subset

$$R \subseteq X \times Y$$

a regulating set when, in particular,

$$R \neq \emptyset, \quad R \subseteq G, \quad h(R) \subseteq R.$$

The last condition says that R is forward closed under the closed-loop dynamics. But in the category **Pred**₂, the condition

$$h(R) \subseteq R$$

has an immediate categorical reading. Regard R as its characteristic predicate

$$\chi_R : X \times Y \longrightarrow 2.$$

Then forward closure says exactly that the same underlying function h lifts to a morphism

$$h : (X \times Y, R) \longrightarrow (X \times Y, R)$$

in **Pred**₂. Similarly,

$$R \subseteq G$$

is simply the implication of predicates

$$\chi_R \leq \chi_G.$$

Thus, apart from the separate nonemptiness requirement, a regulating set is simply an invariant predicate lying below the good predicate:

$$\boxed{R \text{ regulates} \implies h : (X \times Y, R) \rightarrow (X \times Y, R), \quad R \leq G.}$$

This is a useful concrete instance of what we mean by *certifying a wired system*. Part III constructed the closed dynamics. Passing to 2-predicates asks for a certificate that those dynamics preserve an acceptable region.

The same predicate becomes a belief map

The next step in the good-regulator theorem initially looks rather different. The authors associate to R a map

$$\psi : X \longrightarrow \mathcal{P}(Y)$$

defined by

$$\boxed{\psi(x) = \{y \in Y \mid (x, y) \in R\}.}$$

The set $\psi(x)$ is interpreted possibilistically as the set of environment states compatible with the agent state x . From the present point of view, however, no new data have been introduced. The relation

$$R \subseteq X \times Y$$

is equivalently its characteristic map

$$\chi_R : X \times Y \longrightarrow 2.$$

Currying gives

$$\widehat{\chi}_R : X \longrightarrow 2^Y,$$

and since

$$2^Y \cong \mathcal{P}(Y),$$

this is precisely the map ψ . In other words,

$$\boxed{\begin{array}{ccc} R \subseteq X \times Y & \longleftrightarrow & \chi_R : X \times Y \rightarrow 2 \\ & \longleftrightarrow & \psi : X \rightarrow 2^Y \cong \mathcal{P}(Y). \end{array}}$$

The authors likewise turn the good set G into a *normative map*

$$\phi : X \longrightarrow \mathcal{P}(Y), \quad \phi(x) = \{y \in Y \mid (x, y) \in G\}.$$

Consequently the inclusion

$$R \subseteq G$$

is equivalent, pointwise, to

$$\boxed{\psi(x) \subseteq \phi(x) \quad \text{for every } x \in X.}$$

So the belief map and normative map are simply the curried forms of the regulating and good predicates.

Forward invariance becomes consistent belief updating

There remains a dynamical condition on the belief map. Given a set of possible environment states $B \subseteq Y$, an action $a \in A$, and a received sensor value $s \in S$, define

$$\text{update}(B, a, s) = \{y' \in Y \mid \exists y \in B : e(y, a) = (y', s)\}.$$

A belief map is called consistent when

$$\boxed{\text{update}(\psi(x), r(x), s) \subseteq \psi(u(x, s))}$$

for every x and s . Virgo et al. prove that, when the model used by the observer is the actual environment Y , this consistency condition is equivalent to forward closure of R :

$$\boxed{h(R) \subseteq R \iff \psi \text{ updates consistently.}}$$

From our perspective this is particularly transparent. The left-hand side says that

$$h : (X \times Y, R) \rightarrow (X \times Y, R)$$

is a morphism of 2-predicates. The right-hand side is the same condition after the predicate R has been curried from

$$X \times Y \rightarrow 2$$

to

$$X \rightarrow 2^Y.$$

The existential quantifier appearing in update is the direct-image operation of ordinary Boolean predicate logic: we existentially quantify over the previous environment state compatible with the current belief. Thus the passage from invariant regulating sets to possibilistic belief updating is not an unrelated construction. It is the same binary predicate viewed in two ways:

$$\boxed{\begin{array}{c} R : X \times Y \rightarrow 2 \\ \text{joint-state invariant} \\ \updownarrow \text{ currying} \\ \psi : X \rightarrow 2^Y \\ \text{environment belief attributed to each agent state.} \end{array}}$$

The good-regulator theorem in predicate language

The main theorem of Virgo et al. now takes a compact form. Schematically, their three conditions

$$R \neq \emptyset, \quad h(R) \subseteq R, \quad R \subseteq G$$

correspond respectively to

$$\begin{aligned} R \neq \emptyset &\iff \exists x . \psi(x) \neq \emptyset, \\ h(R) \subseteq R &\iff \psi \text{ is a consistent belief map,} \\ R \subseteq G &\iff \psi(x) \subseteq \phi(x) \text{ for all } x. \end{aligned}$$

