Recovering Purity with Comonads and Capabilities

The marriage of purity and comonads

ANONYMOUS AUTHOR(S)

In this paper, we take a pervasively effectful (in the style of ML) typed lambda calculus, and show how to *extend* it to permit capturing pure expressions with types. Our key observation is that, just as the pure simply-typed lambda calculus can be extended to support effects with a monadic type discipline, an impure typed lambda calculus can be extended to support purity with a *comonadic* type discipline.

We establish the correctness of our type system via a simple denotational model, which we call the *capability space* model. Our model formalizes the intuition common to systems programmers that the ability to perform effects should be controlled via access to a permission or capability, and that a program is *capability-safe* if it performs no effects that it does not have a runtime capability for. We then identify the axiomatic categorical structure that the capability space model validates, and use these axioms to give a categorical semantics for our comonadic type system. We then give an equational theory (substitution and the call-by-value β and η laws) for the imperative lambda calculus, and show its soundness relative to this semantics.

Finally, we give a translation of the pure simply-typed lambda calculus into our comonadic imperative calculus, and show that any two terms which are $\beta\eta$-equal in the STLC are equal in the equational theory of the comonadic calculus, establishing that pure programs can be mapped in an equation-preserving way into our imperative calculus.

1 INTRODUCTION

Consider the following definition of the familiar `map` functional.

```
map1 : ∀ a b. (a → b) → List a → List b
map1 f []        = []
map1 f (x :: xs) = let zs = map1 f xs in
                   let  z = f x in
                   z :: zs
```

This definition is the one that might be given in an introductory functional programming class — it recursively examines whether the list is nil or a cons and rebuilds the list, applying the function `f` each time. However, this definition is not ideally suited to be the implementation in a standard library, since it is *not tail-recursive.* As a result, one might be minded to replace it with the following "equivalent" definition:

```
map2 : ∀ a b. (a → b) → List a → List b
map2 f ys =
  let rec loop xs acc =
    match xs with
    | []      → List.reverse acc
    | x :: xs → loop xs (f x :: acc)
  in
  loop xs []
```

This version applies `f` in a tail-recursive loop, building up a reversed list of applications, and then reverses the list again before returning it to the client. This implementation allocates an intermediate list, but will never blow the stack.

2020. 2475-1421/2020/1-ART1 $15.00
https://doi.org/

However, in an *impure* functional language, it is not possible to transparently replace the first definition with the second. The difference between these two implementations is *observable*.

```
let xs : List String = ["left "; "to "; "right "]

let f : String → String = fun s → stdout.print(s); s

let zs1 = map1 f xs  -- Prints "right to left " to stdout
let zs2 = map2 f xs  -- Prints "left to right " to stdout
```

So something as innocuous-seeming as a `print` function can radically change the equational theory of the language: no program transformation that changes the order in which sub-expressions are evaluated is in general sound. This greatly complicates reasoning about programs, as well as hindering many desirable program optimisations such as list fusion and deforestation [Wadler 1990]. Transformations that are unconditionally valid in a pure language must, in an impure language, be gated by complex whole-program analyses tracking the purity of sub-expressions.

Contributions. It is received wisdom that much as a drop of ink cannot be removed from a glass of water, once a language supports ambient effects, there is no way to regain the full equational theory of a pure programming language. In this paper, we show that this folk belief is *false*: we extend an ambiently effectful language to support purity. Entertainingly, it turns out that just as monads are a good tool to extend pure languages with effects, **comonads** are a good tool to extend impure languages with purity!

- We take a pervasively effectful lambda calculus in the style of ML and show how to *extend* it with a *comonadic* type discipline that permits capturing pure expressions with types.
- We give a simple and intuitive denotational model for our language, which we call the *capability space* model. Our semantics is a formalisation of the intuition underpinning the *object-capability model* [Lauer and Needham 1979; Levy 1984; Miller 2006] familiar to systems designers, which says that the ability to perform effects should be controlled via access to a permission or capability, and that a program is *capability-safe* precisely when it can only perform effects that it possesses a runtime capability for.
  We do this by extending the most naive model of the lambda calculus – sets and functions – with just enough structure to model capability-safe programs. In our model, a type is just a set X (denoting a set of values), together with a function w saying which capabilities each value x owns. Then, a morphism $f : X \to Y$ is *capability-safe* if the capabilities of $f(x)$ are always bounded by the capabilities of x.
  It is already known in the systems community that effectful lambda-calculi without ambient authority are capability-safe. Our model demonstrates that this observation is incomplete – having a comonad witnessing the *denial* of a capability is also very beneficial.
- We then identify the axiomatic categorical structure the capability space model validates, and use these axioms to give a categorical semantics for our comonadic type system. We then give an equational theory (substitution and the call-by-value β and η laws) for the imperative lambda calculus, and show its soundness relative to this semantics.
- Finally, we give a translation of the pure simply-typed lambda calculus into our comonadic imperative calculus, and show that any two terms which are $\beta\eta$-equal in the STLC are equal in the equational theory of the comonadic calculus under the translation, establishing that pure programs can be mapped in an equation-preserving way into our imperative calculus.

Detailed proofs of the lemmas and theorems are given in the supplementary appendices.

2 RECOVERING PURITY BY EXAMPLE

In order to reason about purity in an ambiently effectful language, it is necessary to identify whether a program may have effects or not. This is a relatively straightforward task in a first-order language: we can decide whether a procedure may have effects by examining each subphrase of an expression and seeing if it either performs an effect, or calls a procedure which may perform effects. In this way, programs can be partitioned into those which are definitely pure, or those which may have effects. However, this distinction breaks down in a higher-order functional language. Consider again the example of the map functional:

```
map : ∀ a b. (a → b) → List a → List b
map f []        = []
map f (x :: xs) = f x :: map f xs
```

The expression `map g` is effectful, depending on whether the body of the function `g` has an effect or not. So if we want to ensure that calls to `map` are always pure, we have to ensure that it is always passed a pure function. An alternative way of expressing the issue is that, within the definition of `map`, there is a function-valued variable `f`, and we are free to substitute *any* function (including effectful ones) for `f`.

Therefore, we introduce **two kinds of variables:** pure variables and arbitrary (or impure) variables. This lets us define the notion of "pure term" in a simple and brutal fashion: we judge a pure term to be one which *both* performs no obvious effects, *and* all of whose free variables are themselves pure. Then, by restricting the substitution to only permit substituting pure terms for pure variables, the judgement of purity will be stable under substitution. Then, by internalising the purity judgement as a type, we can pass pure expressions around as first-class values.

To understand this, let us begin with a simple call-by-value higher-order functional language extended with types for string constants, channels (or output file handles), and a single effect: outputting a string onto a channel with `chan.print(s)`. There is no monadic or effect typing discipline here; the type of `print` is just as one might see in OCaml or Java.

```
print : Channel → String → Unit
```

For example, here is a simple function to print each element of a pair of strings to a given channel:

```
print_pair : String × String → Channel → Unit
print_pair = fun p chan →
                 chan.print(fst p);
                 chan.print(snd p)
```

Here, for clarity we use a semicolon for sequencing, and write `print` in method-invocation style *a la* Java (to make it easy to distinguish the file handle from the string argument).

To support purity, we extend the language with a new type constructor `Pure a`, denoting the set of expressions of type `a` which are *pure* – i.e., they own no file handles and so their execution cannot do any printing. So we add the introduction form `box(e)` to introduce a value whose type is `Pure a`; the type system accepts this if `e` has type `a` and is recognisably pure, but rejects it otherwise. Here, "recognisably pure" means that the term `e` has no syntactically obvious effects of its own, and all of its free variables are pure variables.

To eliminate a value of type `Pure a`, we will use *pattern matching*, writing the elimination form `let box(x) = e1 in e2` to bind the pure expression in `e1` to the variable `x`. The only difference from ordinary pattern matching is that `x` is marked as a pure variable, permitting it to occur inside of pure expressions. Intuitively, this makes sense – `e1` evaluates to a pure value, and so its result should be allowed to be used by other pure expressions.

We can see how these play out with the following examples, where we try to give a type for an `apply` function, which takes a function and an argument, applies the argument to the function, and returns the output, at varying levels of purity.

First, we consider a function that applies a pure argument to an unrestricted function:

```
apply : (String → Int) → Pure String → Int
apply f box(s) = f s  -- accepted
```

This example is accepted. The `box(s)` pattern tells us that `s` is a pure variable, but there are no restrictions on using pure variables as impure terms (since a pure term is an impure term that happens to not perform side-effects).

Next, we consider a variant of this function which applies an arbitrary function to a pure argument, and tries to return a pure result.

```
apply : (String → Int) → Pure String → Pure Int
apply f box(s) = box(f s)  -- REJECTED
```

This variant is rejected. Intuitively, the call to the function `f` could have side-effects. Syntactically, since `f` is an impure variable, it is simply not allowed to occur in the pure expression `box(f s)`. For similar reasons, it is not possible to write a polymorphic `fmap : ∀ a b. (a → b) → Pure a → Pure b` function for the `Pure` type constructor. However, `Pure` *is* a functor in the semantic sense – the absence of a map action indicates that this functor lacks *tensorial strength*.

We can still make both the function and the argument to `apply` into boxed types.

```
apply : Pure (String → Int) → Pure String → Pure Int
apply box(f) box(s) = box(f s)  -- accepted
```

In this case, `box(f s)` is accepted, since both the variables `f` and `s` are known to be pure, and so are permitted to occur inside of a pure expression.

Our type discipline also permits typing functions whose behaviour is intermediate between pure and effectful. For example, suppose that we see the following type declaration:

```
maybe_print : Pure (Maybe Channel → String)
-- definition not visible
```

We do not know anything about the body of the definition, but due to the typing discipline, we know that `maybe_print` owns no capabilities of its own. As a result, we can make some inferences when we see the following two declarations:

```
x : String
x = let Box(f) = maybe_print in
    f (Some stdout)

y : String
y = let Box(f) = maybe_print in
    f None
```

The definition of `x` passes a channel to `maybe_print`, and so it may have an effect (it might use it to print). On the other hand, we *know* that the evaluation of `y` *will not* have an effect – we know that `maybe_print` owned no channels, and since we did not give it a channel, it can therefore perform no effects. Moreover, we know this without having to see the definition of `maybe_print`!

Types	A, B	$::=$	$\mathsf{unit} \mid A \times B \mid A \Rightarrow B \mid \mathsf{str} \mid \mathsf{cap} \mid \square A$
Terms	e	$::=$	$() \mid (e_1, e_2) \mid \mathsf{fst}\, e \mid \mathsf{snd}\, e \mid x \mid \lambda x : A.\ e \mid e_1\, e_2$
		$\mid$	$s \mid \mathsf{box}\, \boxed{e} \mid \mathsf{let\ box}\, \boxed{x} = e_1\ \mathsf{in}\ e_2 \mid e_1 \cdot \mathsf{print}(e_2)$
Values	v	$::=$	$x \mid () \mid (v_1, v_2) \mid \lambda x : A.\ e \mid s \mid \mathsf{box}\, \boxed{e}$
Qualifiers	q, r	$::=$	$\boldsymbol{p} \mid \boldsymbol{i}$
Contexts	Γ, Δ, Ψ	$::=$	$\cdot \mid \Gamma, x : A^q$
Substitutions	θ, ϕ	$::=$	$\langle\rangle \mid \langle \theta, e^q / x \rangle$

Fig. 1. Grammar

In the next two sections, we will see that this discipline of tracking whether a variable is pure or not is precisely a *comonadic* type discipline, corresponding to the $\square$ modality in S4 modal logic, and that the model arises from a formalisation of object capabilities.

3 TYPING

We give the grammar of our language in figure 1.

We have the usual type constructors for unit, products, and functions from the simply-typed lambda calculus. In addition to this, we have the type str for strings, and the type cap representing output channels (used in the imperative $e_1 \cdot \mathsf{print}(e_2)$ statement). Finally, we add the comonadic $\square$ type constructor which corresponds to the **Pure** type constructor we introduced in section 2.

Despite the fact that there is a *type* cap of channels, and a print operation which uses them, there are no introduction forms for them. This is intentional! The absence of this facility corresponds to the principle of *capability safety* – the only capabilities a program should possess are those that are passed by its caller. So, a complete program will either be a function that receives a capability token as an argument, or have free variables that the system can bind capability tokens to. [1]

The expressions in our language include the usual ones from the simply-typed lambda calculus, constants s for strings, and print. We also have an introduction form box $\boxed{e}$, and a let box elimination form for the $\square A$ type; we'll explain how these work later. Values are a subset of expressions, but box turns any expression into a value. [2]

We would like a modal type system where we can distinguish between expressions with and without side-effects. Following the style of [Pfenning and Davies 2001] for S4 modal logic, we could build a dual-context calculus. However, such a setup makes it difficult to define substitution; we can avoid dual contexts by tagging terms with qualifiers instead. We use two qualifiers that we can annotate terms with, in the appropriate places. We use $\boldsymbol{p}$ to tag *pure* terms, and $\boldsymbol{i}$ to tag *impure* terms. [3]

Next, we define contexts of variables. A well-formed context is either the empty context $\cdot$, or an extended context with a variable x of type A and qualifier q. Finally, we give a grammar for substitutions. A substitution is either the empty substitution $\langle\rangle$, or an extended substitution with an expression e substituted for variable x qualified by q. [4]

---

[1] Of course, a full system should have the ability to create new private capabilities of its own. We omit this to keep the denotational semantics simple, but discuss how to add it in section 8.

[2] We sometimes use the expression form $e_1 ; e_2$, which is just syntactic sugar for $(\lambda x : \mathsf{unit}.\ e_2)\ e_1$.

[3] We use different colours to distinguish *pure* and *impure* syntactic objects, and we'll follow this convention henceforth.

[4] When we have unknown qualifiers occurring on terms, we *highlight* them in a different colour, and the colour changes to the appropriate one when the qualifier is $\boldsymbol{p}$ or $\boldsymbol{i}$.

$$\begin{array}{rl} x : A^q \in \Gamma & x \text{ is a variable of type } A \text{ with qualifier } q \text{ in context } \Gamma \\ \Gamma \vdash e : A & e \text{ is an expression of type } A \text{ in context } \Gamma \\ \Gamma \vdash^{p} e : A & e \text{ is a } \textit{pure} \text{ expression of type } A \text{ in context } \Gamma \end{array}$$

Fig. 2. Typing Judgements

$$\frac{}{\Gamma \vdash () : \mathsf{unit}}\ \text{unitI} \qquad \frac{}{\Gamma \vdash s : \mathsf{str}}\ \text{strI}$$

$$\frac{\Gamma \vdash e_1 : A \qquad \Gamma \vdash e_2 : B}{\Gamma \vdash (e_1, e_2) : A \times B}\ \times\text{I} \qquad \frac{\Gamma \vdash e : A \times B}{\Gamma \vdash \mathsf{fst}\, e : A}\ \times\text{E}_1 \qquad \frac{\Gamma \vdash e : A \times B}{\Gamma \vdash \mathsf{snd}\, e : B}\ \times\text{E}_2$$

$$\frac{x : A^q \in \Gamma}{\Gamma \vdash x : A}\ \textsc{Var} \qquad \frac{\Gamma, x : A^i \vdash e : B}{\Gamma \vdash \lambda x : A.\, e : A \Rightarrow B}\ \Rightarrow\text{I} \qquad \frac{\Gamma \vdash e_1 : A \Rightarrow B \qquad \Gamma \vdash e_2 : A}{\Gamma \vdash e_1\, e_2 : B}\ \Rightarrow\text{E}$$

$$\frac{\Gamma \vdash e_1 : \mathsf{cap} \qquad \Gamma \vdash e_2 : \mathsf{str}}{\Gamma \vdash e_1 \cdot \mathsf{print}(e_2) : \mathsf{unit}}\ \textsc{Print}$$

$$\frac{\Gamma^{p} \vdash e : A}{\Gamma \vdash^{p} e : A}\ \textsc{ctx-pure} \qquad \frac{\Gamma \vdash^{p} e : A}{\Gamma \vdash \mathsf{box}\, \boxed{e} : \square A}\ \square\text{I} \qquad \frac{\Gamma \vdash e_1 : \square A \qquad \Gamma, x : A^{p} \vdash e_2 : B}{\Gamma \vdash \mathsf{let\,box}\, \boxed{x} = e_1\ \mathsf{in}\, e_2 : B}\ \square\text{E}$$

Fig. 3. Typing Rules

$$\frac{}{x : A^q \in (\Gamma, x : A^q)}\ \in\text{-id} \qquad \frac{x : A^q \in \Gamma \qquad (x \neq y)}{x : A^q \in (\Gamma, y : B^r)}\ \in\text{-ex}$$

Fig. 4. Context Membership Rules

3.1 Typing Judgements

We introduce three kinds of judgement forms, as explained in figure 2, and we state our typing rules in figure 3, which we explain below.

We give the standard rules for the context membership judgement in figure 4, following Barendregt's variable convention [Barendregt 1985]. The only difference is that variables now have an extra purity annotation.

We have the usual introduction and elimination rules for constants and products. If a variable is present in the context, we can introduce it, using the Var rule. In the introduction rule for functions ⇒ I, we mark the hypothesis as *impure* when forming a λ-expression, because we do not want to restrict function arguments in general. The elimination rule ⇒ E, or function application works as usual. The print statement performs side-effects but has the type unit, as we've already seen. We need to do more work to add the comonadic type constructor.

$$
\begin{aligned}
(\cdot)^{p} &:= \cdot \\
(\Gamma, x : A^{p})^{p} &:= \Gamma^{p}, x : A^{p} \\
(\Gamma, x : A^{i})^{p} &:= \Gamma^{p}
\end{aligned}
$$

(a) Purify Operation on Contexts

$$
\begin{aligned}
\langle\rangle^{p} &:= \langle\rangle \\
\langle\theta, e^{p}/x\rangle^{p} &:= \langle\theta^{p}, e^{p}/x\rangle \\
\langle\theta, e^{i}/x\rangle^{p} &:= \theta^{p}
\end{aligned}
$$

(b) Purify Operation on Substitutions

Fig. 5. Purify Operations

$$
\begin{array}{rl}
\Gamma \supseteq \Delta & \Gamma \text{ is a weakening of context } \Delta \\
\Gamma \vdash \theta : \Delta & \theta \text{ is a well-formed substitution from context } \Gamma \text{ to } \Delta
\end{array}
$$

Fig. 6. Weakening and Substitution Judgements

We know that we can mark a term as *pure* if it was well-typed in a *pure* context, where every variable has the $\boldsymbol{p}$ annotation. So we define a syntactic *purify* operation, which acts on contexts; applying it drops the terms with the *impure* annotation, as shown in figure 5a. This is expressed by the CTX-PURE rule, which introduces a *pure* expression using the *pure* judgement form. And then, we can put it in a box using the $\square$I rule, to get a $\square$-typed value.

We give an elimination rule $\square$E using the let box binding form. Given an expression in the $\square$ type, we let box-bind the underlying *pure* expression to the variable x. With an extended context that has a free variable x marked *pure*, if we can produce a well-typed expression in the motive, the elimination is complete.

3.2 Weakening and Substitution

We define two more judgement forms for weakening and substitution; these are meta-theoretic operations which are only used to state and prove meta-theoretic properties of the language. Note that we *do not* use explicit substitutions, i.e., substitutions do not appear as part of expressions.

3.2.1 Weakening. The context weakening relation follows the usual rules, as shown in figure 7a, with the extra purity annotation on free variables in contexts. The rule $\supseteq$ -WK allows us to drop a hypothesis to weaken the context, and we add the rules $\supseteq$ -ID and $\supseteq$ -CONG to get the smallest congruence closure.

We show that weakening is sound by proving a syntactic weakening lemma.

LEMMA 3.1 SYNTACTIC WEAKENING. *If $\Gamma \supseteq \Delta$ and $\Delta \vdash e : A$, then $\Gamma \vdash e : A$.*

3.2.2 Substitution. Substitution requires an extra bit of work, as we can see in figure 7b. Since our language is effectful, we have the usual rule SUB-IMPURE which allows substituting *values* for *impure* variables, as in the call-by-value lambda calculus. We also add another rule SUB-PURE, which allows one to substitute *pure expressions* for *pure* variables.

At this point, we can define the syntactic substitution function on raw terms.

