Decorated proofs for states and exceptions

Dominique Duval
with J.-G. Dumas, L. Fousse, J.-C. Reynaud

LJK, University of Grenoble

June 13, 2012
IFIP WG1.3 meeting, Salamanca

Effects

Several approaches for managing a large specification (/program):

- Modularity. Breaking down the specification into modules.
  - Each module is right:
  its semantics is a "part" of the intended semantics.
- Refinement. Stepwise transformation of specifications.
  - Each step is right:
  its semantics is a "generalization" of the intended semantics.
- Effects. Approximations, stepwise improved.
  - Each step is wrong:
  its "apparent" semantics may be far from the intended semantics.
  - Ex.: states, exceptions, ...

States and exceptions

"pure"	states	exceptions
Monads		
$f : X \to Y$	$f : X \to (Y \times S)^S$	$f : X \to Y + E$
Lawvere theories + "handlers"		
operations	lookup, update	raise, (handle)
Decorations		
$f : X \to Y$	$f : X \times S \to Y \times S$	$f : X + E \to Y + E$
operations	lookup, update	raise, catch, handle

Moggi 91, Plotkin-Power 02,
Schröder-Mossakowski 04, Levy 06, Plotkin-Pretnar 09, ...

Semantics of effects

Several kinds of semantics for computational effects

- mathematical (past Aussois meeting)
- logical (this Salamanca meeting)
- operational (some next meeting?...)

Outline

States

Exceptions

Diagrammatic logics

A property of imperative languages

The annihilation lookup-update property:

the command $X := X$ does not modify the state

Proof. Let n be the value of X in the current state.

- First "X" (on the right) is evaluated as n.
- Then "$X :=$" (on the left) puts the value of X to n, without modifying the value of other locations.

Hence the state is not modified. $\square$

A property of imperative languages

The annihilation lookup-update property:

the command $X := X$ does not modify the state

Proof. Let n be the value of X in the current state.

- First "X" (on the right) is evaluated as n.
- Then "$X :=$" (on the left) puts the value of X to n, without modifying the value of other locations.

Hence the state is not modified. □

Remark. Observational equality of states:
a state is characterized by the values of all locations.

Simplification. In this talk, there is only one location X:
a state is characterized by the value of the location X.

A proof

Specification.

Sorts:	N, U (Unit)	
Operations:	$\ell : U \to N$	(lookup)
	$u : N \to U$	(update)
Equation:	$\ell \circ u \equiv id_N$	
Property:	ℓ "mono"	($\equiv$ is observation wrt ℓ)

A proof

Specification.

Sorts:	N, U (Unit)	
Operations:	$\ell : U \to N$	(lookup)
	$u : N \to U$	(update)
Equation:	$\ell \circ u \equiv id_N$	
Property:	ℓ "mono"	($\equiv$ is observation wrt ℓ)

Theorem.

$$\boxed{u \circ \ell \equiv id_U}$$

Proof.

$$\begin{array}{l}(\text{subst})\ \dfrac{\ell \circ u \equiv id}{\ell \circ u \circ \ell \equiv \ell} \\ (\text{mono})\ \dfrac{\ell \circ u \circ \ell \equiv \ell}{u \circ \ell \equiv id}\end{array}$$

A proof

Specification.

Sorts:	N, U (Unit)	
Operations:	$\ell : U \to N$	(lookup)
	$u : N \to U$	(update)
Equation:	$\ell \circ u \equiv id_N$	
Property:	ℓ "mono"	($\equiv$ is observation wrt ℓ)

Theorem.

$$\boxed{u \circ \ell \equiv id_U}$$

Proof.

$$\begin{array}{l}(\text{subst})\ \dfrac{\ell \circ u \equiv id}{\ell \circ u \circ \ell \equiv \ell} \\ (\text{mono})\ \dfrac{\ell \circ u \circ \ell \equiv \ell}{u \circ \ell \equiv id}\end{array}$$

Remark. A shorter proof (since $U = Unit$):

$$(\text{unit})\ \frac{u \circ \ell : U \to U}{u \circ \ell \equiv id}$$

The first proof looks "good", this one looks "bad"!?