Their “objective” and “subjective” notions of good regulation are therefore equivalent. From the viewpoint of this tutorial, the result can be summarised as

nonempty invariant predicate $R \leq G$ $\updownarrow$ consistent predicate-valued belief $\psi \leq$ normative predicate ϕ .
--

This should not be taken to mean that we have already proved that the whole construction is literally an instance of **Run**(**Pred**₂). For that stronger statement we would want the Moore and Mealy theories themselves to admit appropriate lifts, the tight wiring map W to lift coherently, and the resulting certified comodel to recover the invariant closed system. But all the ingredients visible at the level of the closed-loop dynamics line up exactly as one would hope. The example therefore gives us a useful three-stage picture:

Part III : wire the Moore and Mealy theories, $\Downarrow$ $h : X \times Y \rightarrow X \times Y$, Part V : lift the closed system to an invariant predicate, $h : (X \times Y, R) \rightarrow (X \times Y, R), \quad R \leq G$.

The algebraic theory describes the interface and its wiring; the predicate supplies the certificate. This is precisely the division of labour we hoped a predicate-valued version of **Run** would support.

9.3 The good-regulator condition as an adjunction

The transparency of the preceding equivalence has a categorical reason. It is an instance of the familiar adjunction between direct image and reindexing.

For a function

$$f : X \longrightarrow Y,$$

reindexing sends a predicate $q : Y \rightarrow 2$ to

$$f^*q = q \circ f.$$

Its left adjoint is existential direct image:

$$\exists_f : \mathbf{Pred}(X) \longrightarrow \mathbf{Pred}(Y).$$

Thus

$\exists_f p \leq q \iff p \leq f^*q.$
--

The good-regulator inclusion is exactly of this form. The statement that the image of the regulator’s admissible states or behaviours under the closed-loop update lies inside the desired predicate may be written

$$\exists_f \psi \leq \psi',$$

while the pointwise form says

$$\psi \leq f^* \psi'.$$

Their equivalence is therefore not peculiar to the particular regulator model: it is the adjunction

$$\exists_f \dashv f^*.$$

This formulation passes immediately to V -valued predicates. For a complete lattice V , define

$$(\exists_f p)(y) = \bigvee_{f(x)=y} p(x).$$

Then again

$$\boxed{\exists_f p \leq q \iff p \leq f^* q.}$$

Thus the Boolean good-regulator statement has a quantitative analogue essentially for free: crisp inclusion is replaced by the order in V .

There is a further enriched version. If deterministic evolution f is replaced by a V -valued relation or kernel

$$R : X \times Y \rightarrow V,$$

then direct image becomes the weighted existential

$$(\exists_R p)(y) = \bigvee_{x \in X} p(x) \otimes R(x, y).$$

This is the natural form for resource-sensitive, probabilistic, or graded transitions. The ordinary direct image along a function is recovered from the graph of f .

The good-regulator equivalence is an instance of

$$\exists_f \dashv f^*.$$

Its quantitative extension therefore belongs naturally to the predicate fibration itself, rather than requiring a new regulator-specific argument.

9.4 Assume–guarantee refinement as parameter change

The input and output predicates should be regarded as decorations of the Moore parameters themselves:

$$T_{\text{Moore}}((A, \alpha), (O, \gamma)).$$

This makes the familiar assume–guarantee refinement order look like the quantitative analogue of the mixed-variance parameter change discussed in Part III.

Suppose first that

$$\alpha' \leq \alpha.$$

Then the identity on A is a morphism

$$(A, \alpha') \longrightarrow (A, \alpha)$$

in $V\text{-Pred}$. If a transition is certified under α ,

$$\alpha(a) \otimes p(s) \leq p(\delta(a, s)),$$

then certainly

$$\alpha'(a) \otimes p(s) \leq p(\delta(a, s)).$$

Thus restriction along this parameter map turns a machine certified under α into one certified under α' . In contract language, this is strengthening the assumption on the environment.

Now suppose

$$\gamma \leq \gamma'.$$

Then the identity on O gives a morphism

$$(O, \gamma) \longrightarrow (O, \gamma').$$

From the certified observation condition

$$p(s) \leq \gamma(o(s)) \otimes p(s)$$

we obtain

$$p(s) \leq \gamma'(o(s)) \otimes p(s).$$

Thus a machine satisfying the stronger guarantee γ also satisfies the weaker guarantee γ' .

The two directions therefore reproduce the standard contract-refinement rule:

strengthen assumptions, weaken guarantees.

This is precisely what the mixed variance of Moore parameters predicts. Inputs and outputs occupy opposite parameter positions, so refinement acts contravariantly on assumptions and covariantly on guarantees.