Definition 3.2 (Syntactic substitution on variables).

$$
\theta[x] \quad := \quad \begin{cases} \lightning & \theta = \langle\rangle \\ e & \theta = \langle\phi, e^{q}/x\rangle \\ \phi[x] & \theta = \langle\phi, e^{q}/y\rangle, x \neq y \end{cases}
$$

$$\frac{}{\cdot \supseteq \cdot}\ \supseteq\text{-ID} \qquad \frac{\Gamma \supseteq \Delta}{\Gamma, x : A^q \supseteq \Delta, x : A^q}\ \supseteq\text{-CONG} \qquad \frac{\Gamma \supseteq \Delta}{\Gamma, x : A^q \supseteq \Delta}\ \supseteq\text{-WK}$$

(a) Weakening Rules

$$\frac{}{\Gamma \vdash \langle\rangle : \cdot}\ \text{SUB-ID}$$

$$\frac{\Gamma \vdash \theta : \Delta \qquad \Gamma \vdash^{p} e : A}{\Gamma \vdash \langle \theta, e^{p}/x \rangle : \Delta, x : A^{p}}\ \text{SUB-PURE} \qquad \frac{\Gamma \vdash \theta : \Delta \qquad \Gamma \vdash v : A}{\Gamma \vdash \langle \theta, v^{i}/x \rangle : \Delta, x : A^{i}}\ \text{SUB-IMPURE}$$

(b) Substitution Rules

Fig. 7. Weakening and Substitution Rules

Definition 3.3 (Syntactic substitution on terms).

$$\begin{array}{rcl}
\theta(x) & := & \theta[x] \\
\theta(()) & := & () \\
\theta(s) & := & s \\
\theta((e_1\,, e_2)) & := & (\theta(e_1)\,, \theta(e_2)) \\
\theta(\mathsf{fst}\, e) & := & \mathsf{fst}\, \theta(e) \\
\theta(\mathsf{snd}\, e) & := & \mathsf{snd}\, \theta(e) \\
\theta(\lambda x.\, e) & := & \lambda y.\, \langle \theta, y^{i}/x \rangle(e) \\
\theta(e_1\, e_2) & := & \theta(e_1)\, \theta(e_2) \\
\theta(\mathsf{box}\, \boxed{e}) & := & \mathsf{box}\, \boxed{\theta^{p}(e)} \\
\theta(\mathsf{let\ box}\, \boxed{x} = e_1\ \mathsf{in}\, e_2) & := & \mathsf{let\ box}\, \boxed{y} = \theta(e_1)\ \mathsf{in}\ \langle \theta, y^{p}/x \rangle(e_2) \\
\theta(e_1 \cdot \mathsf{print}(e_2)) & := & \theta(e_1) \cdot \mathsf{print}(\theta(e_2))
\end{array}$$

When substituting under a binder, we do a renaming of the bound variable by extending the substitution with an appropriately annotated variable. To substitute inside a box-ed expression, we have to *purify* the substitution when using it. We extend the *purify* operation to substitutions as well; it simply drops the *impure* substitutions, as shown in figure 5b.

Finally, we show the soundness of substitution by proving a syntactic substitution theorem.

Theorem 3.4 Syntactic substitution. *If* $\Gamma \vdash \theta : \Delta$ *and* $\Delta \vdash e : A$, *then* $\Gamma \vdash \theta(e) : A$.

4 SEMANTICS

In this section, we sketch a categorical semantics for our language, motivated by an abstract model of capabilities.

4.1 The Object-Capability Model

The *object-capability* model is a methodology originating in the operating systems community for building secure operating systems and hardware. The idea behind this model is that systems must be able to control permissions to perform potentially dangerous or insecure operations, and that a good way to control access is to tie the right to perform actions to values in a programming language, dubbed *capabilities*. Then, the usual variable-binding and parameter-passing mechanisms of the language can be used to grant rights to perform actions — access to a capability can be prohibited to a client by simply not passing it the capability as an argument. To quote Miller [2006]:

> Our object-capability model is essentially the untyped call-by-value lambda calculus with applicative-order local side effects and a restricted form of **eval** — the model Actors and Scheme are based on. This correspondence of objects, lambda calculus, and capabilities was noticed several times by 1973.

In our kernel language from the previous section, the potentially dangerous operation that must be controlled is the right to print to a particular channel, and so we take channels as capabilities. The $c \cdot \mathsf{print}(s)$ operation takes the channel c and prints the string s to it. We can see here how the print operation uses the channel value to select the channel to print on — in this case, the output channel is the capability. Of course, program values can possess multiple capabilities – for example, a list of channels naturally has a capability for each channel in the list, and a closure can capture channels to perform print actions on. Nevertheless, though, there is no way for a function to print on a channel that it did not either capture in its environment, or receive as an argument.

This property is actually fundamental to the object-capability model, which says that the *only* way to access capabilities must be through capability values. If this is indeed the case, then the language is said to be *capability-safe*. However, if there are ways to conjure up capabilities out of nowhere (e.g., unrestricted filesystem operations in the standard library, or more alarmingly by casting integers to pointers in C), then reasoning about effects based on capability passing is not sound. In this case, the language is said to possess *ambient authority*.

4.2 Capability Spaces

Let C be a fixed set of capability names, possibly countably infinite. We require that C have decidable equality. The powerset $\wp(C)$ denotes the set of all subsets of C, and is a complete lattice ordered by set inclusion $(\wp(C); \emptyset, C, \subseteq)$.

A capability space $X = (|X|, w_X)$ is a set $|X|$ with a weight function $w_X : |X| \to \wp(C)$ that assigns a set of capabilities to each member in X. Intuitively, we think of the set $|X|$ as the set of values of the type X, and we think of the weight function w_X as defining the set of capabilities that each value has access to.

We only allow those maps between capability spaces that preserve weights, i.e., a map between the underlying sets $|X|$ and $|Y|$ is a morphism of capability spaces iff for each x in $|X|$, all the weights in Y for $f(x)$ are contained in the weights in X for x. If we think of a function $f : X \to Y$ as a term of type Y with a free variable of type X, then this condition ensures that the capabilities of the term are limited to at most those of its free variables. In other words, weight-preserving functions are precisely those which are capability-safe; they do not have unauthorised access to arbitrary capabilities, and they do not have any ambient authority.

We now formally define the category of capability spaces $\mathcal{C}$, with objects as capability spaces and morphisms as weight-preserving functions.

Definition 4.1 (Category $\mathcal{C}$ of capability spaces).

$$\begin{array}{rcl} Obj_{\mathcal{C}} & := & X = (|X| : \mathrm{Set},\ w_X : |X| \to \wp(C)) \\ \mathcal{H}om_{\mathcal{C}}(X, Y) & := & \left\{ f \in |X| \to |Y| \;\middle|\; \forall x \in |X|,\ w_Y(f(x)) \subseteq w_X(x) \right\} \end{array}$$

We remark that the definition of this category is inspired by the category of length spaces defined in [Hofmann 2003], which again associates intensional information (in his work, memory usage, and in ours, capabilities) to a set-theoretic semantics.

4.3 Cartesian Closed Structure

We now explain the *cartesian closed* structure of $\mathcal{C}$.

Definition 4.2 (Terminal Object).

$$\begin{array}{rcl} |1| & := & \{ * \} \\ w_1(*) & := & \emptyset \end{array}$$

The terminal object is the usual singleton set, and it has no capabilities. For any object A, the unique map $! : A \to 1$ is given by $!_A(a) = *$, which is evidently weight preserving.

Definition 4.3 (Product).

$$\begin{array}{rcl} |A \times B| & := & |A| \times |B| \\ w_{A\times B}(a, b) & := & w_A(a) \cup w_B(b) \end{array}$$

Products are formed by pairing as usual, and the set of capabilities of a pair of values is the union of their capabilities. The projection maps $\pi_i : A_1 \times A_2 \to A_i$ are just the projections on the underlying sets, which are weight preserving as well.

Definition 4.4 (Exponential).

$$\begin{array}{rcl} |A \to B| & := & |A| \to |B| \\ w_{A\to B}(f) & := & \left\{ c \in C \;\middle|\; \begin{array}{l} \exists a \in |A|, \\ c \in w_B(f(a)), \\ c \notin w_A(a) \end{array} \right\} \end{array}$$

Exponentials are given by functions on the underlying sets, but we have to assign capabilities to the closure. We only record those capabilities which are induced by the function, for some value in the domain. The intuition is that if we have a function closure $f : A \to B$, and for a given value $a \in A$, there is a capability c such that $c \notin w_B(f(a))$, then the closure f must have had access to c in its environment. So by taking the union of all such c over all inputs in the domain, we can bound all the capabilities that f must have access to.

We verify that our definition satisfies the currying isomorphism in lemma 4.5, and we name the currying/uncurrying and evaluation maps. The definitions are the same as in the case of sets, but we additionally have to verify that these maps are weight-preserving.

Lemma 4.5.

$$\begin{array}{rcl} \mathsf{curry/uncurry} & : & \mathcal{H}om_{\mathcal{C}}(\Gamma \times A, B) \xrightarrow{\sim} \mathcal{H}om_{\mathcal{C}}(\Gamma, A \to B) \\ \mathsf{ev}_{A,B} & : & \mathcal{H}om_{\mathcal{C}}(A \to B \times A, B) \end{array}$$

This shows that $\mathcal{C}$ has finite products and exponentials, and is hence a cartesian closed category, which suffices to interpret the simply-typed lambda calculus.

4.4 Monad

Our language supports printing strings along a channel, and to model this effect we will structure our semantics monadically, in the style of Moggi [1991]. To model the print effect, we define a strong monad T on $\mathcal{C}$ as follows, taking the monoid $(\Sigma^*; \varepsilon, \bullet)$ to be the set of strings Σ^* with the empty string ε and string concatenation $\bullet$.

Definition 4.6 ($T : \mathcal{C} \longrightarrow \mathcal{C}$).

$$
\begin{array}{rcl}
|T(A)| & := & |A| \times (C \to \Sigma^*) \\
w_{T(A)}(a, o) & := & w_A(a) \cup \left\{ c \in C \,\middle|\, o(c) \neq \varepsilon \right\}
\end{array}
$$

This monad is essentially the writer monad: it adds an output function which records the output produced in each channel. The weight of a monadic computation is taken to be the weight of the returned value, unioned with all the channels that *anything* was written to. This corresponds to the intuition that a computation which performs I/O on a channel must possess the capability to do so.

Definition 4.7 (T is a monad). The unit and multiplication of the monad are defined below, and we state and verify the monad laws in lemma B.1.

$$
\begin{array}{rcl}
\eta_A : A & \to & TA \\
a & \mapsto & (a, \lambda c.\varepsilon) \\
\mu_A : TTA & \to & TA \\
((a, o_1), o_2) & \mapsto & (a, \lambda c.o_2(c) \bullet o_1(c))
\end{array}
$$

Definition 4.8 (T is a strong monad). T is strong with respect to products, with a natural family of left and right strengthening maps.

$$
\begin{array}{rcl}
\tau_{A,B} : A \times TB & \to & T(A \times B) \\
(a, (b, o)) & \mapsto & ((a, b), o) \\
\sigma_{A,B} : TA \times B & \to & T(A \times B) \\
((a, o), b) & \mapsto & ((a, b), o)
\end{array}
$$

We use this to define the natural map $\beta_{A,B}$, which evaluates a pair of effects, as follows. Notice that it evaluates the effect on the right before the one on the left; we expand more on that in lemma B.2, and verify the appropriate coherences.

Definition 4.9 ($\beta_{A,B} : TA \times TB \to T(A \times B)$).

$$
\beta_{A,B} \quad := \quad \tau_{TA,B} \,;\, T\sigma_{A,B} \,;\, \mu_{A \times B}
$$

4.5 Comonad

To model the $\square$ type constructor, we define an endofunctor $\square$ on $\mathcal{C}$ below; it filters out values that *do not* possess any capabilities, i.e., values that are *pure*.

Definition 4.10 ($\square : \mathcal{C} \longrightarrow \mathcal{C}$).

$$
\begin{array}{rcl}
|\square A| & := & \left\{ a \in |A| \,\middle|\, w_A(a) = \emptyset \right\} \\
w_{\square A}(a) & := & w_A(a) = \emptyset \\
\square : \mathcal{H}om_{\mathcal{C}}(A\,, B) & \to & \mathcal{H}om_{\mathcal{C}}(\square A\,, \square B) \\
f & \mapsto & f \restriction_{|\square A|}
\end{array}
$$

On objects, we simply restrict the set to the subset of values that have the empty set $\emptyset$ of capabilities. $\square$ acts on morphisms by restricting the domain of the functions to $|\square A|$. For any morphism f, since f is a weight-preserving function, we have that $\square(f)$ is a function between sets with empty capabilities, hence it becomes trivially weight-preserving.

This type constructor is especially useful at function type $\Box(A \to B)$, since in general the environment can hold capabilities, and the $\Box$ constructor lets us rule those out. We claim that $\Box$ is an idempotent strong monoidal comonad, as follows.

Definition 4.11 ($\Box$ is an idempotent comonad). The counit and comultiplication of the comonad are the natural families of maps given by the inclusion and the identity maps on the underlying set. δ is a natural isomorphism making it idempotent. We state and verify the comonad laws in lemma B.3.

$$\begin{array}{rcl} \varepsilon_A : \Box A & \to & A \\ a & \mapsto & a \\ \delta_A : \Box A & \xrightarrow{\sim} & \Box\Box A \\ a & \mapsto & a \end{array}$$

Definition 4.12 ($\Box$ is a strong monoidal functor). The functor is strong monoidal, in the sense that it preserves the monoidal structure of both products (and tensors, see the sequel in subsection 4.7). The identity element is preserved, and we have *natural isomorphisms* given by pairing on the underlying sets.

$$\begin{array}{rcl} m^I : 1 & \xrightarrow{\sim} & \Box 1 \\ * & \mapsto & * \\ m^{\times}_{A,B} : (\Box A \times \Box B) & \xrightarrow{\sim} & \Box(A \times B) \\ (a\,,b) & \mapsto & (a\,,b) \\ m^{\otimes}_{A,B} : (\Box A \otimes \Box B) & \xrightarrow{\sim} & \Box(A \otimes B) \\ (a\,,b) & \mapsto & (a\,,b) \end{array}$$

We remark that $\Box$ is not a strong comonad, i.e., it does not possess a tensorial strength. This makes it impossible to evaluate an arbitrary function under the comonad, as seen in section 2. [5]

4.6 The Comonad cancels the Monad

Finally, we make the following observation. There is an isomorphism ϕ_A, natural in A, where the comonad cancels the monad. In programming terms, this says that *an effectful computation with no capabilities can perform no effects* — i.e., it is *pure*. Note that this definition works because of the particular definition of the monad T we chose, in which the weight of a computation includes all the channels it printed on. Consequently computation of weight zero cannot print on any channel, and so must be *pure*! As usual, we verify this fact in lemma B.4.

Definition 4.13 ($\phi : \Box T \Rightarrow \Box$).

$$\begin{array}{rcl} \phi_A : \Box T A & \xrightarrow{\sim} & \Box A \\ (a, o) & \mapsto & a \end{array}$$

This property is crucial and we will exploit this to manage our syntax: it will be how we justify treating terms in *pure* contexts as *pure*, without needing a second grammar for *pure* expressions.

[5]For Haskellers, the $\Box$ functor is not a `Functor`, but it is an `Applicative`!

4.7 Monoidal Closed Structure

While the monad and comonad, together with the cartesian closed structure, suffice to interpret our language, it is worth noting that the category $\mathcal{C}$ also admits a *monoidal closed* structure.

Definition 4.14 (Tensor product).

$$\begin{array}{rcl} |A \otimes B| & := & \{ (a,b) \in |A| \times |B| \mid w_A(a) \cap w_B(b) = \emptyset \} \\ w_{A \otimes B}(a,b) & := & w_A(a) \cup w_A(b) \end{array}$$

The tensor product is given by pairing, with unit 1, but it only restricts to pairs whose sets of capabilities are disjoint. But, this tensor product also enjoys a right adjoint.

Definition 4.15 (Linear exponential).

$$|A \multimap B| \quad := \quad \left\{ f \in |A| \to |B| \;\middle|\; \begin{array}{l} \exists C \in \wp(\mathcal{C}), \forall a \in |A|, \\ C \cap w_A(a) = \emptyset \Rightarrow w_B(f(a)) \subseteq C \cup w_A(a) \end{array} \right\}$$

$$w_{A \to B}(f) \quad := \quad \left\{ c \in \mathcal{C} \;\middle|\; \begin{array}{l} \exists a \in |A|, \\ c \in w_B(f(a)), \\ c \notin w_A(a) \end{array} \right\}$$

The linear exponential works the same way as the exponential, except that we have to restrict it to satisfy the disjointness condition for the tensor product. We verify that this definition satisfies the tensor-hom adjunction in lemma 4.16.

LEMMA 4.16.

$$\mathcal{H}om_{\mathcal{C}}(\Gamma \otimes A\,, B) \cong \mathcal{H}om_{\mathcal{C}}(\Gamma\,, A \multimap B)$$

This supports an interpretation of a *linear* (actually, affine) type theory. The disjointness conditions in the interpretation of tensor product and linear implication are essentially the same as the disjointness conditions in the definition of the separating conjunction $A * B$ and magic wand $A \mathrel{-\!\!*} B$ in separation logic [Reynolds 2002]. In separation logic, capabilities correspond to ownership of particular memory locations, and in our setting, capabilities correspond to the right to access a channel.

Our model reassuringly suggests that operating systems researchers and program verification researchers both identified the same notion of capability. However, it seems that the fact that these are *exactly* the same idea was overlooked because OS researchers focused on the cartesian closed structure, and semanticists focused on the monoidal closed structure!

5 INTERPRETATION

We now interpret the syntax of our language. An important point to note here is that, we only use the algebraic structure of the category, i.e., we use the *cartesian closed* structure, the *monoidal idempotent comonad*, the *strong monad*, and the *cancellation isomorphism* ϕ; the proofs of our theorems use the universal property for each categorical construction. We only need to use the definition of the monad in the interpretation of print. [6]

We adopt some standard notation to work with our categorical combinators. [7] The sequential composition of two arrows, in the diagrammatic order, is $f \,;\, g$. The product of morphisms f and g

[6]Our results will still hold if we switched to another category with this structure, we say more about that in section 8.

[7]We sometimes drop the denotation symbol for brevity, i.e., we write $!_\Gamma$ instead of $!_{[\![\Gamma]\!]}$, or δ_{Γ^p} instead of $\delta_{[\![\Gamma^p]\!]}$.

$$\begin{tikzcd} A \arrow[r, "f"'] \arrow[rr, bend left, "f\,;g"] & B \arrow[r, "g"'] & C \end{tikzcd}$$

$$\begin{tikzcd} & A \arrow[dl, "f"'] \arrow[d, "\langle f\,,g\rangle"] \arrow[dr, "g"] & \\ B & B \times C \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] & C \end{tikzcd} \qquad \begin{tikzcd} A \arrow[d, "f"'] & A \times C \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, "{[f \times g]}"] \arrow[dl, "\pi_1;f"'] \arrow[dr, "\pi_2;g"] & C \arrow[d, "g"] \\ B & B \times D \arrow[l, "\pi_1"] \arrow[r, "\pi_2"'] & D \end{tikzcd}$$

Fig. 8. Composition operations

is $\langle f\,,g\rangle$ (also called a fork operation in the algebra of programming community [Gibbons 2000]), and $[f \times g]$ is parallel composition with products. We define these using the universal property of products and composition, as shown in figure 8.

5.1 Types and Contexts

We interpret types as objects in $\mathcal{C}$. Note that we use the monad in the interpretation of functions, following the call-by-value computational lambda-calculus interpretation in [Moggi 1989]. We use the comonad to interpret the $\boxed{}$ modality. We pick particular sets Σ^* and C to interpret strings and capabilities respectively.

Definition 5.1 ($[\![A]\!] : Obj_{\mathcal{C}}$).

$$\begin{array}{rcl} [\![\mathsf{unit}]\!] & := & 1 \\ [\![\mathsf{str}]\!] & := & \Sigma^* \\ [\![\mathsf{cap}]\!] & := & C \\ [\![A \times B]\!] & := & [\![A]\!] \times [\![B]\!] \\ [\![A \Rightarrow B]\!] & := & [\![A]\!] \to T[\![B]\!] \\ [\![\boxed{}A]\!] & := & \square[\![A]\!] \end{array}$$

We interpret contexts as finite products of objects. The comonad is used to interpret the *pure* variables in the context, while the *impure* variables are just arbitrary objects in $\mathcal{C}$.

Definition 5.2 ($[\![\Gamma]\!] : Obj_{\mathcal{C}}$).

$$\begin{array}{rcl} [\![\cdot]\!] & := & 1 \\ [\![\Gamma, x : A^{\mathbf{p}}]\!] & := & [\![\Gamma]\!] \times \square[\![A]\!] \\ [\![\Gamma, x : A^{\mathbf{i}}]\!] & := & [\![\Gamma]\!] \times [\![A]\!] \end{array}$$

Now we give an interpretation for the context membership relation. [8] The judgement $x : A^q \in \Gamma$ is interpreted as a morphism in $\mathcal{H}om_{\mathcal{C}}(\,[\![\Gamma]\!]\,,[\![A]\!]\,)$. It projects out the appropriately typed and annotated variable from the product in the context. For *pure* variables, we need to use the counit ε to get out of the comonad.

[8] When interpreting judgements and inference rules, we write $[\![\dfrac{\mathcal{J}_1 \cdots \mathcal{J}_n}{\mathcal{J}}]\!]$ to mean the interpretation of $\mathcal{J}$, i.e., we recursively define $[\![\mathcal{J}]\!]$ under the assumption that we have an interpretation for $\mathcal{J}_i$, i.e., $[\![\mathcal{J}_1]\!], \ldots, [\![\mathcal{J}_n]\!]$.

Definition 5.3 ($[\![x : A^q \in \Gamma]\!] : \mathcal{H}om_{\mathcal{C}}([\![\Gamma]\!], [\![A]\!])$).

$$
\begin{aligned}
[\![\frac{}{x : A^i \in (\Gamma, x : A^i)}]\!] &:= \pi_2 \\
[\![\frac{}{x : A^p \in (\Gamma, x : A^p)}]\!] &:= \pi_2 \,;\, \varepsilon_A \\
[\![\frac{x : A^q \in \Gamma \qquad (x \neq y)}{x : A^q \in (\Gamma, y : B^r)}]\!] &:= \pi_1 \,;\, [\![x : A^q \in \Gamma]\!]
\end{aligned}
$$

5.2 Expressions

We now give an interpretation for expressions $\Gamma \vdash e : A$, and *pure* expressions $\Gamma \vdash^p e : A$. We interpret each typing rule as follows.