A decorated logic for states

Terms are classified:

- $f^{(0)}$: f is pure if it cannot use nor modify the state.
- $f^{(1)}$: f is an accessor if it can use the state, not modify it.
- $f^{(2)}$: f is a modifier if it can use and modify the state.

Hierarchy rules: $\dfrac{f^{(0)}}{f^{(1)}}$, $\dfrac{f^{(1)}}{f^{(2)}}$.

Equations are classified:

- $f \equiv g$: strong equation: f and g return the same value and they have the same effect on the state.
- $f \sim g$: weak equation: f and g return the same value but they may have different effects on the state.

Hierarchy rule: $\dfrac{f \equiv g}{f \sim g}$.

A decorated proof

Specification.

Sorts:	N, U (Unit)	
Operations:	$\ell^{(1)} : U \to N$	(lookup)
	$u^{(2)} : N \to U$	(update)
Equation:	$\ell \circ u \sim id_N$	(same value)
Property:	ℓ "mono"	($\equiv$ is observation wrt ℓ)

A decorated proof

Specification.

Sorts:	N, U (Unit)	
Operations:	$\ell^{(1)} : U \to N$	(lookup)
	$u^{(2)} : N \to U$	(update)
Equation:	$\ell \circ u \sim id_N$	(same value)
Property:	ℓ "mono"	($\equiv$ is observation wrt ℓ)

Theorem.

$$\boxed{u \circ \ell \equiv id_U}$$

Proof.

$$\begin{array}{l}(\text{subst})\\(\text{mono})\end{array} \dfrac{\dfrac{\ell \circ u \sim id}{\ell \circ u \circ \ell \sim \ell}}{u \circ \ell \equiv id}$$

A decorated proof

Specification.

Sorts:	N, U (Unit)	
Operations:	$\ell^{(1)} : U \to N$	(lookup)
	$u^{(2)} : N \to U$	(update)
Equation:	$\ell \circ u \sim id_N$	(same value)
Property:	ℓ "mono"	($\equiv$ is observation wrt ℓ)

Theorem.

$$\boxed{u \circ \ell \equiv id_U}$$

Proof.

$$\begin{array}{l}(\text{subst})\\(\text{mono})\end{array} \dfrac{\dfrac{\ell \circ u \sim id}{\ell \circ u \circ \ell \sim \ell}}{u \circ \ell \equiv id}$$

Remark. A "good" proof is a proof which can be decorated.

An explicit proof

Specification.

Sorts:	N, U (Unit)	and $\underline{S}$ (the sort of states)
Projections:	$val_X : X \leftarrow X \times \underline{S} \rightarrow \underline{S} : st_X$	
Operations:	$\ell : \underline{S} \rightarrow N$	(lookup)
	$u : N \times \underline{S} \rightarrow \underline{S}$	(update)
Equation:	$\ell \circ u \equiv val_N$	
Property:	ℓ mono	($\equiv$ is observation wrt ℓ)

An explicit proof

Specification.

Sorts:	N, U (Unit)	and $\underline{S}$ (the sort of states)
Projections:	$val_X : X \leftarrow X \times \underline{S} \rightarrow \underline{S} : st_X$	
Operations:	$\ell : \underline{S} \rightarrow N$	(lookup)
	$u : N \times \underline{S} \rightarrow \underline{S}$	(update)
Equation:	$\ell \circ u \equiv val_N$	
Property:	ℓ mono	($\equiv$ is observation wrt ℓ)

Theorem.

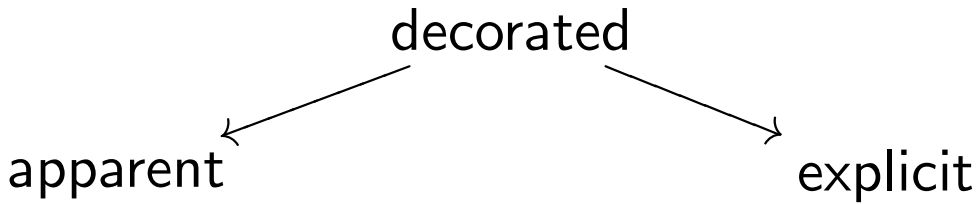
$$\boxed{u \circ \langle \ell, id_S \rangle \equiv id_S}$$