There is, however, a structural point still to verify. To promote this observation from an inequality calculation to a genuine functorial statement, one should construct the lifted family

$$T_{\text{Moore}} : V\text{-}\mathbf{Pred} \times (V\text{-}\mathbf{Pred})^{op} \longrightarrow \mathbf{Th}_V$$

in the appropriate category of quantitative theories, and check that the parameter maps above induce the required tight interpretations. Until that construction is established, the refinement law should be read as a strongly suggestive parameter-level calculation rather than as an already proved functorial theorem.

10 Quantitative Moore certification and the cartesianity obstruction

Now replace $\mathbf{2}$ by a commutative quantale V . Let

$$\alpha : A \rightarrow V, \quad p : S \rightarrow V, \quad \gamma : O \rightarrow V.$$

For the transition cooperation nothing changes in form. Certification asks for

$$\delta : (A, \alpha) \otimes (S, p) \longrightarrow (S, p),$$

hence

$\alpha(a) \otimes p(s) \leq p(\delta(a, s)).$

Observation requires more care. At **Set** level the cooperation for **get** is not just

$$o : S \longrightarrow O,$$

but

$$\text{get}_S : S \longrightarrow O \times S, \quad s \longmapsto (o(s), s),$$

because the operation returns an observation while retaining the current state. Its certified form is therefore

$$\text{get}_S : (S, p) \longrightarrow (O, \gamma) \otimes (S, p).$$

Unpacking the definition of the tensor in **V-Pred** gives

$$\boxed{p(s) \leq \gamma(o(s)) \otimes p(s)}.$$

This is stronger than merely requiring

$$p(s) \leq \gamma(o(s)).$$

The latter says only that the bare observation map

$$o : (S, p) \longrightarrow (O, \gamma)$$

is a morphism. It does not certify the actual Moore cooperation, which must produce the observation and preserve the state simultaneously.

For $V = 2$ with $\otimes = \wedge$, the distinction disappears:

$$p(s) \leq \gamma(o(s)) \wedge p(s) \iff p(s) \leq \gamma(o(s)).$$

This is why the simpler condition in the preceding section was legitimate. More generally, the familiar **Set**-based calculation survives whenever the relevant predicates are freely copyable with respect to the tensor.

The hidden diagonal

The structural source of the difference is the diagonal. In **Set** we have

$$\Delta_S : S \longrightarrow S \times S, \quad s \longmapsto (s, s),$$

and this copying operation is available without further data. To lift it to **V-Pred**, however, we would require

$$\Delta_S : (S, p) \longrightarrow (S, p) \otimes (S, p).$$

Pointwise this says

$$\boxed{p(s) \leq p(s) \otimes p(s)}.$$

For the Boolean tensor this is automatic because

$$p = p \wedge p.$$

For a non-idempotent resource-sensitive tensor it need not hold at all.

The familiar decomposition

$$S \xrightarrow{\Delta_S} S \times S \xrightarrow{o \times_S} O \times S$$

therefore reveals the hidden assumption. In ordinary Set-based comodel calculations, observing a state while retaining it uses copying. Passing from sets to V -valued predicates makes the cost of that copying visible.

For a Moore comodel in $V\text{-Pred}$, observation is not in general free. The certified `get-cooperation` is

$$(S, p) \longrightarrow (O, \gamma) \otimes (S, p),$$

and hence requires

$$p(s) \leq \gamma(o(s)) \otimes p(s).$$

The simpler inequality

$$p(s) \leq \gamma(o(s))$$

is valid only after using additional cartesian or copyability structure.

A sharp non-cartesian example

Take

$$V = [0, 1]$$

with its usual order and multiplicative tensor

$$u \otimes v = uv.$$

Then certified observation requires

$$p(s) \leq \gamma(o(s))p(s).$$

Whenever

$$p(s) > 0,$$

we may divide by $p(s)$ and obtain

$$1 \leq \gamma(o(s)).$$

Since $\gamma(o(s)) \leq 1$, this forces

$$\boxed{\gamma(o(s)) = 1.}$$

Thus every state carrying nonzero certification may emit only a fully certified observation. By contrast, the weaker condition

$$p(s) \leq \gamma(o(s))$$

would allow many intermediate values. The two notions therefore come apart sharply once the tensor ceases to behave cartesianly.

This phenomenon is closely related to the usual Fox-theorem diagnostic: natural copying and deletion are characteristic of cartesian monoidal structure. Here we encounter the same obstruction internally. The comodel side of the theory uses the diagonal, and a quantitative base need not make that diagonal freely available.

It is useful to isolate the relevant condition. Call a predicate $p : S \rightarrow V$ *copyable* when

$$p(s) \leq p(s) \otimes p(s)$$

for every $s \in S$. On the copyable fragment, the familiar Set-based comodel formulas have a chance to survive unchanged. Outside it, operations which inspect state while retaining it carry genuinely additional certification requirements.