Definition 5.4 ($[\![\Gamma \vdash e : A]\!] : \mathcal{H}om_{\mathcal{C}}([\![\Gamma]\!], T[\![A]\!])$, $[\![\Gamma \vdash^p e : A]\!]_p : \mathcal{H}om_{\mathcal{C}}([\![\Gamma]\!], \Box[\![A]\!])$).

$$
[\![\frac{}{\Gamma \vdash () : \mathsf{unit}}]\!] := !_\Gamma \,;\, \eta_1
$$

To interpret unitI, we use the unique ! map to simply get to the terminal object 1, then lift it into the monad using η, without performing any effects.

$$
[\![\frac{\Gamma \vdash e_1 : A \qquad \Gamma \vdash e_2 : B}{\Gamma \vdash (e_1, e_2) : A \times B}]\!] := \begin{array}{l} \mathit{let} \begin{cases} f := [\![\Gamma \vdash e_1 : A]\!] \\ g := [\![\Gamma \vdash e_2 : B]\!] \end{cases} \\ \mathit{in} \;\; \langle f, g\rangle \,;\, \beta_{A,B} \end{array}
$$

$$
\Gamma \xrightarrow{\langle f, g\rangle} TA \times TB \xrightarrow{\beta_{A,B}} T(A \times B)
$$

For pair introduction $\times$I, we evaluate both components of the pair, and compose, then use the strength of the monad T with the β combinator to form the product. [9]

$$
\begin{aligned}
[\![\frac{\Gamma \vdash e : A \times B}{\Gamma \vdash \mathsf{fst}\, e : A}]\!] &:= [\![\Gamma \vdash e : A \times B]\!] \,;\, T\pi_1 \\
[\![\frac{\Gamma \vdash e : A \times B}{\Gamma \vdash \mathsf{snd}\, e : B}]\!] &:= [\![\Gamma \vdash e : A \times B]\!] \,;\, T\pi_2
\end{aligned}
$$

We eliminate products using the $\times E_1$ and $\times E_2$ rules. These are interpreted using the corresponding product projection maps, under the functorial action of T.

$$
[\![\frac{x : A^q \in \Gamma}{\Gamma \vdash x : A}]\!] := [\![x : A^q \in \Gamma]\!] \,;\, \eta_A
$$

Variables are introduced using the Var rule, which is interpreted by looking up in the context, for which we use the interpretation of our context membership judgement. This is followed by a trivial lifting into the monad.

[9] The vigilant reader will have noticed that β evaluates the pair from right to left, so the action on the right will be performed first, like OCaml! This is also useful when interpreting function application, because we evaluate the argument first.

$$\left[\!\!\left[\frac{\Gamma, x : A^{i} \vdash e : B}{\Gamma \vdash \lambda x : A.\, e : A \Rightarrow B} \right]\!\!\right] \quad := \quad \mathsf{curry}\,(\llbracket \Gamma, x : A^{i} \vdash e : B \rrbracket)\,;\,\eta_{A \to TB}$$

To interpret functions using the $\Rightarrow$ I rule, we simply use the currying map, since our context extension is interpreted as a product. Then we lift it into the monad using η.

$$\left[\!\!\left[\frac{\Gamma \vdash e_1 : A \Rightarrow B \qquad \Gamma \vdash e_2 : A}{\Gamma \vdash e_1\, e_2 : B} \right]\!\!\right] \quad := \quad \begin{array}{ll} \mathit{let} & \left\{ \begin{array}{lll} f & := & \llbracket \Gamma \vdash e_1 : A \Rightarrow B \rrbracket \\ g & := & \llbracket \Gamma \vdash e_2 : A \rrbracket \end{array} \right. \\ \mathit{in} & \langle f, g \rangle\,;\,\beta_{A \to TB, A}\,;\,T\,\mathsf{ev}_{A,TB}\,;\,\mu_B \end{array}$$

$$\Gamma \xrightarrow{\langle f, g \rangle} T(A \to TB) \times TA \xrightarrow{\beta_{A \to TB, A}} T(A \to TB \times A) \xrightarrow{T\,\mathsf{ev}_{A,TB}} T^2 B \xrightarrow{\mu_B} TB$$

To eliminate functions using the $\Rightarrow$ E rule, we evaluate the operator and operand in an application, followed by a use of the monad strength β to turn it into a pair. Then we use the evaluation map under the functor T to apply the argument. Since the function is effectful, we have to collapse the effects using a μ.

$$\left[\!\!\left[\frac{\Gamma \vdash^{p} e : A}{\Gamma \vdash \mathsf{box}\,\boxed{e} : \Box A} \right]\!\!\right] \quad := \quad \llbracket \Gamma \vdash^{p} e : A \rrbracket_p\,;\,\eta_{\Box A}$$

$$\Gamma \xrightarrow{\llbracket \Gamma \vdash^{p} e : A \rrbracket_p} \Box A \xrightarrow{\eta_{\Box A}} T\Box A$$

To interpret the $\Box$ I rule, we need to interpret the pure judgement (defined later), which gives a value of type $\Box A$, and then we lift it into the monad.

$$\left[\!\!\left[\frac{\Gamma \vdash e_1 : \Box A \qquad \Gamma, x : A^{p} \vdash e_2 : B}{\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e_1\ \mathsf{in}\, e_2 : B} \right]\!\!\right] \quad := \quad \begin{array}{ll} \mathit{let} & \left\{ \begin{array}{lll} f & := & \llbracket \Gamma \vdash e_1 : \Box A \rrbracket \\ g & := & \llbracket \Gamma, x : A^{p} \vdash e_2 : B \rrbracket \end{array} \right. \\ \mathit{in} & \langle id_\Gamma, f \rangle\,;\,\tau_{\Gamma, \Box A}\,;\,Tg\,;\,\mu_B \end{array}$$

$$\Gamma \xrightarrow{\langle id_\Gamma, f \rangle} \Gamma \times T\Box A \xrightarrow{\tau_{\Gamma, \Box A}} T(\Gamma \times \Box A) \xrightarrow{Tg} T^2 B \xrightarrow{\mu_B} TB$$

To eliminate a box-ed value using the $\Box$ E rule, we first evaluate f, which gives a value of type $\Box A$, but under the monad T. We can use it to introduce a *pure* variable in the context, but we use the strength of the monad to shift the product under the T and get an extended context. We evaluate g under this extended context, and then use a μ to collapse the effects.

$$\begin{array}{rcl} \rho(\cdot) & := & id_1 \\ \rho(\Gamma, x : A^{\boldsymbol{p}}) & := & [\rho(\Gamma) \times id_{\Box A}] \\ \rho(\Gamma, x : A^{\boldsymbol{i}}) & := & \pi_1 \,;\, \rho(\Gamma) \end{array}$$

(a) $\rho(\Gamma) : \mathcal{H}om_{\mathcal{C}}(\llbracket \Gamma \rrbracket, \llbracket \Gamma^{\boldsymbol{p}} \rrbracket)$

$$\begin{array}{rcl} \mathcal{M}(\cdot) & := & id_1 \\ \mathcal{M}(\Gamma, x : A^{\boldsymbol{p}}) & := & [\mathcal{M}(\Gamma) \times \delta_A] \,;\, m^{\times}_{\Gamma^{\boldsymbol{p}}, \Box A} \\ \mathcal{M}(\Gamma, x : A^{\boldsymbol{i}}) & := & \mathcal{M}(\Gamma) \end{array}$$

(b) $\mathcal{M}(\Gamma) : \mathcal{H}om_{\mathcal{C}}(\llbracket \Gamma^{\boldsymbol{p}} \rrbracket, \Box \llbracket \Gamma^{\boldsymbol{p}} \rrbracket)$

Fig. 9. $\rho(\Gamma)$ and $\mathcal{M}(\Gamma)$

$$\left\llbracket \frac{\Gamma \vdash e_1 : \mathsf{cap} \qquad \Gamma \vdash e_2 : \mathsf{str}}{\Gamma \vdash e_1 \cdot \mathsf{print}(e_2) : \mathsf{unit}} \right\rrbracket \quad := \quad \begin{array}{ll} \mathit{let} & \left\{ \begin{array}{rcl} f & := & \llbracket \Gamma \vdash e_1 : \mathsf{cap} \rrbracket \\ g & := & \llbracket \Gamma \vdash e_2 : \mathsf{str} \rrbracket \\ p & : & C \times \Sigma^* \to T1 \\ (c, s) & \mapsto & \left(\ast \,,\, \lambda c'. \begin{cases} s & \text{if } c = c' \\ \varepsilon & \text{otherwise} \end{cases} \right) \end{array} \right. \\ \mathit{in} & \langle f, g \rangle \,;\, \beta_{C, \Sigma^*} \,;\, Tp \,;\, \mu_1 \end{array}$$

Finally, to interpret the Print rule, we need to perform a non-trivial effect. We define the function p which builds an output function that records the output on channels. Given any channel c and string s, it returns a value of type $T1$ containing the trivial value $\ast$; the output function instantiates a channel c' and tests equality with c – if it equals c, we record the string s, otherwise we just choose the empty string ε. We interpret the arguments of print and apply them to p to evaluate it. [10] The rest of the interpretation is similar to the one for $\Rightarrow$ E, with output type 1.

$$\Gamma \xrightarrow{\langle f, g \rangle} TC \times T\Sigma^* \xrightarrow{\beta_{C,\Sigma^*}} T(C \times \Sigma^*) \xrightarrow{Tp} T^2 1 \xrightarrow{\mu_1} T1$$

We used a different interpretation function for *pure* expressions, which we define below. We need to interpret the *purify* operation $\boldsymbol{p}$ on contexts, for which we define the map $\rho(\Gamma)$ in figure 9a. We also need another combinator $\mathcal{M}(\Gamma)$, defined in figure 9b, which uses the monoidal action and the idempotence of the comonad $\Box$ to distribute the $\Box$ over the products in Γ. Note that $\mathcal{M}(\Gamma)$ is an isomorphism because m and δ are.

Now, the interpretation function for pure expressions $\Gamma \vdash^{\boldsymbol{p}} e : A$ uses the ctx-pure rule, and is defined as a morphism in $\mathcal{H}om_{\mathcal{C}}(\llbracket \Gamma \rrbracket, \Box \llbracket A \rrbracket)$.

$$\left\llbracket \frac{\Gamma^{\boldsymbol{p}} \vdash e : A}{\Gamma \vdash^{\boldsymbol{p}} e : A} \right\rrbracket_p \quad := \quad \rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \Box \llbracket \Gamma^{\boldsymbol{p}} \vdash e : A \rrbracket \,;\, \phi_A$$

$$\Gamma \xrightarrow{\rho(\Gamma)} \Gamma^{\boldsymbol{p}} \xrightarrow{\mathcal{M}(\Gamma)} \Box \Gamma^{\boldsymbol{p}} \xrightarrow{\Box \llbracket \Gamma^{\boldsymbol{p}} \vdash e : A \rrbracket} \Box TA \xrightarrow{\phi_A} \Box A$$

We *purify* the context to a *pure* one, so that we can evaluate the expression. However, we need a value in $\Box A$, but the expression interpretation would produce something in TA. Now, we can

[10] We have quietly elided the interpretation of the strI rule so far. It is simply given by $\left\llbracket \frac{}{\Gamma \vdash s : \mathsf{str}} \right\rrbracket := !_\Gamma \,;\, s \,;\, \eta_{\Sigma^*}$, where $s : 1 \to \Sigma^*$ is the global element that picks out the string literal s in Σ^*.

only cancel the monad under the comonad, so we use the $\mathcal{M}(\Gamma)$ map which uses the idempotence of $\Box$ to do a readjustment. We can now evaluate the expression under the $\Box$ in the *pure* context, which gives a monadic value of type TA under the comonad $\Box$. We can finally use ϕ to cancel the monad T under the $\Box$.

5.3 Weakening and Substitution

We now give semantics for syntactic weakening and substitution.

5.3.1 Weakening. For contexts Γ and Δ, we interpret the weakening judgement $\Gamma \supseteq \Delta$ as a morphism in $\mathcal{H}om_{\mathcal{C}}(\llbracket \Gamma \rrbracket, \llbracket \Delta \rrbracket)$. We also refer to it as the weakening map $\mathsf{Wk}(\Gamma \supseteq \Delta)$.

Definition 5.5 ($\mathsf{Wk}(\Gamma \supseteq \Delta) := \llbracket \Gamma \supseteq \Delta \rrbracket : \mathcal{H}om_{\mathcal{C}}(\llbracket \Gamma \rrbracket, \llbracket \Delta \rrbracket)$).

$$
\begin{aligned}
\left\llbracket \frac{}{\cdot \supseteq \cdot} \right\rrbracket &:= id_1 \\
\left\llbracket \frac{\Gamma \supseteq \Delta}{\Gamma, x : A^q \supseteq \Delta} \right\rrbracket &:= \pi_1 \,;\, \llbracket \Gamma \supseteq \Delta \rrbracket \\
\left\llbracket \frac{\Gamma \supseteq \Delta}{\Gamma, x : A^p \supseteq \Delta, x : A^p} \right\rrbracket &:= [\llbracket \Gamma \supseteq \Delta \rrbracket \times id_{\Box A}] \\
\left\llbracket \frac{\Gamma \supseteq \Delta}{\Gamma, x : A^i \supseteq \Delta, x : A^i} \right\rrbracket &:= [\llbracket \Gamma \supseteq \Delta \rrbracket \times id_{A}]
\end{aligned}
$$

We prove a semantic weakening lemma, analogous to the syntactic weakening lemma 3.1.

LEMMA 5.6 SEMANTIC WEAKENING. *If $\Gamma \supseteq \Delta$ and $\Delta \vdash e : A$, then*

$$\llbracket \Gamma \vdash e : A \rrbracket = \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, \llbracket \Delta \vdash e : A \rrbracket.$$

5.3.2 Substitution. We now interpret a substitution $\Gamma \vdash \theta : \Delta$ as a morphism in $\mathcal{H}om_{\mathcal{C}}(\llbracket \Gamma \rrbracket, \llbracket \Delta \rrbracket)$. However, this is not a trivial iteration of the expression interpretation. The reason is that the interpretation of contexts in definition 5.2 interprets a variable $x : A^i$ in the context as an element of the type $\llbracket A \rrbracket$, and a variable $x : A^p$ as an element of the type $\Box \llbracket A \rrbracket$. However, an expression $\Gamma \vdash e : A$ will be interpreted as a morphism in $\mathcal{H}om_{\mathcal{C}}(\llbracket \Gamma \rrbracket, T \llbracket A \rrbracket)$. Operationally, we resolve this mismatch by only substituting *values* for variables in call-by-value languages, and indeed our definition of substitutions in figure 7b restricts the definition of substitution to range over values in the rule SUB-IMPURE.

Therefore, we mimic this syntactic restriction in the semantics, by giving a separate interpretation only for values, interpreting the judgement $\Gamma \vdash v : A$ as a morphism in $\mathcal{H}om_{\mathcal{C}}(\llbracket \Gamma \rrbracket, \llbracket A \rrbracket)$. Note in particular that the value interpretation yields an element of $\llbracket A \rrbracket$, as the context interpretation requires, rather than an element of $T\llbracket A \rrbracket$. This value interpretation makes use of the expression interpretation in the interpretation of λ-expressions, but the expression relation does not directly refer to the value interpretation. There are alternative presentations such as fine-grained call-by-value [Levy et al. 2003], which have a separate syntactic class of values and value judgements, and hence make the value and expression interpretations mutually recursive. However, we choose not to do that in order to remain close to the usual presentation.

Definition 5.7 ($[\![\Gamma \vdash v : A]\!]_v : \mathcal{H}om_{\mathcal{C}}([\![\Gamma]\!], [\![A]\!])$).

$$
\begin{aligned}
[\![\frac{}{\Gamma \vdash () : \mathsf{unit}}]\!]_v &:= !_\Gamma \\
[\![\frac{\Gamma \vdash v_1 : A \qquad \Gamma \vdash v_2 : B}{\Gamma \vdash (v_1, v_2) : A \times B}]\!]_v &:= \langle [\![\Gamma \vdash v_1 : A]\!]_v , [\![\Gamma \vdash v_2 : B]\!]_v \rangle \\
[\![\frac{x : A^q \in \Gamma}{\Gamma \vdash x : A}]\!]_v &:= [\![x : A^q \in \Gamma]\!] \\
[\![\frac{\Gamma, x : A^i \vdash e : B}{\Gamma \vdash \lambda x : A.\, e : A \Rightarrow B}]\!]_v &:= \mathsf{curry}\,([\![\Gamma, x : A^i \vdash e : B]\!]) \\
[\![\frac{\Gamma \vdash^p e : A}{\Gamma \vdash \mathsf{box}\, \boxed{e} : \square A}]\!]_v &:= [\![\Gamma \vdash^p e : A]\!]_p
\end{aligned}
$$

Note that box $\boxed{e}$ expressions are also values, and our *pure* interpretation does the right thing for box values, since the interpretation of $\square A$ uses the comonad, $\square[\![A]\!]$. With the interpretation of values in hand, we can define the substitution interpretation as follows.

Definition 5.8 ($[\![\Gamma \vdash \theta : \Delta]\!] : \mathcal{H}om_{\mathcal{C}}([\![\Gamma]\!], [\![\Delta]\!])$).

$$
\begin{aligned}
[\![\frac{}{\Gamma \vdash \langle\rangle : \cdot}]\!] &:= !_\Gamma \\
[\![\frac{\Gamma \vdash \theta : \Delta \qquad \Gamma \vdash^p e : A}{\Gamma \vdash \langle \theta, e^p/x \rangle : \Delta, x : A^p}]\!] &:= \langle [\![\Gamma \vdash \theta : \Delta]\!] , [\![\Gamma \vdash^p e : A]\!]_p \rangle \\
[\![\frac{\Gamma \vdash \theta : \Delta \qquad \Gamma \vdash v : A}{\Gamma \vdash \langle \theta, v^i/x \rangle : \Delta, x : A^i}]\!] &:= \langle [\![\Gamma \vdash \theta : \Delta]\!] , [\![\Gamma \vdash v : A]\!]_v \rangle
\end{aligned}
$$

We use the *pure* expression interpretation to interpret SUB-PURE, and the *impure* value interpretation for SUB-IMPURE.

Finally, we prove the semantic analogue of the syntactic substitution theorem 3.4. We prove two auxiliary lemmas 5.9 and 5.10, characterising the expression interpretation of *pure expressions* and *impure values*. The lemmas show that the interpretation for each ends in a trivial lifting into the monad T using η. This makes the proof of the semantic substitution theorem 5.11 possible.

LEMMA 5.9 PURE INTERPRETATION. *If* $\Gamma \vdash^p e : A$, *then*

$$[\![\Gamma \vdash e : A]\!] = [\![\Gamma \vdash^p e : A]\!]_p \,;\, \varepsilon_A \,;\, \eta_A.$$

LEMMA 5.10 VALUE INTERPRETATION. *If* $\Gamma \vdash v : A$, *then*

$$[\![\Gamma \vdash v : A]\!] = [\![\Gamma \vdash v : A]\!]_v \,;\, \eta_A.$$

THEOREM 5.11 SEMANTIC SUBSTITUTION. *If* $\Gamma \vdash \theta : \Delta$ *and* $\Delta \vdash e : A$, *then*

$$[\![\Gamma \vdash \theta(e) : A]\!] = [\![\Gamma \vdash \theta : \Delta]\!] \,;\, [\![\Delta \vdash e : A]\!]$$

6 EQUATIONAL THEORY

Since we have an extension of the *pure* call-by-value simply-typed lambda calculus, we want the usual $\beta\eta$-equations to hold in our theory. However, we also have the new expression forms for the $\square$ type. We want computation and extensionality rules for the box form and the let box

$$
\begin{array}{llll}
\textsc{Evaluation Contexts} & \mathcal{C} & ::= & [\cdot] \mid e\ \mathcal{C} \mid \mathcal{C}\ e \mid \lambda x : A.\ \mathcal{C} \\
& & \mid & \mathsf{fst}\ \mathcal{C} \mid \mathsf{snd}\ \mathcal{C} \mid (e\ ,\mathcal{C}) \mid (\mathcal{C}\ , e) \\
& & \mid & \mathsf{box}\ \boxed{\mathcal{C}} \mid \mathsf{let\ box}\ \boxed{x} = \mathcal{C}\ \mathsf{in}\ e \mid \mathsf{let\ box}\ \boxed{x} = e\ \mathsf{in}\ \mathcal{C} \\
& \mathcal{E} & ::= & [\cdot] \mid e\ \mathcal{E} \mid \mathcal{E}\ v \\
& & \mid & \mathsf{fst}\ \mathcal{E} \mid \mathsf{snd}\ \mathcal{E} \mid (e\ ,\mathcal{E}) \mid (\mathcal{E}\ , v) \\
& & \mid & \mathsf{let\ box}\ \boxed{x} = \mathcal{E}\ \mathsf{in}\ e \mid \mathsf{let\ box}\ \boxed{x} = v\ \mathsf{in}\ \mathcal{E}
\end{array}
$$

Fig. 10. Grammar extended with Evaluation Contexts

$$\Gamma \vdash e_1 \approx e_2 : A \quad e_1 \text{ and } e_2 \text{ are equal expressions of type } A \text{ in context } \Gamma$$

Fig. 11. Equality Judgements

binding form, and to handle the commuting conversions [Girard et al. 1989], we use evaluation contexts.

We extend our grammar with two kinds of evaluation contexts — a *pure* evaluation context $\mathcal{C}$, and an *impure* evaluation context $\mathcal{E}$, as shown in figure 10. The intuition is that $\mathcal{E}$ allows safe reductions for impure expressions, i.e., it picks out the contexts consistent with the evaluation order of the call-by-value simply-typed lambda calculus. The *pure* evaluation context $\mathcal{C}$ allows redexes in every sub-expression; but it is restricted only to *pure* expressions. The hole $[\cdot]$ is the empty evaluation context. We use the notation $\mathcal{C}\langle\!\langle e \rangle\!\rangle$ or $\mathcal{E}\langle\!\langle e \rangle\!\rangle$ to indicate that we're replacing the hole in the respective evaluation context with e.