Proof.

$$\text{(trans)}\ \frac{\text{(subst)}\ \dfrac{\ell \circ u \equiv val}{\ell \circ u \circ \langle \ell, id_S \rangle \equiv val \circ \langle \ell, id_S \rangle} \qquad \text{(proj)}\ \dfrac{}{val \circ \langle \ell, id_S \rangle \equiv \ell}}{\text{(mono)}\ \dfrac{\ell \circ u \circ \langle \ell, id_S \rangle \equiv \ell}{u \circ \langle \ell, id_S \rangle \equiv id}}$$

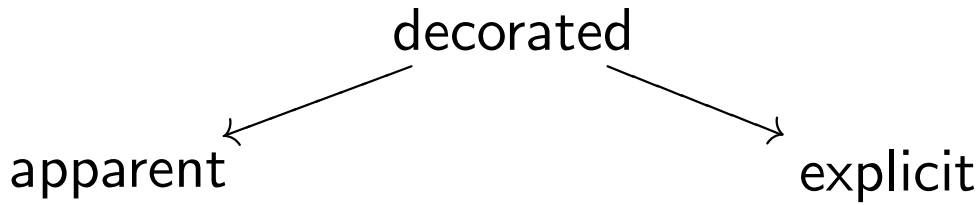
A span for states

A span in a category of logics:

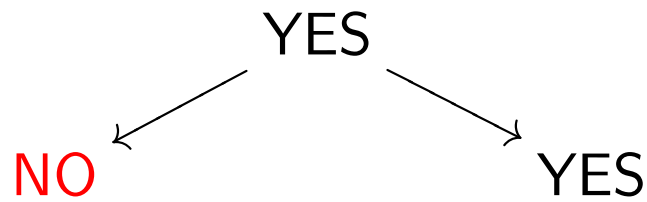


A span for states

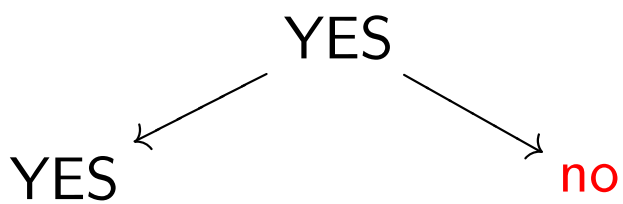
A span in a category of logics:



Does the intended semantics form a model?



Do the proofs respect the effect?



Outline

States

Exceptions

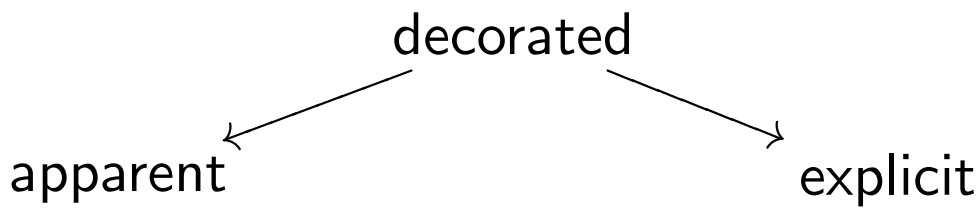
Diagrammatic logics

Duality

Fact. There is a duality between states and exceptions.

[Dumas&Duval&Fousse&Reynaud'12] MSCS (4 p.)

Consequence. A span for exceptions.



A decorated logic for exceptions

Terms are classified:

- $f^{(0)}$: f is pure if it cannot throw nor catch exceptions.
- $f^{(1)}$: f is a propagator if it can throw exceptions, not catch them.
- $f^{(2)}$: f is a catcher if it can throw and catch exceptions.

Hierarchy rules: $\dfrac{f^{(0)}}{f^{(1)}}$, $\dfrac{f^{(1)}}{f^{(2)}}$.

Equations are classified:

- $f \equiv g$: strong equation: f and g coincide on ordinary values and on exceptions.
- $f \sim g$: weak equation: f and g coincide on ordinary values but they may be different on exceptions.

Hierarchy rule: $\dfrac{f \equiv g}{f \sim g}$.

A decorated proof

Simplification. There is only one type of exceptions.