The interpretation again depends on V . A value may encode degree of truth, robustness margin, resource availability, a lower safety bound, or something else. But the tensor is now doing more than combining two numerical annotations: it controls whether semantic information may be duplicated.

This point will matter particularly in Part VI. Additive resource and cost semantics are deliberately non-cartesian. There one should expect copying state, potential, or resource annotations to cease being free. The obstruction appearing here is therefore not an inconvenience to be engineered away; it is the first sign of the resource sensitivity that the next part is intended to expose.

11 Handler induction after lifting

This is where the internal hom from Section 3 earns its keep.

Suppose

$$Y = (S, p)$$

is a certified $\{T, R\}$ -runner and

$$(C, q)$$

is a certified R -handler.

In the Set-based story, handler induction produced a T -handler on

$$C^S.$$

In V -**Pred**, the natural carrier is the internal hom

$$(C, q)^{(S, p)}.$$

Its underlying set is still C^S , but a function $h : S \rightarrow C$ receives value

$$\boxed{[p, q](h) = \bigwedge_{s \in S} [p(s), q(hs)]}.$$

For $V = 2$, this simply says

$$h \text{ is certified } \iff p(s) \Rightarrow q(hs) \text{ for all } s.$$

The runner-induced T -operations should now be morphisms of V -predicates automatically, provided the runner interaction and the R -algebra structure are morphisms in the lifted category.

11.1 Unary example

Let $T = \mathbb{N}y$, $R = \text{Id}$, and let

$$f : (S, P) \rightarrow (S, P)$$

be certified. Let (C, Q) be any predicate object. The induced endofunction on C^S is

$$Uh = h \circ f.$$

If h is certified, then

$$s \in P \Rightarrow f(s) \in P \Rightarrow h(f(s)) \in Q.$$

Hence Uh is certified.

So the familiar backward action on predicates or value functions now preserves the higher-order certification predicate on the function space.

11.2 Moore example

For a certified Moore runner, the transition is a morphism

$$\delta : (A, \alpha) \otimes (S, p) \rightarrow (S, p),$$

while the `get`-cooperation is a morphism

$$\text{get}_S : (S, p) \longrightarrow (O, \gamma) \otimes (S, p), \quad s \longmapsto (o(s), s).$$

Thus

$$\alpha(a) \otimes p(s) \leq p(\delta(a, s)),$$

and

$$p(s) \leq \gamma(o(s)) \otimes p(s).$$

The induced handler on C^S still has

$$(\text{step}_a h)(s) = h(\delta(a, s)),$$

and

$$\text{get}((h_i)_{i \in O})(s) = h_{o(s)}(s).$$

Certification tells us when these operations preserve the function-space predicate. In particular, a certified input a transports certified continuations to certified continuations because δ_a preserves the state predicate under the assumed input condition.

The `get`-case now also displays the cartesianity issue from the preceding section. Evaluation at the observed branch does not merely consume the observation: the continuation is evaluated at the same state s . At Set level that reuse of s is invisible because copying is canonical. In a non-cartesian V -valued setting it is precisely the extra occurrence of $p(s)$ in

$$p(s) \leq \gamma(o(s)) \otimes p(s)$$

that certifies this reuse.

This is the predicate-level version of the backward-transformer picture that already appeared in Part I, with one qualification now made explicit: the Set-based formula silently exploits the availability of copying.

12 Residual effects: safety through nondeterminism

The real test is not $R = \text{Id}$ but a nontrivial residual effect.

Take

$$R = \text{List}_+.$$

At Set level a Moore runner had transition

$$\delta : A \times S \rightarrow \text{List}_+(S).$$

There are several possible liftings of List_+ to predicates. The most obvious safety lifting is the *must* lifting:

$$\text{List}_{+\vee}^-(P)([s_1, \dots, s_n]) \iff s_1, \dots, s_n \in P.$$

Then the certified transition condition is

$$\boxed{\alpha(a) \wedge p(s) \Rightarrow \forall s' \in \delta(a, s), p(s').}$$

This is exactly the usual demonic safety reading: every possible implementation successor must remain safe.

But we might instead choose a *may* lifting

$$\text{List}_{+\exists}^-(P)([s_1, \dots, s_n]) \iff \exists i, s_i \in P.$$

Then the same underlying runner is certified by a much weaker condition.

The Set-level effect monad does not determine the certification logic. The lifting $\bar{R}$ is part of the specification. For nondeterminism, “all branches safe” and “some branch safe” are different liftings of the same underlying effect.

This mirrors Part II, where the same R -runner could be interpreted by different R -algebras. Here the variation occurs one level earlier: before handling the effect, we decide what it means for an effectful computation itself to satisfy a predicate.