We define a judgement form for equality of terms, as shown in figure 11, and state the rules for the equational theory in figure 12. The usual REFL, SYM, and TRANS rules give the reflexive, symmetric, and transitive closure, so that the equality relation is an equivalence. We also give CONG rules for each term former, which makes the relation a congruence closure.

We have the computation rules $\times_1\beta$ and $\times_2\beta$ for pairs; we only allow values for these rules. The $\times\eta$ rule is the extensionality rule for pairs, but again, restricted to values.

The $\Rightarrow\ \beta$ rule is the usual call-by-value computation rule for an application of a λ-expression to an argument. [11] Since the calculus has effects, we only allow the operand to be a value. For example, consider the function $f\ := \lambda x : \mathsf{unit}.\ x\ ;\ x$. We can safely β-reduce $f\ \ ()$ to $()\ ;\ ()$, but allowing a β-reduction for $f\ (c \cdot \mathsf{print}(s))$ would duplicate the effect!

We add η rules for functions, but we need to be careful because we have effects. For example, consider the expression $f\ := c \cdot \mathsf{print}(s)\ ;\ \lambda x.\ x$. On η-expansion, we get $g\ := \lambda y.\ f\ y$, but now the print operation is suspended in the closure, and doesn't evaluate when we apply g. Hence, we add two forms of η rules for functions — the $\Rightarrow\ \eta$-IMPURE rule only allows η-expansion for values, and the $\Rightarrow\ \eta$-PURE rule allows η-expansion also for expressions that are *pure*.

The computation rule $\square\beta$ for the $\square$ type allows computation under the let box binder. If we bind a box-ed expression under the let box binder, we can substitute the underlying expression in the motive. This is safe because e_1 is forced to be a *pure* expression.

Finally, we have the η expansion rules for the $\square$ type, which pushes an expression in an evaluation context under a let box binder. The $\square\eta$-PURE rule uses the *pure* evaluation context $\mathcal{C}$, while the $\square\eta$-IMPURE rule uses the *impure* evaluation context $\mathcal{E}$. The only difference in the rules is that the $\mathcal{C}$ evaluation context can be plugged with *pure* expressions only.

[11] The notation $[v/x]e$ is shorthand for $\langle\langle\Gamma\rangle, v^i/x\rangle(e)$ where $\langle\Gamma\rangle$ is the identity substitution $\Gamma \vdash \langle\Gamma\rangle : \Gamma$.

$$\frac{\Gamma \vdash e : A}{\Gamma \vdash e \approx e : A}\ \textsc{refl} \qquad \frac{\Gamma \vdash e_1 \approx e_2 : A}{\Gamma \vdash e_2 \approx e_1 : A}\ \textsc{sym} \qquad \frac{\Gamma \vdash e_1 \approx e_2 : A \qquad \Gamma \vdash e_2 \approx e_3 : A}{\Gamma \vdash e_1 \approx e_3 : A}\ \textsc{trans}$$

$$\frac{\Gamma \vdash e_1 \approx e_2 : A \times B}{\Gamma \vdash \mathsf{fst}\, e_1 \approx \mathsf{fst}\, e_2 : A}\ \mathsf{fst}\text{-}\textsc{cong} \qquad \frac{\Gamma \vdash e_1 \approx e_2 : A \times B}{\Gamma \vdash \mathsf{snd}\, e_1 \approx \mathsf{snd}\, e_2 : B}\ \mathsf{snd}\text{-}\textsc{cong}$$

$$\frac{\Gamma \vdash e_1 \approx e_2 : A \qquad \Gamma \vdash e_3 \approx e_4 : B}{\Gamma \vdash (e_1\, , e_3) \approx (e_2\, , e_4) : A \times B}\ \textsc{pair-cong} \qquad \frac{\Gamma, x : A^{i} \vdash e_1 \approx e_2 : B}{\Gamma \vdash \lambda x : A.\, e_1 \approx \lambda x : A.\, e_2 : A \Rightarrow B}\ \lambda\text{-}\textsc{cong}$$

$$\frac{\Gamma \vdash e_1 \approx e_2 : A \Rightarrow B \qquad \Gamma \vdash e_3 \approx e_4 : A}{\Gamma \vdash e_1\, e_3 \approx e_2\, e_4 : B}\ \textsc{app-cong} \qquad \frac{\Gamma^{p} \vdash e_1 \approx e_2 : A}{\Gamma \vdash \mathsf{box}\, \boxed{e_1} \approx \mathsf{box}\, \boxed{e_2} : \square A}\ \mathsf{box}\text{-}\textsc{cong}$$

$$\frac{\Gamma \vdash e_1 \approx e_2 : \square A \qquad \Gamma, x : A^{p} \vdash e_3 \approx e_4 : B}{\Gamma \vdash (\mathsf{let\, box}\, \boxed{x} = e_1\, \mathsf{in}\, e_3) \approx (\mathsf{let\, box}\, \boxed{x} = e_2\, \mathsf{in}\, e_4) : B}\ \mathsf{let\ box}\text{-}\textsc{cong}$$

$$\frac{\Gamma \vdash e_1 \approx e_2 : \mathsf{cap} \qquad \Gamma \vdash e_3 \approx e_4 : \mathsf{str}}{\Gamma \vdash e_1 \cdot \mathsf{print}(e_3) \approx e_2 \cdot \mathsf{print}(e_4) : \mathsf{unit}}\ \mathsf{print}\text{-}\textsc{cong}$$

Fig. 12. Equational Theory

We prove that our equality rules are sound with respect to our categorical semantics. If two expressions are equal in the equational theory, they have equal interpretations in the semantics.

Theorem 6.1 Soundness of $\approx$. *If* $\Gamma \vdash e_1 \approx e_2 : A$, *then* $[\![\Gamma \vdash e_1 : A]\!] = [\![\Gamma \vdash e_2 : A]\!]$.

7 EMBEDDING

Our language is an extension of the *pure* call-by-value simply-typed lambda calculus. But how could we claim that it is really an *extension*? In this section, we show that we can *embed* the simply-typed lambda calculus into our calculus, while still preserving its nice properties.

We give the grammar and judgements in figures 13a and 13b, typing rules in figure 13c, and the $\beta\eta$-equational theory in figure 13d, for the *pure* call-by-value simply-typed lambda calculus. Note that we choose to use the base type unit, and we leave out products because their embedding is trivial and uninteresting for our purpose.

Now, we define an embedding function from the simply-typed lambda calculus to our calculus. We use the notation $\underset{\smile}{X}$ to denote the embedding of a raw syntactic object X from STLC into our calculus. We give the syntactic translation of types, contexts, and raw terms in figure 14.

To embed the function type, we embed the domain and codomain, but we apply our comonadic type constructor $\square$ to restrict the domain to a *pure* type. This embedding is quite like the Gödel-McKinsey-Tarski embedding of the intuitionistic propositional calculus into classical S4 modal logic, as outlined in [McKinsey and Tarski 1948], but we do not need to apply the $\square$ type constructor on the codomain, because our functions are *capability-safe*. We remark that this is similar to the embedding of lax logic into S4 modal logic described in [Pfenning and Davies 2001], as well as the embedding of intuitionistic logic into linear logic [Girard 1987].

$$\frac{\Gamma \vdash v_1 : A \qquad \Gamma \vdash v_2 : B}{\Gamma \vdash \mathsf{fst}\,(v_1\,,v_2) \approx v_1 : A}\ \times_1\beta \qquad \frac{\Gamma \vdash v_1 : A \qquad \Gamma \vdash v_2 : B}{\Gamma \vdash \mathsf{snd}\,(v_1\,,v_2) \approx v_2 : B}\ \times_2\beta$$

$$\frac{\Gamma \vdash v : A \times B}{\Gamma \vdash v \approx (\mathsf{fst}\,v\,,\mathsf{snd}\,v) : A \times B}\ \times\eta$$

$$\frac{\Gamma, x : A^{i} \vdash e : B \qquad \Gamma \vdash v : A}{\Gamma \vdash (\lambda x : A.\, e)\, v \approx [v/x]e : B}\ \Rightarrow\beta$$

$$\frac{\Gamma \vdash v : A \Rightarrow B}{\Gamma \vdash v \approx \lambda x : A.\, v\, x : A \Rightarrow B}\ \Rightarrow\eta\text{-IMPURE} \qquad \frac{\Gamma \vdash^{p} e : A \Rightarrow B}{\Gamma \vdash e \approx \lambda x : A.\, e\, x : A \Rightarrow B}\ \Rightarrow\eta\text{-PURE}$$

$$\frac{\Gamma^{p} \vdash e_1 : A \qquad \Gamma, x : A^{p} \vdash e_2 : B}{\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = \mathsf{box}\,\boxed{e_1}\,\mathsf{in}\, e_2 \approx [e_1/x]e_2 : B}\ \square\beta$$

$$\frac{\Gamma \vdash^{p} e : \square A \qquad \Gamma \vdash C\langle\!\langle e \rangle\!\rangle : B \qquad \Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e\,\mathsf{in}\, C\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B}{\Gamma \vdash C\langle\!\langle e \rangle\!\rangle \approx \mathsf{let\ box}\,\boxed{x} = e\,\mathsf{in}\, C\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B}\ \square\eta\text{-PURE}$$

$$\frac{\Gamma \vdash e : \square A \qquad \Gamma \vdash \mathcal{E}\langle\!\langle e \rangle\!\rangle : B \qquad \Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e\,\mathsf{in}\, \mathcal{E}\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B}{\Gamma \vdash \mathcal{E}\langle\!\langle e \rangle\!\rangle \approx \mathsf{let\ box}\,\boxed{x} = e\,\mathsf{in}\, \mathcal{E}\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B}\ \square\eta\text{-IMPURE}$$

Fig. 12. Equational Theory

Types	A, B	$::=$	$\mathsf{unit} \mid A \Rightarrow B$
Terms	e	$::=$	$() \mid x \mid \lambda x : A.\, e \mid e_1\, e_2$
Values	v	$::=$	$() \mid x \mid \lambda x : A.\, e$
Contexts	Γ, Δ, Ψ	$::=$	$\cdot \mid \Gamma, x : A$

(a) Grammar for STLC

$x : A \in \Gamma$	x is a variable of type A in context Γ
$\Gamma \vdash_{\lambda} e : A$	e is an expression of type A in context Γ
$\Gamma \vdash_{\lambda} e_1 \approx e_2 : A$	e_1 and e_2 are equal expressions of type A in context Γ

(b) Judgements for STLC

When embedding contexts, we mark the variables as *pure* using the p annotation. To embed functions and applications, we need to use the introduction and elimination forms for $\square$. When embedding a λ-expression, the bound variable is embedded as a term of $\square$ type, so we eliminate the underlying variable using the let box binding form before using it in the body. To embed an application, we simply put the argument in a box.

We show that this translation is type preserving, i.e., well-typed expressions embed to well-typed expressions, and the type translation is preserved. Then, we show that the $\beta\eta$-equational theory of the *pure* call-by-value simply-typed lambda calculus is preserved under the translation. If

$$\frac{}{\Gamma \vdash_\lambda () : \mathsf{unit}}\ \mathsf{unitI} \qquad \frac{x : A \in \Gamma}{\Gamma \vdash_\lambda x : A}\ \textsc{Var}$$

$$\frac{\Gamma, x : A \vdash_\lambda e : B}{\Gamma \vdash_\lambda \lambda x : A.\, e : A \Rightarrow B}\ {\Rightarrow}\mathrm{I} \qquad \frac{\Gamma \vdash_\lambda e_1 : A \Rightarrow B \qquad \Gamma \vdash_\lambda e_2 : A}{\Gamma \vdash_\lambda e_1\, e_2 : B}\ {\Rightarrow}\mathrm{E}$$

(c) Typing rules for STLC

$$\frac{\Gamma \vdash_\lambda e : A}{\Gamma \vdash_\lambda e \approx e : A}\ \textsc{refl} \qquad \frac{\Gamma \vdash_\lambda e_1 \approx e_2 : A}{\Gamma \vdash_\lambda e_2 \approx e_1 : A}\ \textsc{sym}$$

$$\frac{\Gamma \vdash_\lambda e_1 \approx e_2 : A \qquad \Gamma \vdash_\lambda e_2 \approx e_3 : A}{\Gamma \vdash_\lambda e_1 \approx e_3 : A}\ \textsc{trans} \qquad \frac{\Gamma, x : A \vdash_\lambda e_1 \approx e_2 : B}{\Gamma \vdash_\lambda \lambda x : A.\, e_1 \approx \lambda x : A.\, e_2 : A \Rightarrow B}\ \lambda\textsc{-cong}$$

$$\frac{\Gamma \vdash_\lambda e_1 \approx e_2 : A \Rightarrow B \qquad \Gamma \vdash_\lambda e_3 \approx e_4 : A}{\Gamma \vdash_\lambda e_1\, e_3 \approx e_2\, e_4 : B}\ \textsc{app-cong}$$

$$\frac{\Gamma, x : A \vdash_\lambda e_1 : B \qquad \Gamma \vdash_\lambda e_2 : A}{\Gamma \vdash_\lambda (\lambda x : A.\, e_1)\, e_2 \approx [e_2/x]e_1 : B}\ {\Rightarrow}\beta \qquad \frac{\Gamma \vdash_\lambda e : A \Rightarrow B}{\Gamma \vdash_\lambda e \approx \lambda x : A.\, e\, x : A \Rightarrow B}\ {\Rightarrow}\eta$$

(d) Equational Theory for STLC

Fig. 13. The *pure* call-by-value simply-typed lambda calculus

Types	unit	$:=$	unit
	$A \Rightarrow B$	$:=$	$\square A \Rightarrow B$
Contexts	$\cdot$	$:=$	$\cdot$
	$\Gamma, x : A$	$:=$	$\Gamma, x : A^p$
Terms	$()$	$:=$	$()$
	x	$:=$	x
	$\lambda x : A.\, e$	$:=$	$\lambda z : \square A.\ \mathsf{let\ box}\ \boxed{x} = z\ \mathsf{in}\ e$
	$e_1\, e_2$	$:=$	$e_1\ \mathsf{box}\ \boxed{e_2}$

Fig. 14. Embedding STLC

two expressions are equal in the simply-typed lambda calculus, they *remain equal* after embedding into our imperative calculus.

Theorem 7.1 Type preservation. *If* $\Gamma \vdash_\lambda e : A$, *then* $\Gamma \vdash e : A$.

Theorem 7.2 Equality preservation. *If* $\Gamma \vdash_\lambda e_1 \approx e_2 : A$, *then* $\Gamma \vdash e_1 \approx e_2 : A$.

Finally, we show that our imperative calculus is a conservative extension of the simply-typed lambda calculus. To do so, we claim that if two embedded terms are equal in the extended theory, then they must have been equal in the smaller theory. This shows that the equational theory of the imperative calculus does not introduce any extra equations that would destroy the computational properties of the *pure* simply-typed lambda calculus.

Theorem 7.3 Conservative Extension. *If* $\Gamma \vdash_\lambda e_1 : A, \Gamma \vdash_\lambda e_2 : A$, *and* $\underset{\smile}{\Gamma} \vdash \underset{\smile}{e_1} \approx \underset{\smile}{e_2} : \underset{\smile}{A}$, *then* $\Gamma \vdash_\lambda e_1 \approx e_2 : A$.

8 DISCUSSION AND FUTURE WORK

There has been a vast amount of work on integrating effects into purely functional languages. Ironically though, even the very definition of what a purely functional language is has historically been a contested one. Sabry [1998] proposed that a functional language is pure when its behaviour under different evaluation strategies is "morally" the same, in the sense of Danielsson et al. [2006]. That is, if changing the evaluation strategy from call-by-value to (say) call-by-need could only change the divergence/error behaviour of programs in a language, then the language is pure. In contrast, the definition we use in this paper is less sophisticated: we take purity to be the preservation of the $\beta\eta$ equational theory of the simply-typed lambda calculus. However, it lets us prove the correctness of our embedding in an appealingly simple way, by translating derivations of equality.

The use of substructural type systems to control access to mutable data is a long-running theme in the development of programming languages. It is so long-running, in fact, that it actually predates linear logic [Girard 1987] by nearly a decade! Reynolds' Syntactic Control of Interference [Reynolds 1978] proposed using a substructural type discipline to prevent aliased access to data structures. The intuition that substructural logic corresponds to ownership of capabilities is also a very old one – O'Hearn [1993] uses it to explain his model of SCI, and Crary et al. [1999] compare their static capabilities to the capabilities in the HYDRA system of Wulf et al. [1974].

However, these comparisons remained informal, due to the fact that semanticists tended to use capabilities in a substructural fashion (e.g., see [Crary et al. 1999; Terauchi and Aiken 2006]), but from the very outset ([Dennis and Horn 1966]) to modern day applications like capability-safe Javascript [Maffeis et al. 2010], systems designers have tended to use capabilities *non-linearly*. In particular, they thought it was desirable for a principal to hand a capability to two different deputies, which is a design principle obviously incompatible with linearity.

The idea that the linear implication and intuitionistic implication could coexist, without one reducing to the other, first arose in the logic of bunched implications [O'Hearn and Pym 1999]. This led to separation logic [Reynolds 2002], which has been very successful at verifying programs with aliasable state. However, even though the semantics of separation logic supports BI, the bulk of the tooling infrastructure for separation logic (such as Smallfoot [Berdine et al. 2006]) have focused on the substructural fragment, often even omitting anything not in the linear fragment.

However, one observation very important to our work did arise from work on separation logic. Dodds et al. [2009] made the critical observation that in addition to being able to assert ownership, it is extremely useful to be able to *deny* the ownership of a capability. Basically, knowing that a client program *lacks* a capability can make it safe to invoke it in a secure context.

The idea that denial has comonadic structure was also known informally: it arises in the work of [Morrisett et al. 2005], where the exponential comonad in linear logic is modelled as the *lack* of any heap ownership; and in an intuitionistic context, the work on functional reactive programming [Krishnaswami 2013] used a capability to create temporal values, and a comonad denying ownership of it permitted writing space-leak-free reactive programs. However, both of these papers used operational unary logical relations models, and so did not prove anything about the equational theory.

Equational theories are easier to get with denotational models, and our model derives from the work of Hofmann [2003]. In his work, he developed a denotational model of space-bounded computation, by taking a naive set-theoretic semantics, and then augmenting it with intensional information. His sets were augmented with a *length function* saying how much memory each value used, and in ours, we use a weight function saying how many capabilities each value holds. (In fact, he even notes that his category also forms a model of bunched implications!) We think his approach has a high power-to-weight ratio, and hope we have shown that it has broad applicability as well.

However, this semantics is certainly not the last word: e.g., the semantics in this paper does not model the allocation of new capabilities as a program executes. In the categorical semantics of bunched logics, it is common to use functor categories, such as functors from the *category of finite sets and injections* $\mathcal{I}$, to Set, or presheaves over some other monoidal category. The functor category forms a model of BI, inheriting the cartesian closed structure where the limits are computed Kripke-style in Set, and also a monoidal closed structure using the tensor product from the monoidal category and *Day convolution*. In addition, the ability to move to a bigger set permits modelling allocation of new names and channels (e.g., as is done in models of the ν-calculus [Stark 1996]).

Another natural question is how we might handle recursion, as our explicit description of the category of capability spaces $\mathcal{C}$ in section 4 seems quite tied to Set; our semantics handles coproducts, natural numbers and iteration, but not general recursion. We have not done the work yet, but we remark that our semantics can be viewed as an instance of a more general construction. Both $\mathcal{C}$ and $\wp(\mathcal{C})$ are objects in Set, so we can construct the *slice category* or the *over category* Set$/\wp(\mathcal{C})$. The morphisms in this category are commuting triangles, with on-the-nose equality of capabilities. But, we want the *lax* morphisms that we described in $\mathcal{C}$, which uses the lattice structure of $\wp(\mathcal{C})$ to preserve capabilities. We can do this by considering $\wp(\mathcal{C})$ as a thin category (poset) and constructing the *comma category* using Set as the domain for the functors. Since $\wp(\mathcal{C})$ is finitely complete and co-complete, we get limits and co-limits in the comma category. By replaying this in a category like CPO rather than Set, we may be able to derive a domain-theoretic analogue of capability spaces.

Another direction for future work lies in the observation that our $\Box$ comonad in subsection 4.5 takes away *all* capabilities, yielding a system with a syntax like that of Pfenning and Davies [2001] with an interpretation close to the axiomatic categorical semantics proposed by Alechina et al. [2001] and Kobayashi [1997]. However, we could consider a *graded* or *indexed* version of the same, i.e., $\Box_C$, which only takes away a set of capabilities $C \in \wp(\mathcal{C})$ from a value. Our hope would be that this could form a model of systems like bounded linear logic [Dal Lago and Hofmann 2009; Orchard et al. 2019], or other systems of coeffects [Petricek et al. 2014]. One issue we foresee is that while this indexed comonad would still be a strong monoidal functor, it loses the idempotence property, which we used in our interpretation and proofs.

There has also been a great deal of work on using monads and effect systems [Gifford and Lucassen 1986; Moggi 1989; Nielson and Nielson 1999; Wadler 1998] to control the usage of effects. However, the general idea of using a static tag which broadcasts that an effect *may* occur seems somewhat the reverse of the idea of object capabilities, where access to a dynamically-passed value determines whether an effect can occur. The key feature of our system is that the comonad does not say what effects are possible, but rather asserts that effects are *absent*. This manifests in the cancellation law (in subsection 4.6) of the comonad and the monad. Still, the very phrases *"may perform"* and *"does not possess"* hint that some sort of duality ought to exist.