Sorts:	N, 0 (empty)	
Operations:	$t^{(1)} : N \to 0$	(tag, for throw)
	$c^{(2)} : 0 \to N$	(untag, for catch)
Equation:	$c \circ t \sim id_N$	(same on ordinary values)
Property:	t "epi"	($\equiv$ is "same on the image of t")

A decorated proof

Simplification. There is only one type of exceptions.

Sorts:	N, 0 (empty)	
Operations:	$t^{(1)} : N \to 0$	(tag, for throw)
	$c^{(2)} : 0 \to N$	(untag, for catch)
Equation:	$c \circ t \sim id_N$	(same on ordinary values)
Property:	t "epi"	($\equiv$ is "same on the image of t")

Theorem.

$$\boxed{t \circ c \equiv id_0}$$

Proof.

$$\begin{array}{r} (\text{repl}) \\ (\text{epi}) \end{array} \dfrac{\dfrac{c \circ t \sim id}{t \circ c \circ t \sim t}}{t \circ c \equiv id}$$

Encapsulation

The public operations throw and try/catch
or raise and handle
are defined in terms of the private operations tag and untag.

- For each Y:

$$throw_Y = [\,]_Y \circ t : N \to Y$$

- For each $f : X \to Y$ and $g : N \to Y$:

$$try\{f\}catch\{g\} = \nabla(TRY\{f\}catch\{g\}) : X \to Y$$

where $TRY\{f\}catch\{g\} = [\, id_Y \mid g \circ c \,] \circ f$

Encapsulation

The public operations throw and try/catch
or raise and handle
are defined in terms of the private operations tag and untag.

- For each Y:

$$throw_Y = [\,]_Y \circ t : N \to Y$$

- For each $f : X \to Y$ and $g : N \to Y$:

$$try\{f\}catch\{g\} = \nabla(TRY\{f\}catch\{g\}) : X \to Y$$
$$\text{where } TRY\{f\}catch\{g\} = [\, id_Y \mid g \circ c \,] \circ f$$

Corollary. For each $f : X \to Y$:

$$\boxed{try\{f\}catch\{throw_Y\} \equiv f}$$

Outline

States

Exceptions

Diagrammatic logics

Some category theory

A diagrammatic logic is

- a left adjoint functor between categories with colimits
- and a localization.

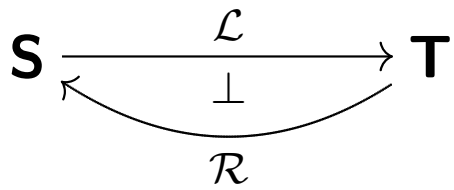
$$\mathbf{S} \underset{\mathcal{R}}{\overset{\mathcal{L}}{\underset{\longleftarrow}{\overset{\longrightarrow}{\bot}}}} \mathbf{T} \quad \text{with} \quad \mathcal{L} \circ \mathcal{R} \cong Id_{\mathbf{T}}$$

induced by a morphism of limit sketches

$$\mathbf{E}_S \xrightarrow{\mathbf{e}} \mathbf{E}_T$$

[Gabriel-Zisman 67, Ehresmann 68]

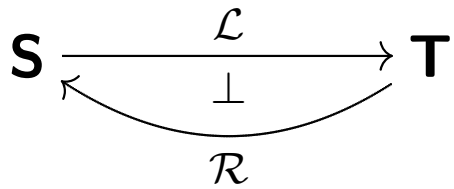
Models



- **S** is the category of specifications
- **T** is the category of theories
- the set of models of a specification Σ with values in a theory Θ is:

$$Mod_{\mathcal{L}}(\Sigma, \Theta) = Hom_{\mathbf{T}}(\mathcal{L}\Sigma, \Theta) \cong Hom_{\mathbf{S}}(\Sigma, \mathcal{R}\Theta)$$

Proofs



- a rule with hypothesis $\mathcal{H}$ and conclusion $\mathcal{C}$ is a fraction from $\mathcal{C}$ to $\mathcal{H}$