13 Residual probability: several quantitative lifts

Now return to

$$R = \text{rand}.$$

At Set level we had stochastic Moore transitions

$$\delta : A \times S \rightarrow \text{rand}(S),$$

with deterministic observation.

For Boolean safety, a strong lifting says that every ticket with nonzero probability must lead to a safe state. Schematically,

$$\text{rand}_{\text{must}}^-(P)(N, \pi, k) \iff \forall n \text{ with } \pi(n) > 0, k(n) \in P.$$

Then a certified stochastic runner is almost-surely safe in the strongest support-wise sense.

But for quantitative predicates

$$p : S \rightarrow [0, 1]$$

we may choose a different lift. One natural example is expectation:

$$\boxed{\text{rand}_{\mathbb{E}}(p)(N, \pi, k) = \sum_{n \in N} \pi(n) p(k(n)).}$$

There is now a compatibility condition on the tensor of V which did not arise at Set level. For the expectation lifting to be strong, the canonical Set-level map

$$X \times \mathbf{rand}(Y) \longrightarrow \mathbf{rand}(X \times Y)$$

must lift to a morphism

$$(X, p) \otimes \bar{\mathbf{rand}}_{\mathbb{E}}(Y, q) \longrightarrow \bar{\mathbf{rand}}_{\mathbb{E}}((X, p) \otimes (Y, q)).$$

Pointwise this requires

$$\boxed{p(x) \otimes \mathbb{E}_{\pi}[q] \leq \mathbb{E}_{\pi}[p(x) \otimes q].}$$

Thus the tensor on $[0, 1]$ is not merely part of the interpretation of the resulting inequality. It determines whether the expectation lifting is admissible as a strong residual monad at all.

For the product tensor,

$$u \otimes v = uv,$$

we have equality:

$$p(x) \mathbb{E}[q] = \mathbb{E}[p(x)q].$$

For the Łukasiewicz tensor

$$u \otimes v = \max(0, u + v - 1),$$

fixing u gives a convex function of v . Jensen's inequality therefore yields

$$u \otimes \mathbb{E}[q] \leq \mathbb{E}[u \otimes q],$$

so the expectation lift again admits the required strength.

By contrast, for the Gödel tensor

$$u \otimes v = \min(u, v),$$

the condition can fail. Take

$$u = \frac{1}{2}$$

and let q take the values 0 and 1, each with probability 1/2. Then

$$u \otimes \mathbb{E}[q] = \frac{1}{2},$$

while

$$\mathbb{E}[u \otimes q] = \frac{1}{4}.$$

Hence

$$u \otimes \mathbb{E}[q] \not\leq \mathbb{E}[u \otimes q].$$

The expectation lifting is therefore not strong for this tensor.

The choice of quantale tensor constrains which quantitative effect liftings can participate in the certified-runner calculus. A lifting must satisfy not only the monad laws but also the strength inequalities required to move ordinary certified data through the residual effect.

For expectation, the relevant condition may be read as a convexity requirement: for each fixed $u \in [0, 1]$, the map

$$v \longmapsto u \otimes v$$

must satisfy the Jensen inequality in the direction required above.

14 Liftings must respect the monad laws and the tensor

It is easy to write down a formula assigning a predicate to TX . Not every such formula gives a monad lifting, and not every monad lifting is suitable for the runner calculus.

A genuine lifting $\bar{T}$ must make the unit

$$\eta : (X, p) \rightarrow \bar{T}(X, p)$$

and multiplication

$$\mu : \bar{T}\bar{T}(X, p) \rightarrow \bar{T}(X, p)$$

morphisms in $V\text{-Pred}$, and the monad laws must still hold. For use as a theory or residual inside the monoidal runner construction, it must in addition carry the required tensorial strength, coherently with η and μ .

For probability, for example, an expected-value lifting behaves well because flattening a compound lottery interacts coherently with iterated expectation. A support-based safety lifting behaves well because support of a flattened lottery is generated from supports of the outer and inner lotteries.

But this compatibility should be checked, not assumed.

This is one reason TT-lifting and codensity lifting are useful: they provide general constructions that build the monad laws into the lift rather than leaving us to verify ad hoc formulas one by one.

15 Assume–guarantee contracts as boundary predicates

The simple Moore example used

$$\alpha : A \rightarrow V, \quad \gamma : O \rightarrow V.$$

But realistic assume–guarantee reasoning often needs contextual conditions.

One can instead place predicates on products such as

$$\text{Ass} : A \times S \rightarrow V,$$

$$\text{Gar} : S \times O \rightarrow V.$$

The first may say which inputs are admissible in which states; the second may express a guarantee relating current state and observation.