REFERENCES

Natasha Alechina, Michael Mendler, Valeria de Paiva, and Eike Ritter. 2001. Categorical and Kripke Semantics for Constructive S4 Modal Logic. In *Computer Science Logic, 15th International Workshop, CSL 2001. 10th Annual Conference of the EACSL, Paris, France, September 10-13, 2001, Proceedings (Lecture Notes in Computer Science)*, Laurent Fribourg (Ed.), Vol. 2142. Springer, 292–307. https://doi.org/10.1007/3-540-44802-0_21

Hendrik Pieter Barendregt. 1985. *The lambda calculus - its syntax and semantics*. Studies in logic and the foundations of mathematics, Vol. 103. North-Holland. 132 pages. https://doi.org/10.2307/2185110

Josh Berdine, Cristiano Calcagno, and Peter W. O'Hearn. 2006. Smallfoot: Modular Automatic Assertion Checking with Separation Logic. In *Formal Methods for Components and Objects*, Frank S. de Boer, Marcello M. Bonsangue, Susanne Graf, and Willem-Paul de Roever (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 115–137.

Karl Crary, David Walker, and J. Gregory Morrisett. 1999. Typed Memory Management in a Calculus of Capabilities. In *POPL '99, Proceedings of the 26th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, San Antonio, TX, USA, January 20-22, 1999*, Andrew W. Appel and Alex Aiken (Eds.). ACM, 262–275. https://doi.org/10.1145/292540.292564

Ugo Dal Lago and Martin Hofmann. 2009. Bounded Linear Logic, Revisited. In *Typed Lambda Calculi and Applications*, Pierre-Louis Curien (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 80–94.

Nils Anders Danielsson, John Hughes, Patrik Jansson, and Jeremy Gibbons. 2006. Fast and Loose Reasoning is Morally Correct. In *Conference Record of the 33rd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL '06)*. ACM, 206–217. https://doi.org/10.1145/1111037.1111056 Charleston, South Carolina, USA.

Jack B. Dennis and Earl C. Van Horn. 1966. Programming semantics for multiprogrammed computations. *Commun. ACM* 9, 3 (1966), 143–155. https://doi.org/10.1145/365230.365252

Mike Dodds, Xinyu Feng, Matthew Parkinson, and Viktor Vafeiadis. 2009. Deny-Guarantee Reasoning. In *Programming Languages and Systems*, Giuseppe Castagna (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 363–377.

Jeremy Gibbons. 2000. Calculating Functional Programs. In *Algebraic and Coalgebraic Methods in the Mathematics of Program Construction, International Summer School and Workshop, Oxford, UK, April 10-14, 2000, Revised Lectures (Lecture Notes in Computer Science)*, Roland Carl Backhouse, Roy L. Crole, and Jeremy Gibbons (Eds.), Vol. 2297. Springer, 149–202. https://doi.org/10.1007/3-540-47797-7_5

David K. Gifford and John M. Lucassen. 1986. Integrating Functional and Imperative Programming. In *Proceedings of the 1986 ACM Conference on LISP and Functional Programming (LFP '86)*. ACM, New York, NY, USA, 28–38. https://doi.org/10.1145/319838.319848

Jean-Yves Girard. 1987. Linear logic. *Theoretical Computer Science* 50, 1 (Jan 1987), 1–101. https://doi.org/10.1016/0304-3975(87)90045-4

Jean-Yves Girard, Paul Taylor, and Yves Lafont. 1989. *Proofs and Types*. Cambridge University Press, New York, NY, USA. 217–241 pages. https://doi.org/10.1007/978-1-4612-2822-6_8

Martin Hofmann. 2003. Linear types and non-size-increasing polynomial time computation. *Information and Computation* 183, 1 (may 2003), 57–85. https://doi.org/10.1016/s0890-5401(03)00009-9

Satoshi Kobayashi. 1997. Monad as modality. *Theoretical Computer Science* 175, 1 (1997), 29 – 74. https://doi.org/10.1016/S0304-3975(96)00169-7

Neelakantan R. Krishnaswami. 2013. Higher-Order Reactive Programming without Spacetime Leaks. In *International Conference on Functional Programming (ICFP)*.

Hugh C. Lauer and Roger M. Needham. 1979. On the Duality of Operating System Structures. *ACM SIGOPS Operating Systems Review* 13, 2 (apr 1979), 3–19. https://doi.org/10.1145/850657.850658

Henry M Levy. 1984. *Capability-based computer systems*. Digital Press.

Paul Blain Levy, John Power, and Hayo Thielecke. 2003. Modelling environments in call-by-value programming languages. *Information and Computation* 185, 2 (Sep 2003), 182–210. https://doi.org/10.1016/S0890-5401(03)00088-9

S. Maffeis, J. C. Mitchell, and A. Taly. 2010. Object Capabilities and Isolation of Untrusted Web Applications. In *2010 IEEE Symposium on Security and Privacy*. 125–140. https://doi.org/10.1109/SP.2010.16

J. C. C. McKinsey and Alfred Tarski. 1948. Some Theorems About the Sentential Calculi of Lewis and Heyting. *J. Symb. Log.* 13, 1 (1948), 1–15. https://doi.org/10.2307/2268135

Mark Samuel Miller. 2006. *Robust Composition: Towards a Unified Approach to Access Control and Concurrency Control*. Ph.D. Dissertation. USA. Advisor(s) Shapiro, Jonathan S. AAI3245526.

Eugenio Moggi. 1989. Computational Lambda-Calculus and Monads. In *Proceedings of the Fourth Annual Symposium on Logic in Computer Science (LICS '89), Pacific Grove, California, USA, June 5-8, 1989*. IEEE Computer Society, 14–23. https://doi.org/10.1109/LICS.1989.39155

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

Greg Morrisett, Amal Ahmed, and Matthew Fluet. 2005. L3: A Linear Language with Locations. In *Typed Lambda Calculi and Applications*, Paweł Urzyczyn (Ed.). Springer Berlin Heidelberg, Berlin, Heidelberg, 293–307.

Flemming Nielson and Hanne Riis Nielson. 1999. *Type and Effect Systems.* Springer Berlin Heidelberg, Berlin, Heidelberg, 114–136. https://doi.org/10.1007/3-540-48092-7_6

Peter W. O'Hearn and David J. Pym. 1999. The Logic of Bunched Implications. *Bulleting Symbolic Logic* 5, 2 (06 1999), 215–244. https://projecteuclid.org:443/euclid.bsl/1182353620

Dominic A. Orchard, Vilem Liepelt, and Harley Eades. 2019. Quantitative program reasoning with graded modal types. *Proceedings of the ACM on Programming Languages* (June 2019). https://kar.kent.ac.uk/74450/

P. W. O'Hearn. 1993. A model for syntactic control of interference. *Mathematical Structures in Computer Science* 3, 4 (Dec 1993), 435–465. https://doi.org/10.1017/S0960129500000311

Tomas Petricek, Dominic A. Orchard, and Alan Mycroft. 2014. Coeffects: a calculus of context-dependent computation. In *Proceedings of the 19th ACM SIGPLAN international conference on Functional programming, Gothenburg, Sweden, September 1-3, 2014*, Johan Jeuring and Manuel M. T. Chakravarty (Eds.). ACM, 123–135. https://doi.org/10.1145/2628136.2628160

Frank Pfenning and Rowan Davies. 2001. A judgmental reconstruction of modal logic. *Mathematical Structures in Computer Science* 11, 4 (2001), 511–540. https://doi.org/10.1017/S0960129501003322

John C. Reynolds. 1978. Syntactic Control of Interference. In *Proceedings of the 5th ACM SIGACT-SIGPLAN Symposium on Principles of Programming Languages (POPL '78).* ACM, 39–46. https://doi.org/10.1145/512760.512766 event-place: Tucson, Arizona.

J. C. Reynolds. 2002. Separation logic: a logic for shared mutable data structures. In *Proceedings 17th Annual IEEE Symposium on Logic in Computer Science.* 55–74. https://doi.org/10.1109/LICS.2002.1029817

Amr Sabry. 1998. What is a purely functional language? *Journal of Functional Programming* 8, 1 (Jan 1998), 1–22. https://doi.org/10.1017/S0956796897002943

Ian Stark. 1996. Categorical models for local names. *LISP and Symbolic Computation* 9, 1 (01 Feb 1996), 77–107. https://doi.org/10.1007/BF01806033

Tachio Terauchi and Alex Aiken. 2006. A Capability Calculus for Concurrency and Determinism. In *CONCUR 2006 - Concurrency Theory, 17th International Conference, CONCUR 2006, Bonn, Germany, August 27-30, 2006, Proceedings (Lecture Notes in Computer Science)*, Christel Baier and Holger Hermanns (Eds.), Vol. 4137. Springer, 218–232. https://doi.org/10.1007/11817949_15

Philip Wadler. 1990. Deforestation: transforming programs to eliminate trees. *Theoretical Computer Science* 73, 2 (jun 1990), 231–248. https://doi.org/10.1016/0304-3975(90)90147-a

Philip Wadler. 1998. The Marriage of Effects and Monads. In *Proceedings of the Third ACM SIGPLAN International Conference on Functional Programming (ICFP '98).* ACM, New York, NY, USA, 63–74. https://doi.org/10.1145/289423.289429

W. Wulf, E. Cohen, W. Corwin, A. Jones, R. Levin, C. Pierson, and F. Pollack. 1974. HYDRA: The Kernel of a Multiprocessor Operating System. *Commun. ACM* 17, 6 (Jun 1974), 337–345. https://doi.org/10.1145/355616.364017

A PROOFS FOR SECTION 3 (TYPING)

LEMMA A.1. *The weakening relation is reflexive.*

PROOF.

(1)	$\boxed{\Gamma}$	
(2)	$\quad\boxed{\Gamma = \cdot}$	
(3)	$\quad\cdot \supseteq \cdot$	$\supseteq$-ID
(4)	$\quad\boxed{\Gamma = \Gamma', x : A^q}$	
(5)	$\qquad\Gamma' \supseteq \Gamma'$	induction hypothesis
(6)	$\quad\Gamma', x : A^q \supseteq \Gamma', x : A^q$	$\supseteq$-CONG
(7)	$\Gamma \supseteq \Gamma$	

□

LEMMA A.2. *The weakening relation is transitive.*

PROOF.

(1)	$\boxed{\Gamma \supseteq \Delta, \Delta \supseteq \Psi}$	
(2)	$\quad\boxed{\Gamma = \cdot, \Delta = \cdot}$	case $\supseteq$-ID
(3)	$\qquad\Psi = \cdot$	inversion
(4)	$\quad\cdot \supseteq \cdot$	$\supseteq$-ID
(5)	$\quad\boxed{\Gamma = \Gamma', x : A^q, \Delta = \Delta', x : A^q}$	case $\supseteq$-CONG
(6)	$\qquad\boxed{\Psi = \Psi', x : A^q, \Delta' \supseteq \Psi'}$	case $\supseteq$-CONG
(7)	$\qquad\quad\Gamma' \supseteq \Psi'$	induction hypothesis
(8)	$\qquad\Gamma', x : A^q \supseteq \Psi', x : A^q$	$\supseteq$-CONG
(9)	$\qquad\boxed{\Delta' \supseteq \Psi}$	case $\supseteq$-WK
(10)	$\qquad\quad\Gamma' \supseteq \Psi$	induction hypothesis
(11)	$\quad\Gamma', x : A^q \supseteq \Psi$	induction hypothesis
(12)	$\quad\boxed{\Gamma' \supseteq \Delta}$	case $\supseteq$-WK
(13)	$\qquad\Gamma' \supseteq \Psi$	induction hypothesis
(14)	$\quad\Gamma', x : A^q \supseteq \Psi$	
(15)	$\Gamma \supseteq \Psi$	

□

Lemma A.3. *If* $x : A^q \in \Delta$ *and* $\Gamma \supseteq \Delta$, *then* $x : A^q \in \Gamma$.

Proof. Assuming $\Gamma \supseteq \Delta$, we do induction on $x : A^q \in \Delta$.

◇ $\in$ -id

(1) $\boxed{\dfrac{}{x : A^q \in (\Delta', x : A^q)}}$ $\in$-id

(2) $\dfrac{\Gamma' \supseteq \Delta'}{\Gamma', x : A^q \supseteq \Delta', x : A^q}$ $\supseteq$-cong

(3) $x : A^q \in (\Gamma', x : A^q)$ $\in$-id

◇ $\in$ -ex

(1) $\boxed{\dfrac{x : A^q \in \Delta' \quad (x \neq y)}{x : A^q \in (\Delta', y : B^r)}}$ $\in$-ex

(2) $\dfrac{\Gamma' \supseteq \Delta'}{\Gamma', y : B^r \supseteq \Delta', y : B^r}$ $\supseteq$-cong

(3) $x : A^q \in \Delta'$ inversion

(4) $\Gamma' \supseteq \Delta'$ inversion

(5) $x : A^q \in \Gamma'$ induction hypothesis

(6) $x : A^q \in (\Gamma', y : B^r)$ $\in$-ex

□

Lemma A.4. *If* $\Gamma \supseteq \Delta$, *then* $\Gamma^{p} \supseteq \Delta^{p}$.

Proof. We do induction on $\Gamma \supseteq \Delta$.

◇ $\supseteq$ -id

(1) $\boxed{\dfrac{}{\cdot \supseteq \cdot}}$ $\supseteq$-id

(2) $\cdot \supseteq \cdot$ $\supseteq$-id

◇ $\supseteq$ -cong

(1) $\dfrac{\Gamma' \supseteq \Delta'}{\Gamma', x : A^q \supseteq \Delta', x : A^q}$ $\supseteq$-cong

(2) $\Gamma' \supseteq \Delta'$ inversion

(3) $\Gamma'^{p} \supseteq \Delta'^{p}$ induction hypothesis

(4) $\boxed{q = p}$

(5) $\Gamma'^{p}, x : A^{p} \supseteq \Delta'^{p}, x : A^{p}$ $\supseteq$-cong (3)

(6) $\boxed{q = i}$

(7) $\Gamma'^{p} \supseteq \Delta'^{p}$ (3)

(8) $(\Gamma', x : A^q)^{p} \supseteq (\Delta', x : A^q)^{p}$

◇ $\supseteq$ -wk

(1) $\dfrac{\Gamma' \supseteq \Delta}{\Gamma', x : A^q \supseteq \Delta}$ $\supseteq$-wk

(2) $\Gamma' \supseteq \Delta$ inversion

(3) $\Gamma'^{p} \supseteq \Delta^{p}$ induction hypothesis

(4) $\boxed{q = p}$

(5) $\Gamma'^{p}, x : A^{p} \supseteq \Delta^{p}$ $\supseteq$-wk (3)

(6) $\boxed{q = i}$

(7) $\Gamma'^{p} \supseteq \Delta^{p}$ (3)

(8) $(\Gamma', x : A^q)^{p} \supseteq \Delta^{p}$

□

Lemma 3.1 Syntactic weakening. *If* $\Gamma \supseteq \Delta$ *and* $\Delta \vdash e : A$, *then* $\Gamma \vdash e : A$.

Proof. Assuming $\Gamma \supseteq \Delta$, we do induction on $\Delta \vdash e : A$.

◇ Var

(1) $\dfrac{x : A^q \in \Delta}{\Delta \vdash x : A}$ Var

(2) $x : A^q \in \Delta$ inversion

(3) $x : A^q \in \Gamma$ lemma A.3

(4) $\Gamma \vdash x : A$ Var

◇ unitI

(1) $\boxed{\dfrac{}{\Delta \vdash () : \mathsf{unit}}}$ unitI

(2) $\Gamma \vdash () : \mathsf{unit}$ unitI

◇ ×I

(1) $\boxed{\dfrac{\Delta \vdash e_1 : A \qquad \Delta \vdash e_2 : B}{\Delta \vdash (e_1 , e_2) : A \times B}}$ ×I

(2) $\Delta \vdash e_1 : A$ inversion

(3) $\Delta \vdash e_2 : B$ inversion

(4) $\Gamma \vdash e_1 : A$ induction hypothesis

(5) $\Gamma \vdash e_2 : B$ induction hypothesis

(6) $\Gamma \vdash (e_1 , e_2) : A \times B$ ×I

◇ $\times E_i$

(1) $\boxed{\dfrac{\Delta \vdash e : A \times B}{\Delta \vdash \mathsf{fst}\, e : A}}$ $\times E_1$

(2) $\Delta \vdash e : A \times B$ inversion

(3) $\Gamma \vdash e : A \times B$ induction hypothesis

(4) $\Gamma \vdash \mathsf{fst}\, e : A$ $\times E_1$

(1) $\boxed{\dfrac{\Delta \vdash e : A \times B}{\Delta \vdash \mathsf{snd}\, e : B}}$ $\times E_2$

(2) $\Delta \vdash e : A \times B$ inversion

(3) $\Gamma \vdash e : A \times B$ induction hypothesis

(4) $\Gamma \vdash \mathsf{snd}\, e : B$ $\times E_2$

◇ $\square$I

(1) $$\frac{\Delta \vdash^{p} e : A}{\Delta \vdash \mathsf{box}\,\boxed{e} : \square A}$$ $\square$I

(2) $\Delta \vdash^{p} e : A$ inversion

(3) $\Delta^{p} \vdash e : A$ inversion

(4) $\Gamma^{p} \supseteq \Delta^{p}$ lemma A.4

(5) $\Gamma^{p} \vdash e : A$ induction hypothesis

(6) $\Gamma \vdash^{p} e : A$ CTX-PURE

(7) $\Gamma \vdash \mathsf{box}\,\boxed{e} : \square A$ $\square$I

◇ $\square$E

(1) $$\frac{\Delta \vdash e_1 : \square A \qquad \Delta, x : A^{p} \vdash e_2 : B}{\Delta \vdash \mathsf{let\ box}\,\boxed{x} = e_1\ \mathsf{in}\ e_2 : B}$$ $\square$E

(2) $\Delta \vdash e_1 : \square A$ inversion

(3) $\Delta, x : A^{p} \vdash e_2 : B$ inversion

(4) $\Gamma \vdash e_1 : \square A$ induction hypothesis (2)

(5) $\Gamma, x : A^{p} \supseteq \Delta, x : A^{p}$ $\supseteq$-CONG

(6) $\Gamma, x : A^{p} \vdash e_2 : B$ induction hypothesis (3) (5)

(7) $\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e_1\ \mathsf{in}\ e_2 : B$ $\square$E

◇ $\Rightarrow$ I

(1) $$\frac{\Delta, x : A^{i} \vdash e : B}{\Delta \vdash \lambda x : A.\, e : A \Rightarrow B}$$ $\Rightarrow$I

(2) $\Delta, x : A^i \vdash e : B$ inversion

(3) $\Gamma, x : A^i \supseteq \Delta, x : A^i$ $\supseteq$-CONG

(4) $\Gamma, x : A^i \vdash e : B$ induction hypothesis (3)

(5) $\Gamma \vdash \lambda x.\, e : A \Rightarrow B$ $\Rightarrow$I

◇ $\Rightarrow$ E

(1) $\boxed{\dfrac{\Delta \vdash e_1 : A \Rightarrow B \qquad \Delta \vdash e_2 : A}{\Delta \vdash e_1\, e_2 : B}}$ $\Rightarrow$E

(2) $\Delta \vdash e_1 : A \Rightarrow B$ inversion

(3) $\Delta \vdash e_2 : A$ inversion

(4) $\Gamma \vdash e_1 : A \Rightarrow B$ induction hypothesis (2)

(5) $\Gamma \vdash e_2 : A$ induction hypothesis (3)

(6) $\Gamma \vdash e_1\, e_2 : B$ $\Rightarrow$E

◇ strI

(1) $\boxed{\dfrac{}{\Delta \vdash s : \mathsf{str}}}$ strI

(2) $\Gamma \vdash s : \mathsf{str}$ strI

◇ PRINT

(1) $\boxed{\dfrac{\Delta \vdash e_1 : \mathsf{cap} \qquad \Delta \vdash e_2 : \mathsf{str}}{\Delta \vdash e_1 \cdot \mathsf{print}(e_2) : \mathsf{unit}}}$ PRINT

(2) $\Delta \vdash e_1 : \mathsf{cap}$ inversion

(3) $\Delta \vdash e_2 : \mathsf{str}$ inversion

(4) $\Gamma \vdash e_1 : \mathsf{cap}$ induction hypothesis (2)

(5) $\Gamma \vdash e_2 : \mathsf{str}$ induction hypothesis (3)

(6) $\Gamma \vdash e_1 \cdot \mathsf{print}(e_2) : \mathsf{unit}$ PRINT

$\square$

Lemma A.5. *If* $\Gamma \supseteq \Delta$ *and* $\Delta \vdash \theta : \Psi$, *then* $\Gamma \vdash \theta : \Psi$.

Proof. Assuming $\Gamma \supseteq \Delta$, we do induction on $\Delta \vdash \theta : \Psi$.