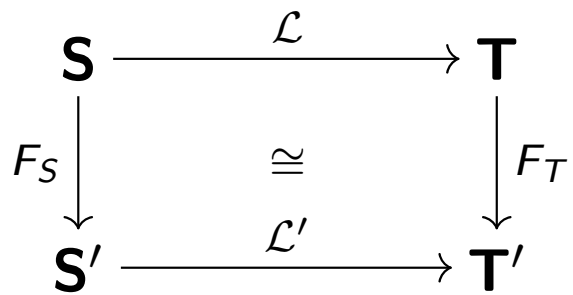
$$\mathcal{H} \longleftarrow \mathcal{H}' \longrightarrow \mathcal{C}$$

- an instance of $\mathcal{H}$ in Σ is a fraction from $\mathcal{H}$ to Σ
- an inference step is a composition of fractions

$$\begin{array}{ccccc} \mathcal{H} & & \longleftarrow & & \mathcal{H}' & & \longrightarrow & & \mathcal{C} \\ \downarrow & & (PO) & & \downarrow & & \swarrow & & \\ \Sigma & \longleftarrow & \Sigma_H & \longrightarrow & \Sigma_C & & & & \end{array}$$

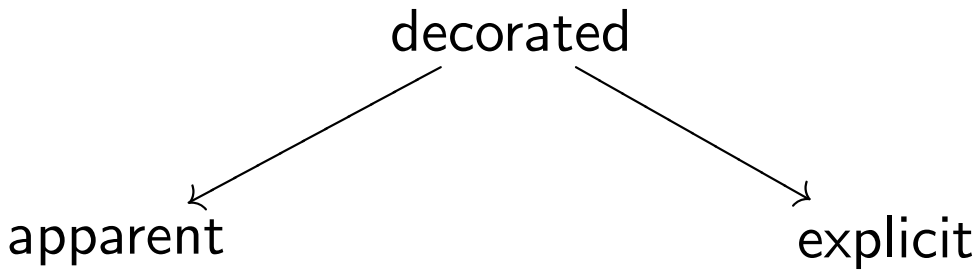
The category of diagrammatic logics

A morphism of diagrammatic logics $F : \mathcal{L} \to \mathcal{L}'$
is a pair of locally presentable functors such that:

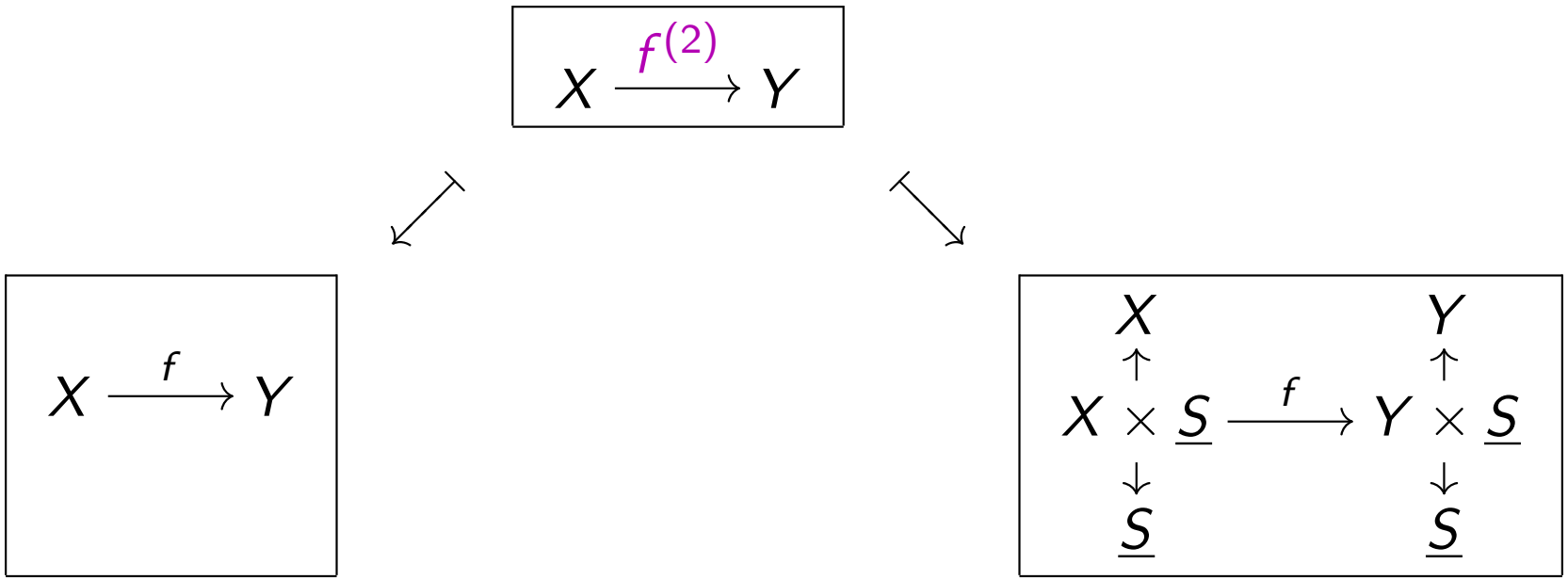


induced by a commutative square of limit sketches

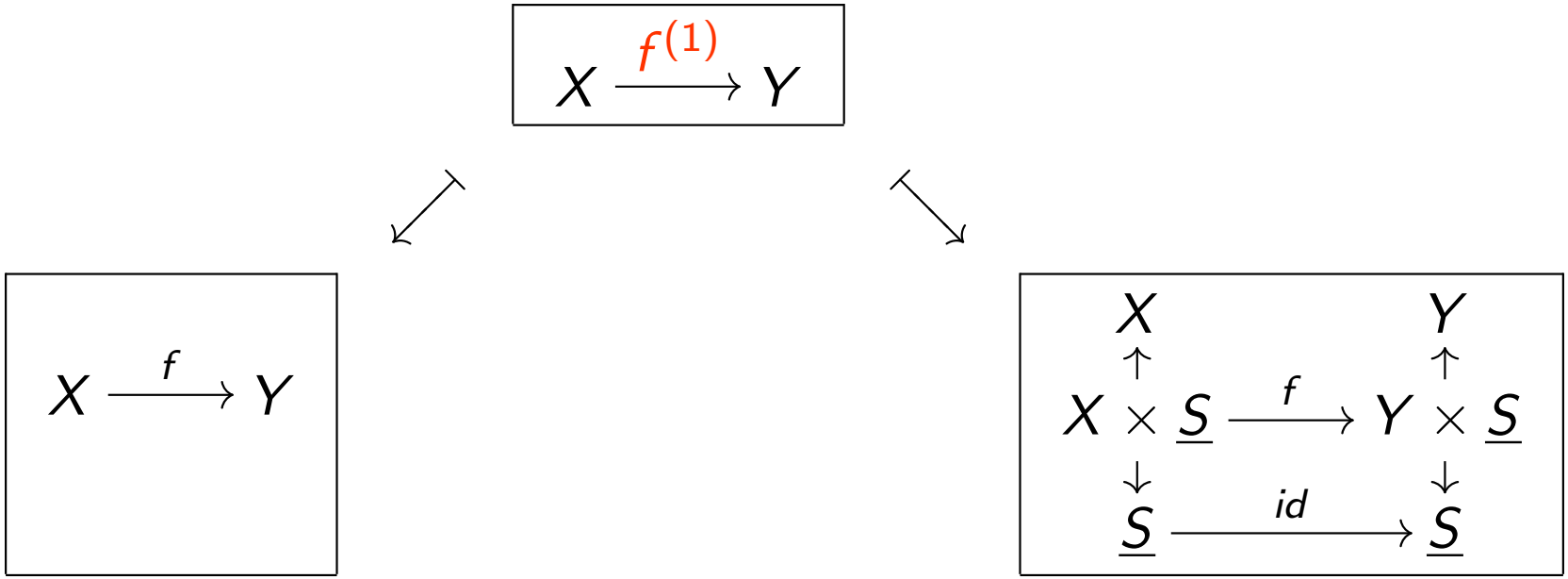
A span of logics for states



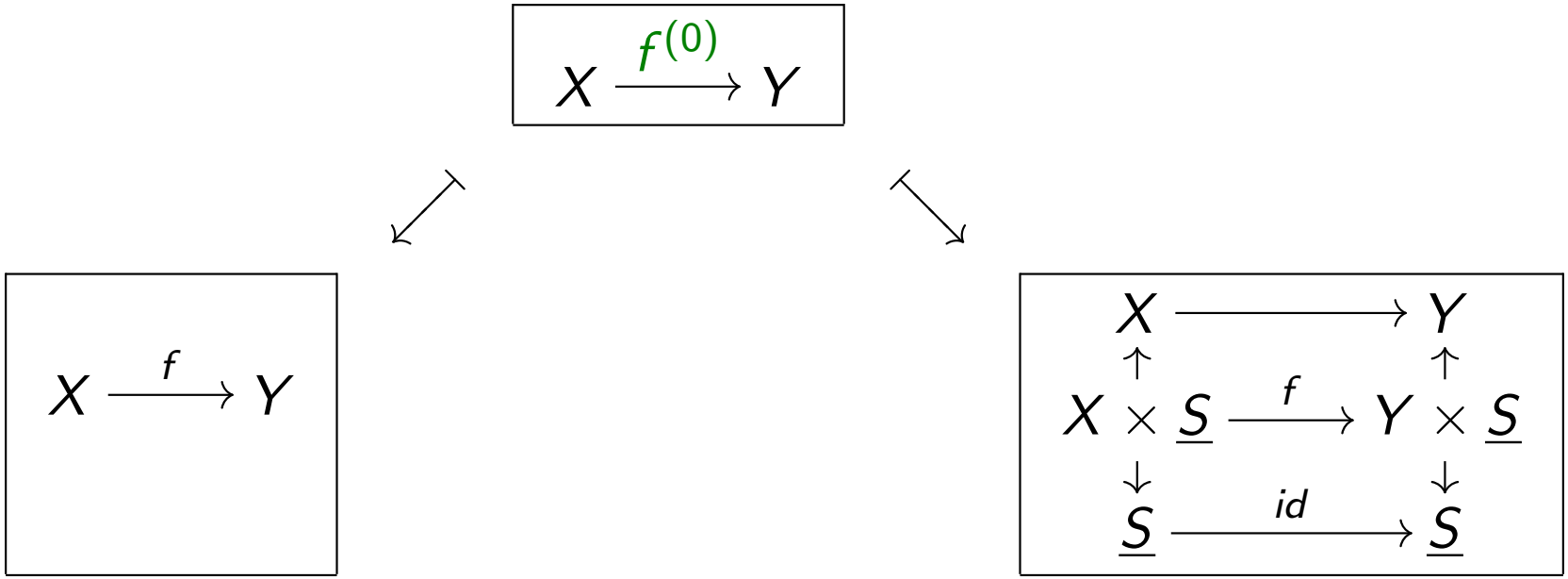
Modifiers


$X \xrightarrow{f^{(2)}} Y$
$X \xrightarrow{f} Y$
$\begin{array}{ccc} X & & Y \\ \uparrow & & \uparrow \\ X \times \underline{S} & \xrightarrow{f} & Y \times \underline{S} \\ \downarrow & & \downarrow \\ \underline{S} & & \underline{S} \end{array}$

Accessors


$X \xrightarrow{f^{(1)}} Y$
$X \xrightarrow{f} Y$
$X \quad Y$
$X \times \underline{S} \xrightarrow{f} Y \times \underline{S}$
$\underline{S} \xrightarrow{id} \underline{S}$

Pure expressions



Conclusion

Decorated logics work quite well for effects,
mainly because of their notion of morphism.

A morphism of logics
maps specifications to specifications
and proofs to proofs.
and it is made of left adjoint functors.

Future work include:

- operational semantics for effects,
- decorated proofs in Coq?. . .

An assistant for decorated proofs?

It seems quite easy to use Coq for decorated proofs.
(Work in progress with Damien Pous)

Remark. This is easy because there is NO rule like:

$$\frac{f^{(2)} \equiv g^{(0)}}{f^{(0)}}$$

Property. The extension
from "the logic for signatures" to "the logic for specifications",
which is conservative for the apparent logic,
remains conservative for the decorated logic.

$\Rightarrow$ Towards a notion of signature for diagrammatic logics?
(Work in progress with Arthur Guillon)