For Boolean predicates, a certified Moore transition may then satisfy

$$p(s) \wedge \text{Ass}(a, s) \Rightarrow p(\delta(a, s)).$$

For observation there are two related conventions. If

$$\text{Gar} : S \times O \rightarrow V$$

is intended as a predicate on the whole state–output pair produced by the cooperation, then certification asks directly for

$$\boxed{p(s) \leq \text{Gar}(s, o(s))}.$$

In this reading Gar already includes whatever certification is required of the retained state.

If instead the guarantee is separated into an output predicate

$$\gamma : O \rightarrow V$$

and the existing state predicate $p : S \rightarrow V$, then the codomain predicate is their tensor, and the condition becomes

$$\boxed{p(s) \leq \gamma(o(s)) \otimes p(s)}.$$

For $V = 2$ these two presentations collapse to the familiar Boolean assume–guarantee reading. For a non-cartesian tensor they should not be conflated.

This suggests an important conceptual division:

- the *theory* says which operations exist and which equations they satisfy;
- the *runner* says how those operations are realised through state and residual effects;
- the *contract* says under which boundary conditions that realisation is certified.

A mature fibred version of **Run** should organise all three levels rather than encoding contracts by hand for each machine type.

Regulation and assume–guarantee structure. The earlier good-regulator example is deliberately elementary: its certificate is simply an invariant predicate on the joint state space of a closed Moore–Mealy system. Nevertheless it exhibits the same division of labour that motivates assume–guarantee reasoning. The component theories and the tight map between them determine how the system is interconnected; predicates state which joint states are admissible; and preservation of those predicates is a property of the resulting dynamics. More elaborate assume–guarantee systems refine this by separating conditions imposed on a component’s environment from conditions promised by the component itself. The simple invariant

$$R \leq G$$

is therefore a useful first model of the certified-wiring principle before introducing the richer assume–guarantee interface.

16 Comparison with categorical assume–guarantee reasoning

Capucci and Myers develop a categorical assume–guarantee framework for generalized Moore machines. Their setting treats a flavour of generalized Moore machine through a tangency—a fibration equipped with a section—and constructs certified systems compositionally over certified wiring diagrams. Their examples include ordinary Moore machines, partially observable Markov(-decision) processes, and parameterised ODEs; the 2026 work applies the framework to Lyapunov reasoning for input-to-state stability.

The Sweedler route begins from a different place.

Assume–guarantee generalized Moore approach	Proposed Sweedler lift
Starts from generalized Moore/lens-like system structure.	Starts from algebraic theories and residual runners $T \rightsquigarrow R$.
Certification is organised through fibrational/tangency data.	Certification is intended to arise by lifting theories, handlers, and runners to V -valued predicates.
Wiring composition is primary.	Stateful interpretation between effect theories is primary; tight wiring maps and runner composition coexist.
Lyapunov and safety certificates are attached compositionally to generalized machines.	The aspiration is to certify not only machines but translations between different operational theories.

At present this is a comparison target, not an established equivalence. A successful Part V programme would have to show precisely how the two constructions meet in their common Moore-machine fragment.

The attractive possibility is that assume–guarantee certification lives not only on objects and wires, but on *stateful implementations between theories*. Then a certificate could survive translation from one effect language to another, rather than being reconstructed after every change of machine formalism.

17 Certified tight maps and certified squares

Part III distinguished tight theory maps

$$F : T \rightarrow R$$

from loose runners

$$T \rightsquigarrow R.$$

The predicate lift should refine both.

A lifted tight map

$$\bar{F} : \bar{T} \rightarrow \bar{R}$$

should induce restriction

$$\mathbf{Alg}(\bar{R}) \rightarrow \mathbf{Alg}(\bar{T})$$

without changing the underlying carrier or losing its certificate.

Likewise a cell

$$\begin{array}{ccc} T & \xRightarrow{Y} & R \\ F \downarrow & & \downarrow G \\ T' & \xRightarrow{Y'} & R' \end{array}$$

should now compare certified runners. Its underlying state map should itself be a morphism in $V\text{-}\mathbf{Pred}$, so refinement of implementations also refines certification.

This is one reason for wanting the whole double category to live over the predicate world, rather than certifying isolated horizontal arrows after the fact.

18 What about the left adjoint to handler induction?

In Parts I–II a runner-induced handler functor

$$\Theta_Y : \mathbf{Alg}(R) \rightarrow \mathbf{Alg}(T)$$

had a left adjoint in the examples we studied. At Set level its underlying carrier transformation was based on the exponential $(-)^S$, with the left adjoint obtained by a model–comodel or residual execution quotient.