◇ sub-id

(1) $\boxed{\dfrac{}{\Delta \vdash \langle\rangle : \cdot}}$ sub-id

(2) $\Gamma \vdash \langle\rangle : \cdot$ sub-id

◇ sub-pure

(1) $\boxed{\dfrac{\Delta \vdash \theta : \Psi' \qquad \Delta \vdash^{p} e : A}{\Delta \vdash \langle \theta, e^{p}/x \rangle : \Psi', x : A^{p}}}$ sub-pure

(2) $\Delta \vdash \theta' : \Psi'$ inversion

(3) $\dfrac{\Delta^{p} \vdash e : A}{\Delta \vdash^{p} e : A}$ ctx-pure

(4) $\Delta^{p} \vdash e : A$ inversion

(5) $\Gamma \vdash \theta' : \Psi'$ induction hypothesis (2)

(6) $\Gamma^{p} \supseteq \Delta^{p}$ lemma A.4

(7) $\Gamma^{p} \vdash e : A$ syntactic weakening lemma 3.1 (3)

(8) $\Gamma \vdash^{p} e : A$ ctx-pure

(9) $\Gamma \vdash \langle \theta', e^{p}/x \rangle : \Psi', x : A^{p}$ sub-pure

◇ sub-impure

(1) $\boxed{\dfrac{\Delta \vdash \theta : \Psi' \qquad \Delta \vdash v : A}{\Delta \vdash \langle \theta, v^{i}/x \rangle : \Psi', x : A^{i}}}$ sub-impure

(2) $\Delta \vdash \theta' : \Psi'$ inversion

(3) $\Delta \vdash v : A$ inversion

(4) $\Gamma \vdash \theta' : \Psi'$ induction hypothesis (2)

(5) $\Gamma \vdash v : A$ syntactic weakening lemma 3.1 (3)

(6) $\Gamma \vdash \langle \theta', v^i/x \rangle : \Psi', x : A^i$ SUB-IMPURE

$\square$

Lemma A.6. *If* $\Gamma \vdash \theta : \Delta$ *then* $\Gamma^p \vdash \theta^p : \Delta^p$.

Proof. We do induction on $\Gamma \vdash \theta : \Delta$.

(1) $\boxed{\Gamma \vdash \theta : \Delta}$

(2) $\boxed{\dfrac{}{\Gamma \vdash \langle\rangle : \cdot}}$ SUB-ID

(3) $\Gamma^p \vdash \langle\rangle : \cdot$ SUB-ID

(4) $\boxed{\dfrac{\Gamma \vdash \theta : \Delta \qquad \Gamma \vdash^p e : A}{\Gamma \vdash \langle \theta, e^p/x \rangle : \Delta, x : A^p}}$ SUB-PURE

(5) $\Gamma \vdash \theta : \Delta$ inversion

(6) $\dfrac{\Gamma^p \vdash e : A}{\Gamma \vdash^p e : A}$ CTX-PURE

(7) $\Gamma^p \vdash e : A$ inversion

(8) $\Gamma^p \vdash \theta^p : \Delta^p$ induction hypothesis

(9) $(\Gamma^p)^p \vdash e : A$ $(\Gamma^p)^p = \Gamma^p$

(10) $\Gamma^p \vdash^p e : A$ CTX-PURE

(11) $\Gamma^p \vdash \langle \theta^p, e^p/x \rangle : \Delta^p, x : A^p$ SUB-PURE

(12) $\boxed{\dfrac{\Gamma \vdash \theta : \Delta \qquad \Gamma \vdash v : A}{\Gamma \vdash \langle \theta, v^i/x \rangle : \Delta, x : A^i}}$ SUB-IMPURE

(13) $\Gamma \vdash \theta : \Delta$ inversion

(14) $\Gamma^p \vdash \theta^p : \Delta^p$ induction hypothesis

(15) $\Gamma^p \vdash \theta^p : \Delta^p$

$\square$

Lemma A.7. *For any context* Γ*, we have* $\Gamma \supseteq \Gamma^p$.

Proof. We do induction on Γ.

(1) $\boxed{\Gamma}$

(2) $\boxed{\Gamma = \cdot}$

(3)	$\cdot \supseteq \cdot$	$\supseteq$-ID
(4)	$\boxed{\Gamma = \Delta, x : A^{p}}$	
(5)	$\Delta \supseteq \Delta^{p}$	induction hypothesis
(6)	$\Delta, x : A^{p} \supseteq \Delta^{p}, x : A^{p}$	$\supseteq$-CONG
(7)	$\boxed{\Gamma = \Delta, x : A^{i}}$	
(8)	$\Delta \supseteq \Delta^{p}$	induction hypothesis
(9)	$\Delta, x : A^{i} \supseteq \Delta^{p}$	$\supseteq$-WK
(10)	$\Gamma \supseteq \Gamma^{p}$	

□

Lemma A.8. *If $\Gamma \vdash \theta : \Delta$ and $x : A^{q} \in \Delta$, then $\Gamma \vdash \theta[x] : A$.*

Proof. Assuming $\Gamma \vdash \theta : \Delta$, we do induction on $x : A^{q} \in \Delta$.

$\diamond \in$ -ID

(1)	$\boxed{\dfrac{}{x : A^{q} \in (\Delta', x : A^{q})}}$	$\in$-ID
(2)	$\boxed{q = p}$	
(3)	$\dfrac{\Gamma \vdash \phi : \Delta' \qquad \Gamma \vdash^{p} e : A}{\Gamma \vdash \langle \phi, e^{p}/x \rangle : \Delta', x : A^{p}}$	SUB-PURE
(4)	$\dfrac{\Gamma^{p} \vdash e : A}{\Gamma \vdash^{p} e : A}$	CTX-PURE
(5)	$\Gamma^{p} \vdash e : A$	inversion
(6)	$\Gamma \supseteq \Gamma^{p}$	lemma A.7
(7)	$\Gamma \vdash e : A$	syntactic weakening lemma 3.1
(8)	$\Gamma \vdash \langle \phi, e^{p}/x \rangle[x] : A$	definition
(9)	$\boxed{q = i}$	
(10)	$\dfrac{\Gamma \vdash \phi : \Delta' \qquad \Gamma \vdash v : A}{\Gamma \vdash \langle \phi, v^{i}/x \rangle : \Delta', x : A^{i}}$	SUB-IMPURE
(11)	$\Gamma \vdash v : A$	inversion
(12)	$\Gamma \vdash \langle \phi, v^{i}/x \rangle[x] : A$	definition
(13)	$\Gamma \vdash \theta[x] : A$	

◇ ∈ -EX

(1) $\dfrac{x : A^q \in \Delta' \qquad (x \neq y)}{x : A^q \in (\Delta', y : B^r)}$ ∈-EX

(2) $x : A^q \in \Delta'$ inversion

(3) $\boxed{q = p}$

(4) $\dfrac{\Gamma \vdash \phi : \Delta' \qquad \Gamma \vdash^{p} e : B}{\Gamma \vdash \langle \phi, e^p / y \rangle : \Delta', y : B^p}$ SUB-PURE

(5) $\Gamma \vdash \phi : \Delta'$ inversion

(6) $\Gamma \vdash \phi[x] : A$ induction hypothesis

(7) $\Gamma \vdash \langle \phi, e^p / y \rangle[x] : A$ definition

(8) $\boxed{q = i}$

(9) $\dfrac{\Gamma \vdash \phi : \Delta' \qquad \Gamma \vdash v : B}{\Gamma \vdash \langle \phi, v^i / y \rangle : \Delta', y : B^i}$ SUB-IMPURE

(10) $\Gamma \vdash \phi : \Delta'$ inversion

(11) $\Gamma \vdash \phi[x] : A$ induction hypothesis

(12) $\Gamma \vdash \langle \phi, v^i / y \rangle[x] : A$ definition

(13) $\Gamma \vdash \theta[x] : A$

□

Theorem 3.4 Syntactic substitution. *If $\Gamma \vdash \theta : \Delta$ and $\Delta \vdash e : A$, then $\Gamma \vdash \theta(e) : A$.*

Proof. Assuming $\Gamma \vdash \theta : \Delta$, we do induction on $\Delta \vdash e : A$.

◇ Var

(1) $\dfrac{x : A^q \in \Delta}{\Delta \vdash x : A}$ Var

(2) $x : A^q \in \Delta$ inversion

(3) $\Gamma \vdash \theta[x] : A$ lemma A.8

(4) $\Gamma \vdash \theta(x) : A$ definition

◇ unitI

(1)	$\dfrac{}{\Delta \vdash () : \mathsf{unit}}$	unitI
(2)	$\Gamma \vdash () : \mathsf{unit}$	unitI
(3)	$\Gamma \vdash \theta(()) : \mathsf{unit}$	definition

◇ ×I

(1)	$\dfrac{\Delta \vdash e_1 : A \qquad \Delta \vdash e_2 : B}{\Delta \vdash (e_1\,, e_2) : A \times B}$	×I
(2)	$\Delta \vdash e_1 : A$	inversion
(3)	$\Delta \vdash e_2 : B$	inversion
(4)	$\Gamma \vdash \theta(e_1) : A$	induction hypothesis
(5)	$\Gamma \vdash \theta(e_2) : B$	induction hypothesis
(6)	$\Gamma \vdash (\theta(e_1)\,, \theta(e_2)) : A \times B$	×I
(7)	$\Gamma \vdash \theta((e_1\,, e_2)) : A \times B$	definition

◇ $\times\mathrm{E}_i$

(1)	$\dfrac{\Delta \vdash e : A \times B}{\Delta \vdash \mathsf{fst}\, e : A}$	$\times\mathrm{E}_1$
(2)	$\Delta \vdash e : A \times B$	inversion
(3)	$\Gamma \vdash \theta(e) : A \times B$	induction hypothesis
(4)	$\Gamma \vdash \mathsf{fst}\,\theta(e) : B$	$\times\mathrm{E}_1$
(5)	$\Gamma \vdash \theta(\mathsf{fst}\, e) : B$	definition

(1)	$\dfrac{\Delta \vdash e : A \times B}{\Delta \vdash \mathsf{snd}\, e : B}$	$\times\mathrm{E}_2$
(2)	$\Delta \vdash e : A \times B$	inversion

(3) $\Gamma \vdash \theta(e) : A \times B$ induction hypothesis

(4) $\Gamma \vdash \mathsf{snd}\,\theta(e) : B$ $\times\mathrm{E}_2$

(5) $\Gamma \vdash \theta(\mathsf{snd}\,e) : B$ definition

◇ $\Rightarrow$ I

(1) $\boxed{\dfrac{\Delta, x : A^{i} \vdash e : B}{\Delta \vdash \lambda x : A.\, e : A \Rightarrow B}}$ $\Rightarrow$I

(2) $\Delta, x : A^{i} \vdash e : B$ inversion

(3) $\Gamma, y : A^{i} \supseteq \Gamma$ $\supseteq$-WK

(4) $\Gamma, y : A^{i} \vdash \theta : \Delta$ lemma A.5

(5) $\Gamma, y : A^{i} \vdash y : A$ VAR

(6) $\Gamma, y : A^{i} \vdash \langle \theta, y^{i}/x \rangle : \Delta, x : A^{i}$ SUB-IMPURE (4) (5)

(7) $\Gamma, y : A^{i} \vdash \langle \theta, y^{i}/x \rangle(e) : B$ induction hypothesis (6) (2)

(8) $\Gamma \vdash \lambda y.\, \langle \theta, y^{i}/x \rangle(e) : A \Rightarrow B$ $\Rightarrow$I

(9) $\Gamma \vdash \theta(\lambda y.\, e) : A \Rightarrow B$ definition

◇ $\Rightarrow$ E

(1) $\boxed{\dfrac{\Delta \vdash e_1 : A \Rightarrow B \qquad \Delta \vdash e_2 : A}{\Delta \vdash e_1\, e_2 : B}}$ $\Rightarrow$E

(2) $\Delta \vdash e_1 : A \Rightarrow B$ inversion

(3) $\Delta \vdash e_2 : A$ inversion

(4) $\Gamma \vdash \theta(e_1) : A \Rightarrow B$ induction hypothesis (2)

(5) $\Gamma \vdash \theta(e_2) : A$ induction hypothesis (3)

(6) $\Gamma \vdash \theta(e_1)\, \theta(e_2) : B$ $\Rightarrow$E

(7) $\Gamma \vdash \theta(e_1\, e_2) : B$ definition

◇ strI

(1) $\dfrac{}{\Delta \vdash s : \mathsf{str}}$ strI

(2) $\Gamma \vdash s : \mathsf{str}$ strI

(3) $\Gamma \vdash \theta(s) : \mathsf{str}$ definition

⋄ Print

(1) $\dfrac{\Delta \vdash e_1 : \mathsf{cap} \qquad \Delta \vdash e_2 : \mathsf{str}}{\Delta \vdash e_1 \cdot \mathsf{print}(e_2) : \mathsf{unit}}$ Print

(2) $\Delta \vdash e_1 : \mathsf{cap}$ inversion

(3) $\Delta \vdash e_2 : \mathsf{str}$ inversion

(4) $\Gamma \vdash \theta(e_1) : \mathsf{cap}$ induction hypothesis (2)

(5) $\Gamma \vdash \theta(e_2) : \mathsf{str}$ induction hypothesis (3)

(6) $\Gamma \vdash \theta(e_1) \cdot \mathsf{print}(\theta(e_2)) : \mathsf{unit}$ Print

(7) $\Gamma \vdash \theta(e_1 \cdot \mathsf{print}(e_2)) : \mathsf{unit}$ definition

⋄ $\square$I

(1) $\dfrac{\Delta \vdash^{p} e : A}{\Delta \vdash \mathsf{box}\,\boxed{e} : \square A}$ $\square$I

(2) $\dfrac{\Delta^{p} \vdash e : A}{\Delta \vdash^{p} e : A}$ ctx-pure

(3) $\Delta^{p} \vdash e : A$ inversion

(4) $\Gamma^{p} \vdash \theta^{p} : \Delta^{p}$ lemma A.6

(5) $\Gamma^{p} \vdash \theta^{p}(e) : A$ induction hypothesis (3) (4)

(6) $\Gamma \vdash^{p} \theta^{p}(e) : A$ ctx-pure

(7) $\Gamma \vdash \mathsf{box}\,\boxed{\theta^{p}(e)} : \square A$ $\square$I

(8) $\Gamma \vdash \theta(\mathsf{box}\,\boxed{e}) : \square A$ definition

⋄ $\square$E

$$(1)\quad \dfrac{\Delta \vdash e_1 : \square A \qquad \Delta, x : A^p \vdash e_2 : B}{\Delta \vdash \mathsf{let\ box}\ \boxed{x} = e_1\ \mathsf{in}\ e_2 : B} \qquad \square\mathrm{E}$$

(2)	$\Delta \vdash e_1 : \square A$	inversion
(3)	$\Delta, x : A^p \vdash e_2 : B$	inversion
(4)	$\Gamma, y : A^p \supseteq \Gamma$	$\supseteq$-WK
(5)	$\Gamma, y : A^p \vdash \theta : \Delta$	lemma A.5 (4)
(6)	$y : A^p \in \Gamma^p, y : A^p$	$\supseteq$-ID
(7)	$\Gamma^p, y : A^p \vdash y : A$	VAR
(8)	$\Gamma, y : A^p \vdash \langle \theta, y^p/x \rangle : \Delta, x : A^p$	SUB-PURE
(9)	$\Gamma, y : A^p \vdash \langle \theta, y^p/x \rangle (e_2) : B$	induction hypothesis (8) (3)
(10)	$\Gamma \vdash \theta(e_1) : \square A$	induction hypothesis (2)
(11)	$\Gamma \vdash \mathsf{let\ box}\ \boxed{y} = \theta(e_1)\ \mathsf{in}\ \langle \theta, y^p/x \rangle (e_2) : B$	$\square$E (9) (10)
(12)	$\Gamma \vdash \theta(\mathsf{let\ box}\ \boxed{x} = e_1\ \mathsf{in}\ e_2) : B$	definition

□

B PROOFS FOR SECTION 4 (SEMANTICS)

Lemma 4.5.

$$\begin{aligned} \mathsf{curry/uncurry} &: \mathcal{H}om_{\mathcal{C}}(\Gamma \times A, B) \xrightarrow{\sim} \mathcal{H}om_{\mathcal{C}}(\Gamma, A \to B) \\ \mathsf{ev}_{A,B} &: \mathcal{H}om_{\mathcal{C}}(A \to B \times A, B) \end{aligned}$$

Proof. Let $f \in \mathcal{H}om_{\mathcal{C}}(\Gamma \times A, B)$. So $f \in (|\Gamma \times A|) \to |B| = (|\Gamma| \times |A|) \to |B|$, and

$$\forall \gamma, \forall a,\ w_B(f(\gamma, a)) \subseteq w_{\Gamma \times A}(\gamma, a) = w_\Gamma(\gamma) \cup w_A(a)$$

Now, $\hat{f} \in |\gamma| \to (|A| \to |B|)$, and we claim, $\hat{f} \in \mathcal{H}om_{\mathcal{C}}(\Gamma, A \to B)$.

For any γ, we want to show that, $\hat{f}(\gamma) \in |A \to B|$. Let $C = w_{A\to B}(\hat{f}(\gamma))$, and for any a, let $c \in w_B(\hat{f}(\gamma)(a))$. Either $c \in w_A(a)$, or $c \notin w_A(a)$. If $c \in w_A(a)$, then $c \in C \cup w_A(a)$ and we're done. If $c \notin w_A(a)$, then we've found an a such that $c \in w_B(f(a))$ and $c \notin w_A(a)$, hence $c \in C$, and $c \in C \cup w_A(a)$. Thus, $w_B(f(a)) \subseteq C \cup w_A(a)$.

Next, for any γ, we want to show that $w_{A\to B}(\hat{f}(\gamma)) \subseteq w_\Gamma(\gamma)$. Let $c \in w_{A\to B}(\hat{f}(\gamma))$, then there exists an a, such that $c \in w_B(\hat{f}(\gamma)(a))$ and $c \notin w_A(a)$. But, $w_B(\hat{f}(\gamma)(a)) = w_B(f(\gamma, a)) \subseteq w_\Gamma(\gamma) \cup w_A(a)$. Since $c \notin w_A(a)$, it must be the case that $c \in w_\Gamma(\gamma)$. Hence, $w_{A\to B}(\hat{f}(\gamma)) \subseteq w_\Gamma(\gamma)$.

For the other direction, let $\hat{g} \in \mathcal{H}om_{\mathcal{C}}(\Gamma, A \to B)$. So $\hat{g} \in |\Gamma| \to (|A \to B|)$ such that,

$$\forall \gamma,\ w_{A\to B}(\hat{g}(\gamma)) \subseteq w_\Gamma(\gamma)$$

Also, $\hat{g} \in |\Gamma| \to (|A| \to |B|)$, so $g \in |\Gamma \times A| \to |B|$, and we claim $g \in \mathcal{H}om_{\mathcal{C}}(\Gamma \times A, B)$.

Now, for any γ, a, we want to show that $w_B(g(\gamma,a)) \subseteq w_{\Gamma\times A}(\gamma,a)$. Let $c \in w_B(g(\gamma,a))$. For any $a \in |A|$, either $c \in w_A(a)$, or $c \notin w_A(a)$. If $c \in w_A(a)$, then $c \in w_\Gamma(\gamma) \cup w_A(a) = w_{\Gamma\times A}(\gamma,a)$. If $c \notin w_A(a)$, then $c \in w_C(g(\gamma,a))$ and $c \notin w_A(a)$, so $c \in w_{A\to B}(\hat{g}(\gamma))$. Therefore, $c \in w_\Gamma(\gamma)$, and $c \in w_\Gamma(\gamma) \cup w_A(a) = w_{\Gamma\times A}(\gamma,a)$. Hence, $w_B(\hat{g}(\gamma)(a)) \subseteq w_{\Gamma\times A}(\gamma,a)$. □

Lemma B.1. *The following diagrams commute.*

$$\begin{tikzcd}
T \arrow[r, "\eta T"] \arrow[dr] & TT \arrow[d, "\mu"] & T \arrow[l, "T\eta"'] \arrow[dl] \\
 & T &
\end{tikzcd}
\qquad
\begin{tikzcd}
TTT \arrow[r, "\mu T"] \arrow[d, "T\mu"'] & TT \arrow[d, "\mu"] \\
TT \arrow[r, "\mu"'] & T
\end{tikzcd}$$

Proof.

$$\begin{array}{rlcrl}
 & \mu(\eta T(a,o)) & \qquad & & \mu(T\eta(a,o)) \\
= & \mu((a,\lambda c.\varepsilon),o) & & = & \mu((a,o),\lambda c.\varepsilon) \\
= & (a,\lambda c.o(c) \bullet \varepsilon) & & = & (a,\lambda c.\varepsilon \bullet o(c)) \\
= & (a,\lambda c.o(c)) & & = & (a,\lambda c.o(c)) \\
= & (a,o) & & = & (a,o)
\end{array}$$

$$\begin{array}{rlcrl}
 & \mu(\mu T(((a,o_1),o_2),o_3)) & \qquad & & \mu(T\mu(((a,o_1),o_2),o_3)) \\
= & \mu((a,\lambda c.o_2(c) \bullet o_1(c)),o_3) & & = & \mu((a,o_1),\lambda c.o_3(c) \bullet o_2(c)) \\
= & (a,\lambda c.o_3(c) \bullet (o_2(c) \bullet o_1(c))) & & = & (a,\lambda c.(o_3(c) \bullet o_2(c)) \bullet o_1(c)) \\
= & (a,\lambda c.o_3(c) \bullet o_2(c) \bullet o_1(c)) & & = & (a,\lambda c.o_3(c) \bullet o_2(c) \bullet o_1(c))
\end{array}$$

□

Lemma B.2. *Strengthening with 1 is irrelevant.*

$$\begin{tikzcd}
1 \times TA \arrow[r] \arrow[dr, "\tau_{1,A}"'] & TA \arrow[d] \\
 & T(1 \times A)
\end{tikzcd}$$

Consecutive applications of strength commute.

$$\begin{tikzcd}
(A \times B) \times TC \arrow[rr, "\tau_{A\times B,C}"] \arrow[d, "\cong"'] & & T((A \times B) \times C) \arrow[d, "\cong"] \\
A \times (B \times TC) \arrow[dr, "A\times\tau_{B,C}"'] & & T(A \times (B \times C)) \\
 & A \times T(B \times C) \arrow[ur, "\tau_{A,B\times C}"'] &
\end{tikzcd}$$

Strength commutes with monad unit and multiplication.

$$\begin{tikzcd}
 & & A \times B \arrow[dl, "A\times\eta_B"'] \arrow[dr, "\eta_{A\times B}"] & & \\
 & A \times TB \arrow[rr, "\tau_{A,B}"] & & T(A \times B) & \\
A \times T^2B \arrow[ur, "A\times\mu_B"] \arrow[rr, "\tau_{A,TB}"] & & T(A \times TB) \arrow[rr, "T\tau_{A,B}"] & & T^2(A \times B) \arrow[ul, "\mu_{A\times B}"']
\end{tikzcd}$$

Left are right strengths are compatible.

$$\begin{tikzcd}
A \times TB \arrow[r, "\tau_{A,B}"] \arrow[d, "\mathcal{B}_{A,TB}"'] & T(A \times B) \arrow[d, "T\mathcal{B}_{A,B}"] \\
TB \times A \arrow[r, "\sigma_{B,A}"'] & T(B \times A)
\end{tikzcd}$$

Proof. All monads on Set are strong, and Set is symmetric monoidal for products. Note that, T is *not* a commutative monad, because the following natural transformations are *not* equal.

$$\alpha : \quad TA \times TB \xrightarrow{\sigma_{A,TB}} T(A \times TB) \xrightarrow{T\tau_{A,B}} T^2(A \times B) \xrightarrow{\mu_{A\times B}} T(A \times B)$$

$$\beta : \quad TA \times TB \xrightarrow{\tau_{TA,B}} T(TA \times B) \xrightarrow{T\sigma_{A,B}} T^2(A \times B) \xrightarrow{\mu_{A\times B}} T(A \times B)$$

$$\begin{array}{rlrl}
 & \sigma_{A,TB}((a,o_1),(b,o_2)) & & \tau_{TA,B}((a,o_1),(b,o_2)) \\
= & T\tau_{A,B}((a,(b,o_2)),o_1) & = & T\sigma_{A,B}((((a,o_1),b)),o_2) \\
= & \mu_{A\times B}(((a,b),o_2),o_1) & = & \mu_{A\times B}(((a,b),o_1),o_2) \\
= & ((a,b),\lambda c.o_1(c) \bullet o_2(c)) & = & ((a,b),\lambda c.o_2(c) \bullet o_1(c))
\end{array}$$

This means that the order of evaluation matters depending on whether we choose α or β for evaluating products. □

Lemma B.3. *The following diagrams commute.*

$$\begin{tikzcd}
\Box & \Box\Box \arrow[l, "\varepsilon\Box"'] \arrow[r, "\Box\varepsilon"] & \Box \\
 & \Box \arrow[ul] \arrow[u, "\delta"] \arrow[ur] &
\end{tikzcd}
\qquad
\begin{tikzcd}
\Box\Box\Box & \Box\Box \arrow[l, "\delta\Box"'] \\
\Box\Box \arrow[u, "\Box\delta"] & \Box \arrow[l, "\delta"] \arrow[u, "\delta"']
\end{tikzcd}$$

Proof. Since δ and ε are identities, it follows trivially. Each arrow is weight-preserving because the weight is not altered by $\Box$, δ, or ε. □

Lemma B.4.

$$\Box TA \simeq \Box A$$

Proof. Let $a \in |A|$ such that $(a,o) \in |\Box TA|$. Since, $w_{TA}(a,o) = w_A(a) \cup \{ c \in \mathcal{C} \mid o(c) \neq \varepsilon \} = \emptyset$, we have $w_A(a) = \emptyset$ and $o(c) = \varepsilon$ for all $c \in \mathcal{C}$. Hence, $a \in |\Box A|$, and $o = \lambda c.\varepsilon$. This gives the map $\phi_A : \Box TA \to \Box A$, which is natural in A. We also have $\Box\eta_A : \Box A \to \Box TA$ sending $a \in |A|$ to $(a, \lambda c.\varepsilon)$. This gives a bijection, which is weight-preserving. □

Lemma 4.16.

$$\mathcal{H}om_{\mathcal{C}}(\Gamma \otimes A\,, B) \cong \mathcal{H}om_{\mathcal{C}}(\Gamma\,, A \multimap B)$$

Proof. Let $f \in \mathcal{H}om_{\mathcal{C}}(\Gamma \otimes A\,, B)$. So $f \in (|\Gamma \otimes A|) \to |B|$, and

$$\forall \gamma, \forall a, w_\Gamma(\gamma) \cap w_A(a) = \emptyset \Rightarrow w_B(f(\gamma, a)) \subseteq w_{\Gamma \otimes A}(\gamma, a) = w_\Gamma(\gamma) \cup w_A(a)$$

Now, $\hat{f} \in |\gamma| \to (|A| \to |B|)$, and we claim, $\hat{f} \in \mathcal{H}om_{\mathcal{C}}(\Gamma\,, A \multimap B)$.

For any γ, we want to show that, $\hat{f}(\gamma) \in |A \multimap B|$. Let $C = w_{A \multimap B}(\hat{f}(\gamma))$, and for any a such that $C \cap w_A(a) = \emptyset$, let $c \in w_B(\hat{f}(\gamma)(a))$. Either $c \in w_A(a)$, or $c \notin w_A(a)$. If $c \in w_A(a)$, then $c \in C \cup w_A(a)$ and we're done. If $c \notin w_A(a)$, then we've found an a such that $c \in w_B(f(a))$ and $c \notin w_A(a)$, hence $c \in C$, and $c \in C \cup w_A(a)$. Thus, $w_B(f(a)) \subseteq C \cup w_A(a)$.

Next, for any γ, we want to show that $w_{A \multimap B}(\hat{f}(\gamma)) \subseteq w_\Gamma(\gamma)$. Let $c \in w_{A \multimap B}(\hat{f}(\gamma))$, then there exists an a, such that $c \in w_B(\hat{f}(\gamma)(a))$ and $c \notin w_A(a)$. But, $w_B(\hat{f}(\gamma)(a)) = w_B(f(\gamma, a)) \subseteq w_\Gamma(\gamma) \cup w_A(a)$. Since $c \notin w_A(a)$, it must be the case that $c \in w_\Gamma(\gamma)$. Hence, $w_{A \multimap B}(\hat{f}(\gamma)) \subseteq w_\Gamma(\gamma)$.

For the other direction, let $\hat{g} \in \mathcal{H}om_{\mathcal{C}}(\Gamma\,, A \multimap B)$. So $\hat{g} \in |\Gamma| \to (|A \multimap B|)$ such that,

$$\forall \gamma,\ w_{A \multimap B}(\hat{g}(\gamma)) \subseteq w_\Gamma(\gamma)$$

Also, $\hat{g} \in |\Gamma| \to (|A| \to |B|)$, so $g \in |\Gamma \times A| \to |B| \subseteq |\Gamma \otimes A| \to |B|$, and we claim $g \in \mathcal{H}om_{\mathcal{C}}(\Gamma \otimes A\,, B)$.

Now, for any γ, a, such that $w_\Gamma(\gamma) \cap w_A(a) = \emptyset$, we want to show that $w_B(g(\gamma, a)) \subseteq w_{\Gamma \otimes A}(\gamma, a)$. Let $c \in w_B(g(\gamma, a))$. For any $a \in |A|$, either $c \in w_A(a)$, or $c \notin w_A(a)$. If $c \in w_A(a)$, then $c \in w_\Gamma(\gamma) \cup w_A(a) = w_{\Gamma \otimes A}(\gamma, a)$. If $c \notin w_A(a)$, then $c \in w_C(g(\gamma, a))$ and $c \notin w_A(a)$, so $c \in w_{A \multimap B}(\hat{g}(\gamma))$. Therefore, $c \in w_\Gamma(\gamma)$, and $c \in w_\Gamma(\gamma) \cup w_A(a) = w_{\Gamma \otimes A}(\gamma, a)$. Hence, $w_B(\hat{g}(\gamma)(a)) \subseteq w_{\Gamma \otimes A}(\gamma, a)$. □

C PROOFS FOR SECTION 5 (INTERPRETATION)

Lemma C.1. *If $\Gamma \supseteq \Delta$, then*

$$\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \square\, \mathsf{Wk}(\Gamma^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}}) = \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, \rho(\Delta) \,;\, \mathcal{M}(\Delta)$$

Proof. We do induction on $\Gamma \supseteq \Delta$.

◇ $\dfrac{}{\cdot \supseteq \cdot}\ \supseteq\text{-ID}$

$\rho(\cdot) \,;\, \mathcal{M}(\cdot) \,;\, \square\, \mathsf{Wk}(\cdot^{\boldsymbol{p}} \supseteq \cdot^{\boldsymbol{p}})$

$=\langle$ definition $\rangle$

$id_1 \,;\, id_1 \,;\, \square id_1$

$=\langle$ $\square$ preserves id $\rangle$

$id_1 \,;\, id_1 \,;\, id_1$

$=\langle$ definition $\rangle$

$\mathsf{Wk}(\cdot \supseteq \cdot) \,;\, \rho(\cdot) \,;\, \mathcal{M}(\cdot)$

◇ $\dfrac{\Gamma \supseteq \Delta}{\Gamma, x : A^q \supseteq \Delta, x : A^q}\ \supseteq\text{-CONG}$

When $q = \boldsymbol{p}$,

$$\boxed{\rho(\Gamma, x : A^{p}) \mathbin{;} \mathcal{M}(\Gamma, x : A^{p}) \mathbin{;} \Box\,\mathsf{Wk}(\Gamma^{p}, x : A^{p} \supseteq \Delta^{p}, x : A^{p})}$$

$\eqsim\langle$ definition $\rangle$

$$\boxed{[\rho(\Gamma) \times id_{\Box A}] \mathbin{;} [\mathcal{M}(\Gamma) \times \delta_A] \mathbin{;} m^{\times}_{\Gamma^{p},\Box A} \mathbin{;} \Box[\mathsf{Wk}(\Gamma^{p} \supseteq \Delta^{p}) \times id_{\Box A}]}$$

$\eqsim\langle$ monoidal action of $\Box$ $\rangle$

$$\boxed{[\rho(\Gamma) \times id_{\Box A}] \mathbin{;} [\mathcal{M}(\Gamma) \times \delta_A] \mathbin{;} [\Box\,\mathsf{Wk}(\Gamma^{p} \supseteq \Delta^{p}) \times \Box id_{\Box A}]; m^{\times}_{\Delta^{p},\Box A}}$$

$\eqsim\langle$ exchange law $\rangle$

$$\boxed{[\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \Box\,\mathsf{Wk}(\Gamma^{p} \supseteq \Delta^{p}) \times id_{\Box A} \mathbin{;} \delta_A \mathbin{;} \Box id_{\Box A}] \mathbin{;} m^{\times}_{\Delta^{p},\Box A}}$$

$\eqsim\langle$ identity law $\rangle$

$$\boxed{[\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \Box\,\mathsf{Wk}(\Gamma^{p} \supseteq \Delta^{p}) \times \delta_A] \mathbin{;} m^{\times}_{\Delta^{p},\Box A}}$$

$\eqsim\langle$ induction hypothesis $\rangle$

$$\boxed{[\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta) \times \delta_A] \mathbin{;} m^{\times}_{\Delta^{p},\Box A}}$$

$\eqsim\langle$ identity law $\rangle$

$$\boxed{[\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta) \times id_{\Box A} \mathbin{;} id_{\Box A} \mathbin{;} \delta_A] \mathbin{;} m^{\times}_{\Delta^{p},\Box A}}$$

$\eqsim\langle$ exchange law $\rangle$

$$\boxed{[\mathsf{Wk}(\Gamma \supseteq \Delta) \times id_{\Box A}] \mathbin{;} [\rho(\Delta) \times id_{\Box A}] \mathbin{;} [\mathcal{M}(\Delta) \times \delta_A] \mathbin{;} m^{\times}_{\Delta^{p},\Box A}}$$

$\eqsim\langle$ definition $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma, x : A^{p} \supseteq \Delta, x : A^{p}) \mathbin{;} \rho(\Delta, x : A^{p}) \mathbin{;} \mathcal{M}(\Delta, x : A^{p})}$$

When $q = \boldsymbol{i}$,

$$\boxed{\rho(\Gamma, x : A^{i}) \mathbin{;} \mathcal{M}(\Gamma, x : A^{i}) \mathbin{;} \Box\,\mathsf{Wk}((\Gamma, x : A^{i})^{p} \supseteq (\Delta, x : A^{i})^{p})}$$

$\eqsim\langle$ definition $\rangle$

$$\boxed{\rho(\Gamma, x : A^{i}) \mathbin{;} \mathcal{M}(\Gamma, x : A^{i}) \mathbin{;} \Box\,\mathsf{Wk}(\Gamma^{p} \supseteq \Delta^{p})}$$

$\eqsim\langle$ definition $\rangle$

$$\boxed{\pi_1 \mathbin{;} \rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \Box\,\mathsf{Wk}(\Gamma^{p} \supseteq \Delta^{p})}$$

$\eqsim\langle$ induction hypothesis $\rangle$

$$\boxed{\pi_1 \mathbin{;} \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta)}$$

$\eqsim\langle$ definition of π_1 $\rangle$

$$\boxed{\langle \pi_1 \mathbin{;} \mathsf{Wk}(\Gamma \supseteq \Delta)\ ,\ \pi_2 \mathbin{;} id_A \rangle \mathbin{;} \pi_1 \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta)}$$

$\eqsim\langle$ universal property of product $\rangle$

$$\boxed{[\mathsf{Wk}(\Gamma \supseteq \Delta) \times id_A] \mathbin{;} \pi_1 \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta)}$$

$\eqsim\langle$ definition $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma, x : A^{i} \supseteq \Delta, x : A^{i}) \mathbin{;} \rho(\Delta, x : A^{i}) \mathbin{;} \mathcal{M}(\Delta, x : A^{i})}$$

$$\diamond\ \frac{\Gamma \supseteq \Delta}{\Gamma, x : A^q \supseteq \Delta}\ \supseteq\text{-WK}$$

When $q = \boldsymbol{p}$,

$$\rho(\Gamma, x : A^{\boldsymbol{p}}) \,;\, \mathcal{M}(\Gamma, x : A^{\boldsymbol{p}}) \,;\, \Box\,\mathrm{Wk}(\Gamma^{\boldsymbol{p}}, x : A^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}})$$

$=\langle$ definition $\rangle$

$$[\rho(\Gamma) \times id_{\Box A}] \,;\, [\mathcal{M}(\Gamma) \times \delta_A] \,;\, m^{\times}_{\Gamma^{\boldsymbol{p}},\Box A} \,;\, \Box(\pi_1 \,;\, \mathrm{Wk}(\Gamma^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}}))$$

$=\langle$ $\Box$ preserves composition $\rangle$

$$[\rho(\Gamma) \times id_{\Box A}] \,;\, [\mathcal{M}(\Gamma) \times \delta_A] \,;\, m^{\times}_{\Gamma^{\boldsymbol{p}},\Box A} \,;\, \Box\pi_1 \,;\, \Box\,\mathrm{Wk}(\Gamma^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}})$$

$=\langle$ exchange law $\rangle$

$$[\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \times id_{\Box A} \,;\, \delta_A] \,;\, m^{\times}_{\Gamma^{\boldsymbol{p}},\Box A} \,;\, \Box\pi_1 \,;\, \Box\,\mathrm{Wk}(\Gamma^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}})$$

$=\langle$ identity law $\rangle$

$$[\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \times \delta_A] \,;\, m^{\times}_{\Gamma^{\boldsymbol{p}},\Box A} \,;\, \Box\pi_1 \,;\, \Box\,\mathrm{Wk}(\Gamma^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}})$$

$=\langle$ definition of $m^{\times}$ $\rangle$

$$[\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \times \delta_A] \,;\, \pi_1 \,;\, \Box\,\mathrm{Wk}(\Gamma^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}})$$

$=\langle$ universal property of product $\rangle$

$$\langle \pi_1 \,;\, \rho(\Gamma) \,;\, \mathcal{M}(\Gamma)\ ,\ \pi_2 \,;\, \delta_A \rangle \,;\, \pi_1 \,;\, \Box\,\mathrm{Wk}(\Gamma^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}})$$

$=\langle$ definition of π_1 $\rangle$

$$\pi_1 \,;\, \rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \Box\,\mathrm{Wk}(\Gamma^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}})$$

$=\langle$ induction hypothesis $\rangle$

$$\pi_1 \,;\, \mathrm{Wk}(\Gamma \supseteq \Delta) \,;\, \rho(\Delta) \,;\, \mathcal{M}(\Delta)$$

$=\langle$ definition $\rangle$

$$\mathrm{Wk}(\Gamma, x : A^{\boldsymbol{p}} \supseteq \Delta) \,;\, \rho(\Delta) \,;\, \mathcal{M}(\Delta)$$

When $q = \boldsymbol{i}$,

$$\rho(\Gamma, x : A^{\boldsymbol{i}}) \,;\, \mathcal{M}(\Gamma, x : A^{\boldsymbol{i}}) \,;\, \Box\,\mathrm{Wk}((\Gamma, x : A^{\boldsymbol{i}})^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}})$$

$=\langle$ definition $\rangle$

$$\pi_1 \,;\, \rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \Box\,\mathrm{Wk}(\Gamma^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}})$$

$=\langle$ induction hypothesis $\rangle$

$$\pi_1 \,;\, \mathrm{Wk}(\Gamma \supseteq \Delta) \,;\, \rho(\Delta) \,;\, \mathcal{M}(\Delta)$$

$=\langle$ definition $\rangle$

$$\mathrm{Wk}(\Gamma, x : A^{\boldsymbol{i}} \supseteq \Delta) \,;\, \rho(\Delta) \,;\, \mathcal{M}(\Delta)$$

$\Box$

Lemma C.2. *If $x : A^q \in \Delta$ and $\Gamma \supseteq \Delta$, then*

$$[\![x : A^q \in \Gamma]\!] = \mathrm{Wk}(\Gamma \supseteq \Delta) \,;\, [\![x : A^q \in \Delta]\!]$$

Proof. Assume $\Gamma \supseteq \Delta$. We do induction on $x : A^q \in \Delta$ followed by inversion on $\Gamma \supseteq \Delta$.

$$\diamond \quad \frac{}{x : A^q \in (\Gamma, x : A^q)} \in\text{-ID}$$

When $q = \boldsymbol{i}$,

$$
\begin{aligned}
& \boxed{[\![x : A^{\boldsymbol{i}} \in (\Gamma, x : A^{\boldsymbol{i}})]\!]} \\
\dashv\langle \; & \text{definition} \; \rangle \\
& \boxed{\pi_2} \\
\dashv\langle \; & \text{identity law} \; \rangle \\
& \boxed{\pi_2 \,;\, id_A} \\
\dashv\langle \; & \text{definition of } \pi_2 \; \rangle \\
& \boxed{\langle \pi_1 \,;\, \mathsf{Wk}(\Gamma \supseteq \Delta) \,,\, \pi_2 \,;\, id_A \rangle \,;\, \pi_2} \\
\dashv\langle \; & \text{universal property of products} \; \rangle \\
& \boxed{[\mathsf{Wk}(\Gamma \supseteq \Delta) \times id_A] \,;\, \pi_2} \\
\dashv\langle \; & \text{definition} \; \rangle \\
& \boxed{\mathsf{Wk}(\Gamma, x : A^{\boldsymbol{i}} \supseteq \Delta, x : A^{\boldsymbol{i}}) \,;\, [\![x : A^{\boldsymbol{i}} \in (\Delta, x : A^{\boldsymbol{i}})]\!]}
\end{aligned}
$$

When $q = \boldsymbol{p}$,

$$
\begin{aligned}
& \boxed{[\![x : A^{\boldsymbol{p}} \in (\Gamma, x : A^{\boldsymbol{p}})]\!]} \\
\dashv\langle \; & \text{definition} \; \rangle \\
& \boxed{\pi_2 \,;\, \varepsilon_A} \\
\dashv\langle \; & \text{identity law} \; \rangle \\
& \boxed{\pi_2 \,;\, id_{\Box A} \,;\, \varepsilon_A} \\
\dashv\langle \; & \text{definition of } \pi_2 \; \rangle \\
& \boxed{\langle \pi_1 \,;\, \mathsf{Wk}(\Gamma \supseteq \Delta) \,,\, \pi_2 \,;\, id_{\Box A} \rangle \,;\, \pi_2 \,;\, \varepsilon_A} \\
\dashv\langle \; & \text{universal property of products} \; \rangle \\
& \boxed{[\mathsf{Wk}(\Gamma \supseteq \Delta) \times id_{\Box A}] \,;\, \pi_2 \,;\, \varepsilon_A} \\
\dashv\langle \; & \text{definition} \; \rangle \\
& \boxed{\mathsf{Wk}(\Gamma, x : A^{\boldsymbol{p}} \supseteq \Delta, x : A^{\boldsymbol{p}}) \,;\, [\![x : A^{\boldsymbol{p}} \in (\Delta, x : A^{\boldsymbol{p}})]\!]}
\end{aligned}
$$

$$\diamond \quad \frac{x : A^q \in \Gamma \qquad (x \neq y)}{x : A^q \in (\Gamma, y : B^r)} \in\text{-EX}$$

When $r = \boldsymbol{i}$,

$$
\begin{aligned}
& \boxed{[\![x : A^q \in (\Gamma, y : B^r)]\!]} \\
\dashv\langle \; & \text{definition} \; \rangle \\
& \boxed{\pi_1 \,;\, [\![x : A^q \in \Gamma]\!]}
\end{aligned}
$$

$=\langle$ induction hypothesis $\rangle$

$$\pi_1 \mathbin{;} \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![x : A^q \in \Delta]\!]$$