In the predicate world, the right-hand carrier becomes an internal hom

$$(C, q)^{(S, p)}.$$

Since $V\text{-}\mathbf{Pred}$ is monoidal closed under our quantale hypotheses, the ambient tensor–hom adjunction is still available:

$$- \otimes (S, p) \dashv (-)^{(S, p)}.$$

This strongly suggests a lifted execution adjunction

$$L_Y \dashv \Theta_Y$$

when the relevant algebra categories and coequalizers exist and when the liftings are compatible with the monoidal structure.

But this is precisely the sort of statement that should be proved in the full duoidal development rather than asserted from Set-level intuition. Part V therefore records it as a structural target.

19 A useful hierarchy of certification strength

The examples suggest several distinct places where certification may enter.

1. **State certificate only.** A predicate $p : S \rightarrow V$ is preserved by the dynamics.
2. **Boundary certificate.** Inputs carry assumptions and outputs carry guarantees.
3. **Effect certificate.** A lifting $\bar{R}$ says what it means for a residual effect itself to satisfy a predicate or quantitative bound.
4. **Handler certificate.** An R -algebra is required to preserve the lifted predicate, so interpreting the residual effect does not destroy certification.

5. **Implementation certificate.** The runner interaction $\bar{\theta}$ is a morphism in $V\text{-}\mathbf{Pred}$.
6. **Refinement certificate.** Tight maps and 2-cells respect or improve the chosen values.

The virtue of an ambient categorical lift is that these levels can potentially be derived from the same notion of morphism rather than represented by unrelated proof obligations.

20 Why the full duoidal question has not gone away

It would be easy at this point to declare victory: we have written sensible predicate inequalities for transitions and outputs.

But Part IV taught us to ask a harder question. Sweedler theory needs not just individual lifted monads but a compatible ambient pair of monoidal structures.

If we take

$$\mathcal{D}_V = [V\text{-}\mathbf{Pred}, V\text{-}\mathbf{Pred}]_{\text{acc}},$$

we still need to know:

- whether $V\text{-}\mathbf{Pred}$ has the local presentability and closed monoidal structure required for the accessible convolution construction;
- whether composition and Day convolution on $\mathcal{D}_V$ satisfy the duoidal interchange in the desired subcategory;
- whether the specific monad liftings used for certification lie inside that subcategory;
- whether the relevant universal measuring comonoids exist;
- whether the forgetful comparison to $\mathbf{Set}$ is duoidal enough to send certified Sweedler homs and runners to their underlying ordinary ones.

Alternatively, if we restrict to *liftings* of $\mathbf{Set}$ -endofunctors over the predicate fibration, we need to know whether those liftings themselves form a duoidal category closed under the constructions used in Sweedler theory.

Neither question is settled merely by constructing one safe lifting of one monad.

Part V separates two achievements:

elementary certified runners can already be written down and used;
a functorial duoidal category of all such lifted runners is a further theorem to prove.

21 Where Lyapunov-style values might enter

The quantitative language of $V\text{-}\mathbf{Pred}$ is suggestive for stability, although we should not overstate what has been established.

A Lyapunov function is itself a valuation

$$L : S \rightarrow [0, \infty],$$

with an inequality expressing decrease along dynamics. If one chooses the order on $[0, \infty]$ so that “better” certificates are larger in the categorical order, the Lyapunov condition can be made to look like preservation or improvement of a V -predicate.

For a deterministic transition $f : S \rightarrow S$, one wants an inequality of the form

$$L(f(s)) \leq L(s)$$

or a strengthened contraction condition. For stochastic transitions, the corresponding condition may involve expected value or another risk functional.

This is exactly the sort of situation where the choice of quantale order and monad lifting matters. It is too early to claim that arbitrary Lyapunov theory is simply “a V -predicate lift.” But the formal similarity explains why the valued-predicate route is worth pursuing, especially in light of the assume–guarantee treatment of Lyapunov functions in generalized Moore machines.

22 A worked summary: one Moore machine, three certification semantics

Fix the same underlying stochastic Moore runner

$$\delta : A \times S \rightarrow \text{rand}(S), \quad o : S \rightarrow O.$$

We can place different certification layers over it.

Boolean support safety

Let $P \subseteq S$. Under the must-support lift,

$$P(s) \wedge \alpha(a) \Rightarrow \text{every } s' \text{ in the support of } \delta(a, s) \text{ lies in } P.$$

Probability-of-safety value

Let $p : S \rightarrow [0, 1]$ measure a degree of safety and use the expectation lift. Then

$$\alpha(a) \otimes p(s) \leq \mathbb{E}[p(S') \mid a, s].$$

Worst-case nondeterministic safety

Replace rand by List_+ and use the must lifting. Then

$$\alpha(a) \wedge P(s) \Rightarrow \bigwedge_{s' \in \delta(a, s)} P(s').$$

The underlying abstract Moore interface can remain unchanged throughout. What changes is the residual effect and the lifting that interprets certification on that effect.

This is exactly the modularity the Sweedler architecture is supposed to buy us.

23 What Part V has added

We can now enlarge the dictionary from the first four parts.

Object	Certified reading
$(X, p) \in V\text{-}\mathbf{Pred}$	A set together with a truth, safety, resource, robustness, or quantitative value.
Morphisms in $V\text{-}\mathbf{Pred}$	Maps that preserve or improve the chosen value.
Lifted monad $\bar{T}$	A theory together with a semantics of certification for T -computations.
$\bar{T}$ -algebra	A handler that respects certification.
Certified runner $\bar{\theta}$	A stateful implementation that preserves the lifted source/residual values.
Internal hom $(C, q)^{(S, p)}$	The handler space C^S equipped with a certificate measuring how well functions transport state values into handler values.
Boundary predicates	Assumptions on admissible inputs and guarantees on outputs or whole state–output boundaries.
Copyability	The additional condition required when a cooperation retains state while producing an observation in a non-cartesian base.
Choice of lifting	The logic of certification for an effect: must, may, expectation, risk, and so on.

The main conceptual lesson is

certification belongs to the semantics of effects and implementations, not just to state sets.

A second lesson is equally important:

quantitative certification can make copying visible.

There is a broader structural lesson behind these examples. Passing from **Set** to a quantitative base makes visible two pieces of cartesian structure which were previously automatic:

$$\begin{aligned} \text{copying} &: S \longrightarrow S \otimes S, \\ \text{strength} &: X \otimes RY \longrightarrow R(X \otimes Y). \end{aligned}$$

They govern different sides of the runner construction. Copying controls the comodel side: an operation such as **get**, which observes a state while retaining it, silently uses the diagonal. Strength controls the residual side: an effect such as probabilistic choice must interact coherently with independently certified data. Over **Set**, both structures are canonical and therefore largely invisible. Over $V\text{-}\mathbf{Pred}$, neither is automatic.

This gives a useful diagnostic for quantitative generalisations of the earlier theory. One must ask separately whether states are copyable and whether residual effects are strong for the chosen

tensor. The first question constrains the certified cooperations; the second constrains which lifted monads can serve as residual theories. In this sense, moving away from a cartesian base does not merely enrich the values attached to the old semantics: it exposes structural assumptions built into the Set-based runner calculus.

This observation also prepares the transition to Part VI. In resource-sensitive and amortised settings, copying is deliberately no longer free, while strength and interchange become substantive compatibility conditions. The phenomena encountered there are therefore not new complications unrelated to the predicate semantics of this part; they are the same loss of hidden cartesianness seen from the resource side.

24 Where to go next

The next part should change the quantitative question. Instead of asking whether a transition preserves a safety or truth value, we ask how computations carry a *grade* or *resource index*.

Let C be an ordered commutative monoid, regarded as a thin monoidal category. Copresheaves on C support Day convolution, with representables behaving like canonical resource states:

$$y(p) \otimes_{\text{Day}} y(q) \cong y(p + q).$$

This is the setting in which cost and potential for amortised analysis already form an adjoint pair. So the natural continuation is

Part VI: graded runners, C -sets, and amortised resources.

There we can ask whether graded monads appear as monoids in a convolution category, whether the endofunctor category of C -sets passes the Part IV duoidal checklist, and whether Sweedler runners can transport not merely effects but amortised resource bounds.

References and orientation

References

- [1] Nathaniel Virgo, Martin Biehl, Manuel Baltieri, and Matteo Capucci, *A “Good Regulator Theorem” for Embodied Agents*. arXiv:2508.06326v2, revised August 21, 2025

For the general Sweedler architecture and the Set-based residual runner construction:

- I. López Franco and C. Vasilakopoulou, *Duoidal categories, measuring comonoids and enrichment*, arXiv:2005.01340.
- D. McDermott, E. Rivas, and T. Uustalu, *Sweedler Theory of Monads*, FoSSaCS 2022, LNCS 13242, 428–448.

For lifting monads along predicate and related fibrations:

- S. Katsumata, *A Semantic Formulation of TT-lifting and Logical Predicates for Computational Metalanguage*, CSL 2005.

- S. Katsumata, T. Sato, and T. Uustalu, *Codensity Lifting of Monads and its Dual*, Logical Methods in Computer Science 14(4), 2018. Codensity lifting applies to a broad range of fibrations and studies lifting of algebraic operations as well as monads.

For categorical assume–guarantee reasoning and its Lyapunov application:

- M. Capucci and D. J. Myers, *Compositionality of Lyapunov functions via assume–guarantee reasoning*, arXiv:2604.03017, 2026. The paper develops assume–guarantee certified generalized Moore machines using fibrational/tangency structure and applies the framework to input-to-state stability and Lyapunov functions.

The picture to retain from Part V is