$=\langle$ definition of π_2 $\rangle$

$$\langle \pi_1 \mathbin{;} \mathsf{Wk}(\Gamma \supseteq \Delta) \,,\, \pi_2 \mathbin{;} id_B \rangle \mathbin{;} \pi_1 \mathbin{;} [\![x : A^q \in \Delta]\!]$$

$=\langle$ universal property of products $\rangle$

$$[\mathsf{Wk}(\Gamma \supseteq \Delta) \times id_B] \mathbin{;} \pi_1 \mathbin{;} [\![x : A^q \in \Delta]\!]$$

$=\langle$ definition $\rangle$

$$\mathsf{Wk}(\Gamma, y : B^r \supseteq \Delta, y : B^r) \mathbin{;} [\![x : A^q \in (\Delta, y : B^r)]\!]$$

When $r = \boldsymbol{p}$,

$$[\![x : A^q \in (\Gamma, y : B^r)]\!]$$

$=\langle$ definition $\rangle$

$$\pi_1 \mathbin{;} [\![x : A^q \in \Gamma]\!]$$

$=\langle$ induction hypothesis $\rangle$

$$\pi_1 \mathbin{;} \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![x : A^q \in \Delta]\!]$$

$=\langle$ definition of π_2 $\rangle$

$$\langle \pi_1 \mathbin{;} \mathsf{Wk}(\Gamma \supseteq \Delta) \,,\, \pi_2 \mathbin{;} id_{\square} B \rangle \mathbin{;} \pi_1 \mathbin{;} [\![x : A^q \in \Delta]\!]$$

$=\langle$ universal property of products $\rangle$

$$[\mathsf{Wk}(\Gamma \supseteq \Delta) \times id_{\square} B] \mathbin{;} \pi_1 \mathbin{;} [\![x : A^q \in \Delta]\!]$$

$=\langle$ definition $\rangle$

$$\mathsf{Wk}(\Gamma, y : B^r \supseteq \Delta, y : B^r) \mathbin{;} [\![x : A^q \in (\Delta, y : B^r)]\!]$$

□

Lemma 5.6 Semantic weakening. *If $\Gamma \supseteq \Delta$ and $\Delta \vdash e : A$, then*

$$[\![\Gamma \vdash e : A]\!] = \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e : A]\!].$$

Proof. We proceed by induction on $\Delta \vdash e : A$.

$$\diamond \quad \frac{x : A^q \in \Gamma}{\Gamma \vdash x : A}\ \textsc{Var}$$

$$[\![\Gamma \vdash x : A]\!]$$

$=\langle$ definition $\rangle$

$$[\![x : A^i \in \Gamma]\!]$$

$=\langle$ lemma C.2 $\rangle$

$$\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![x : A^i \in \Delta]\!]$$

$\preceq\langle$ definition $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash x : A]\!]}$$

$\diamond$ $$\frac{}{\Gamma \vdash () : \mathsf{unit}}\ \mathrm{unitI}$$

$$\boxed{[\![\Gamma \vdash () : \mathsf{unit}]\!]}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{!_\Gamma \mathbin{;} \eta_1}$$

$\preceq\langle$ universal property of 1 $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} !_\Delta ; \eta_1}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash () : \mathsf{unit}]\!]}$$

$\diamond$ $$\frac{\Gamma \vdash e_1 : A \qquad \Gamma \vdash e_2 : B}{\Gamma \vdash (e_1\,, e_2) : A \times B}\ \times\mathrm{I}$$

$$\boxed{[\![\Gamma \vdash (e_1\,, e_2) : A \times B]\!]}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{\langle [\![\Gamma \vdash e_1 : A]\!]\,, [\![\Gamma \vdash e_2 : B]\!] \rangle \mathbin{;} \beta_{A,B}}$$

$\preceq\langle$ induction hypothesis $\rangle$

$$\boxed{\langle \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Gamma \vdash e_1 : A]\!]\,, \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Gamma \vdash e_2 : B]\!] \rangle \mathbin{;} \beta_{A,B}}$$

$\preceq\langle$ universal property of products $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} \langle [\![\Delta \vdash e_1 : A]\!]\,, [\![\Delta \vdash e_2 : B]\!] \rangle \mathbin{;} \beta_{A,B}}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash (e_1\,, e_2) : A \times B]\!]}$$

$\diamond$ $$\frac{\Gamma \vdash e : A \times B}{\Gamma \vdash \mathsf{fst}\, e : A}\ \times\mathrm{E}_1$$

$$\boxed{[\![\Gamma \vdash \mathsf{fst}\, e : A \times B]\!]}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash e : A \times B]\!] \mathbin{;} T\pi_1}$$

$\preceq\langle$ induction hypothesis $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e : A \times B]\!] \mathbin{;} T\pi_1}$$

$\preceq\langle$ definition $\rangle$

$$\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash \mathsf{fst}\, e : A]\!]$$

$$\diamond\ \frac{\Gamma \vdash e : A \times B}{\Gamma \vdash \mathsf{snd}\, e : B}\ {\times}\mathrm{E}_2$$

$$\begin{aligned}
& [\![\Gamma \vdash \mathsf{snd}\, e : A \times B]\!] \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& [\![\Gamma \vdash e : A \times B]\!] \mathbin{;} T\pi_2 \\
\preceq\ & \langle\ \text{induction hypothesis}\ \rangle \\
& \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e : A \times B]\!] \mathbin{;} T\pi_2 \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash \mathsf{snd}\, e : B]\!]
\end{aligned}$$

$$\diamond\ \frac{\Gamma, x : A^{i} \vdash e : B}{\Gamma \vdash \lambda x : A.\, e : A \Rightarrow B}\ {\Rightarrow}\mathrm{I}$$

$$\begin{aligned}
& [\![\Gamma \vdash \lambda x.\, e : A \Rightarrow B]\!] \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \mathsf{curry}\,([\![\Gamma, x : A^{i} \vdash e : B]\!]) \mathbin{;} \eta_{A \to TB} \\
\preceq\ & \langle\ \text{induction hypothesis}\ \rangle \\
& \mathsf{curry}\,(\mathsf{Wk}(\Gamma, x : A^{i} \supseteq \Delta, x : A^{i}) \mathbin{;} [\![\Delta, x : A^{i} \vdash e : B]\!]) \mathbin{;} \eta_{A \to TB} \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \mathsf{curry}\,([\mathsf{Wk}(\Gamma \supseteq \Delta) \times id_A] \mathbin{;} [\![\Delta, x : A^{i} \vdash e : B]\!]) \mathbin{;} \eta_{A \to TB} \\
\preceq\ & \langle\ \text{universal property of exponential}\ \rangle \\
& \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} \mathsf{curry}\,([\![\Delta, x : A^{i} \vdash e : B]\!]) \mathbin{;} \eta_{A \to TB} \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash \lambda x.\, e : A \Rightarrow B]\!]
\end{aligned}$$

$$\diamond\ \frac{\Gamma \vdash e_1 : A \Rightarrow B \qquad \Gamma \vdash e_2 : A}{\Gamma \vdash e_1\, e_2 : B}\ {\Rightarrow}\mathrm{E}$$

$$\begin{aligned}
& [\![\Gamma \vdash e_1\, e_2 : B]\!] \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \langle [\![\Gamma \vdash e_1 : A \Rightarrow B]\!] \mathbin{,} [\![\Gamma \vdash e_2 : A]\!] \rangle \\
& \mathbin{;} \beta_{A \to TB, A} \mathbin{;} T\,\mathsf{ev}_{A, TB} \mathbin{;} \mu_B
\end{aligned}$$

$$
\begin{aligned}
&\preccurlyeq \langle\ \text{induction hypothesis}\ \rangle\\
&\quad \boxed{\langle \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_1 : A \Rightarrow B]\!] \mathbin{,} \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_2 : A]\!]\rangle \mathbin{;} \beta_{A\to TB,A} \mathbin{;} T\,\mathsf{ev}_{A,TB} \mathbin{;} \mu_B}\\
&\preccurlyeq \langle\ \text{universal property of products}\ \rangle\\
&\quad \boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} \langle [\![\Delta \vdash e_1 : A \Rightarrow B]\!] \mathbin{,} [\![\Delta \vdash e_2 : A]\!]\rangle \mathbin{;} \beta_{A\to TB,A} \mathbin{;} T\,\mathsf{ev}_{A,TB} \mathbin{;} \mu_B}\\
&\preccurlyeq \langle\ \text{definition}\ \rangle\\
&\quad \boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_1\, e_2 : B]\!]}
\end{aligned}
$$

$$
\diamond\ \frac{\Gamma \vdash e_1 : \mathsf{cap} \qquad \Gamma \vdash e_2 : \mathsf{str}}{\Gamma \vdash e_1 \cdot \mathsf{print}(e_2) : \mathsf{unit}}\ \textsc{Print}
$$

$$
\begin{aligned}
&\quad \boxed{[\![\Gamma \vdash e_1 \cdot \mathsf{print}(e_2) : \mathsf{unit}]\!]}\\
&\preccurlyeq \langle\ \text{definition}\ \rangle\\
&\quad \boxed{\langle [\![\Gamma \vdash e_1 : \mathsf{cap}]\!] \mathbin{,} [\![\Gamma \vdash e_2 : \mathsf{str}]\!]\rangle \mathbin{;} \beta_{C,\Sigma^*} \mathbin{;} Tp \mathbin{;} \mu_1}\\
&\preccurlyeq \langle\ \text{induction hypothesis}\ \rangle\\
&\quad \boxed{\langle \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_1 : \mathsf{cap}]\!] \mathbin{,} \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_2 : \mathsf{str}]\!]\rangle \mathbin{;} \beta_{C,\Sigma^*} \mathbin{;} Tp \mathbin{;} \mu_1}\\
&\preccurlyeq \langle\ \text{universal property of products}\ \rangle\\
&\quad \boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} \langle [\![\Delta \vdash e_1 : \mathsf{cap}]\!] \mathbin{,} [\![\Delta \vdash e_2 : \mathsf{str}]\!]\rangle \mathbin{;} \beta_{C,\Sigma^*} \mathbin{;} Tp \mathbin{;} \mu_1}\\
&\preccurlyeq \langle\ \text{definition}\ \rangle\\
&\quad \boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_1 \cdot \mathsf{print}(e_2) : \mathsf{unit}]\!]}
\end{aligned}
$$

$$
\diamond\ \frac{\Gamma \vdash^{p} e : A}{\Gamma \vdash \mathsf{box}\,\boxed{e} : \square A}\ \square\mathrm{I}
$$

$$
\begin{aligned}
&\quad \boxed{[\![\Gamma \vdash \mathsf{box}\,\boxed{e} : \square A]\!]}\\
&\preccurlyeq \langle\ \text{definition}\ \rangle\\
&\quad \boxed{[\![\Gamma \vdash^{p} e : A]\!]_p \mathbin{;} \eta_{\square A}}\\
&\preccurlyeq \langle\ \text{definition}\ \rangle\\
&\quad \boxed{\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \square[\![\Gamma^{p} \vdash e : A]\!] \mathbin{;} \phi_A \mathbin{;} \eta_{\square A}}\\
&\preccurlyeq \langle\ \text{induction hypothesis}\ \rangle\\
&\quad \boxed{\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \square(\mathsf{Wk}(\Gamma^{p} \supseteq \Delta^{p}) \mathbin{;} [\![\Delta^{p} \vdash e : A]\!]) \mathbin{;} \phi_A \mathbin{;} \eta_{\square A}}\\
&\preccurlyeq \langle\ \square\ \text{preserves composition}\ \rangle\\
&\quad \boxed{\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \square\,\mathsf{Wk}(\Gamma^{p} \supseteq \Delta^{p}) \mathbin{;} \square[\![\Delta^{p} \vdash e : A]\!] \mathbin{;} \phi_A \mathbin{;} \eta_{\square A}}\\
&\preccurlyeq \langle\ \text{lemma C.1}\ \rangle
\end{aligned}
$$

$$\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta) \mathbin{;} \Box[\![\Delta^{p} \vdash e : A]\!] \mathbin{;} \phi_A \mathbin{;} \eta_{\Box A}$$

$\curlyeqprec\langle$ definition $\rangle$

$$\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash^{p} e : A]\!]_p \mathbin{;} \eta_{\Box A}$$

$\curlyeqprec\langle$ definition $\rangle$

$$\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash \mathsf{box}\,\boxed{e} : \Box A]\!]$$

$$\diamond \quad \frac{\Gamma \vdash e_1 : \Box A \qquad \Gamma, x : A^{p} \vdash e_2 : B}{\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e_1 \,\mathsf{in}\, e_2 : B}\ \Box\mathrm{E}$$

$$[\![\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e_1 \,\mathsf{in}\, e_2 : B]\!]$$

$\curlyeqprec\langle$ definition $\rangle$

$$\langle id_\Gamma \,,\, [\![\Gamma \vdash e_1 : \Box A]\!]\rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} T[\![\Gamma, x : A^{p} \vdash e_2 : B]\!] \mathbin{;} \mu_B$$

$\curlyeqprec\langle$ induction hypothesis $\rangle$

$$\langle id_\Gamma \,,\, \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_1 : \Box A]\!]\rangle \mathbin{;} \tau_{\Gamma,\Box A}$$
$$\mathbin{;} T(\mathsf{Wk}(\Gamma, x : A^{p} \supseteq \Delta, x : A^{p}) \mathbin{;} [\![\Delta, x : A^{p} \vdash e_2 : B]\!]) \mathbin{;} \mu_B$$

$\curlyeqprec\langle$ definition $\rangle$

$$\langle id_\Gamma \,,\, \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_1 : \Box A]\!]\rangle \mathbin{;} \tau_{\Gamma,\Box A}$$
$$\mathbin{;} T([\mathsf{Wk}(\Gamma \supseteq \Delta) \times id_{\Box A}] \mathbin{;} [\![\Delta, x : A^{p} \vdash e_2 : B]\!]) \mathbin{;} \mu_B$$

$\curlyeqprec\langle$ T preserves composition $\rangle$

$$\langle id_\Gamma \,,\, \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_1 : \Box A]\!]\rangle \mathbin{;} \tau_{\Gamma,\Box A}$$
$$\mathbin{;} T[\mathsf{Wk}(\Gamma \supseteq \Delta) \times id_{\Box A}] \mathbin{;} T[\![\Delta, x : A^{p} \vdash e_2 : B]\!] \mathbin{;} \mu_B$$

$\curlyeqprec\langle$ tensorial strength of T $\rangle$

$$\langle id_\Gamma \,,\, \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_1 : \Box A]\!]\rangle \mathbin{;} [\mathsf{Wk}(\Gamma \supseteq \Delta) \times id_{T\Box A}] \mathbin{;} \tau_{\Delta,\Box A}$$
$$\mathbin{;} T[\![\Delta, x : A^{p} \vdash e_2 : B]\!] \mathbin{;} \mu_B$$

$\curlyeqprec\langle$ composition of products $\rangle$

$$\langle id_\Gamma \mathbin{;} \mathsf{Wk}(\Gamma \supseteq \Delta) \,,\, \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_1 : \Box A]\!] \mathbin{;} id_{T\Box A}\rangle \mathbin{;} \tau_{\Delta,\Box A}$$
$$\mathbin{;} T[\![\Delta, x : A^{p} \vdash e_2 : B]\!] \mathbin{;} \mu_B$$

$\curlyeqprec\langle$ identity law $\rangle$

$$\langle \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} id_\Delta \,,\, \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash e_1 : \Box A]\!]\rangle \mathbin{;} \tau_{\Delta,\Box A}$$
$$\mathbin{;} T[\![\Delta, x : A^{p} \vdash e_2 : B]\!] \mathbin{;} \mu_B$$

$\curlyeqprec\langle$ universal property of products $\rangle$

$$\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} \langle id_\Delta \,,\, [\![\Delta \vdash e_1 : \Box A]\!]\rangle \mathbin{;} \tau_{\Delta,\Box A} \mathbin{;} T[\![\Delta, x : A^{p} \vdash e_2 : B]\!] \mathbin{;} \mu_B$$

$\curlyeqprec\langle$ definition $\rangle$

$$\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash \mathsf{let\ box}\,\boxed{x} = e_1 \,\mathsf{in}\, e_2 : B]\!]$$

□

Lemma C.3. *If* $\Gamma \supseteq \Delta$ *and* $\Delta \vdash^{p} e : A$, *then*

$$[\![\Gamma \vdash^{p} e : A]\!]_p = \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash^{p} e : A]\!]_p.$$

Proof.

$$\boxed{[\![\Gamma \vdash^{p} e : A]\!]_p}$$

$\equiv\langle$ definition $\rangle$

$$\boxed{\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \Box[\![\Gamma^{p} \vdash e : A]\!] \mathbin{;} \phi_A}$$

$\equiv\langle$ semantic weakening lemma 5.6 $\rangle$

$$\boxed{\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \Box(\mathsf{Wk}(\Gamma^{p} \supseteq \Delta^{p}) \mathbin{;} [\![\Delta^{p} \vdash e : A]\!]) \mathbin{;} \phi_A}$$

$\equiv\langle$ $\Box$ preserves composition $\rangle$

$$\boxed{\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \Box\,\mathsf{Wk}(\Gamma^{p} \supseteq \Delta^{p}) \mathbin{;} \Box[\![\Delta^{p} \vdash e : A]\!] \mathbin{;} \phi_A}$$

$\equiv\langle$ lemma C.1 $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta) \mathbin{;} [\![\Delta^{p} \vdash e : A]\!] \mathbin{;} \phi_A}$$

$\equiv\langle$ definition $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash^{p} e : A]\!]_p}$$

□

Lemma C.4. *If* $\Gamma \supseteq \Delta$ *and* $\Delta \vdash v : A$, *then*

$$[\![\Gamma \vdash v : A]\!]_v = \mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash v : A]\!]_v.$$

Proof. Assuming $\Gamma \supseteq \Delta$, we do induction on $\Delta \vdash v : A$.

◇ $$\dfrac{x : A^{q} \in \Gamma}{\Gamma \vdash x : A}\ \textsc{Var}$$

$$\boxed{[\![\Gamma \vdash v : A]\!]_v}$$

$\equiv\langle$ definition $\rangle$

$$\boxed{[\![x : A^{q} \in \Gamma]\!]}$$

$\equiv\langle$ lemma C.2 $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![x : A^{q} \in \Delta]\!]}$$

$\equiv\langle$ definition $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \mathbin{;} [\![\Delta \vdash x : A]\!]_v}$$

◇ $$\dfrac{}{\Gamma \vdash () : \mathsf{unit}}\ \mathsf{unitI}$$

$$\boxed{[\![\Gamma \vdash () : \mathsf{unit}]\!]_v}$$
$=\langle$ definition $\rangle$
$$\boxed{!_\Gamma}$$
$=\langle$ universal property of 1 $\rangle$
$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, !_\Delta}$$
$=\langle$ definition $\rangle$
$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Gamma \vdash () : \mathsf{unit}]\!]_v}$$

$$\diamond \quad \frac{\Gamma \vdash e_1 : A \qquad \Gamma \vdash e_2 : B}{\Gamma \vdash (e_1 \,, e_2) : A \times B} \times\mathrm{I}$$

$$\boxed{[\![\Gamma \vdash (v_1 \,, v_2) : A \times B]\!]_v}$$
$=\langle$ definition $\rangle$
$$\boxed{\langle [\![\Gamma \vdash v_1 : A]\!]_v \,, [\![\Gamma \vdash v_2 : B]\!]_v \rangle}$$
$=\langle$ induction hypothesis $\rangle$
$$\boxed{\langle \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash v_1 : A]\!]_v \,, \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash v_2 : B]\!]_v \rangle}$$
$=\langle$ universal property of products $\rangle$
$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, \langle [\![\Delta \vdash v_1 : A]\!]_v \,, [\![\Delta \vdash v_2 : B]\!]_v \rangle}$$
$=\langle$ definition $\rangle$
$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash (v_1 \,, v_2) : A \times B]\!]_v}$$

$$\diamond \quad \frac{\Gamma, x : A^i \vdash e : B}{\Gamma \vdash \lambda x : A.\, e : A \Rightarrow B} \Rightarrow\mathrm{I}$$

$$\boxed{[\![\Gamma \vdash \lambda x.\, e : A \Rightarrow B]\!]_v}$$
$=\langle$ definition $\rangle$
$$\boxed{\mathsf{curry}\,([\![\Gamma, x : A^i \vdash e : B]\!])}$$
$=\langle$ semantic weakening lemma 5.6 $\rangle$
$$\boxed{\mathsf{curry}\,(\mathsf{Wk}(\Gamma, x : A^i \supseteq \Delta, x : A^i) \,;\, [\![\Delta, x : A^i \vdash e : B]\!])}$$
$=\langle$ definition $\rangle$
$$\boxed{\mathsf{curry}\,([\mathsf{Wk}(\Gamma \supseteq \Delta) \times id_A]; [\![\Delta, x : A^i \vdash e : B]\!])}$$
$=\langle$ universal property of exponential $\rangle$
$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, \mathsf{curry}\,([\![\Delta, x : A^i \vdash e : B]\!])}$$
$=\langle$ definition $\rangle$
$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash \lambda x.\, e : A \Rightarrow B]\!]_v}$$

$$\diamond\quad \frac{\Gamma \vdash^{p} e : A}{\Gamma \vdash \mathsf{box}\,\boxed{e} : \Box A}\ \Box\mathrm{I}$$

$$\begin{aligned}
&\boxed{[\![\Gamma \vdash \mathsf{box}\,\boxed{e} : \Box A]\!]_{v}} \\
\preceq\ &\langle \text{definition} \rangle \\
&\boxed{[\![\Gamma \vdash^{p} e : A]\!]_{p}} \\
\preceq\ &\langle \text{lemma C.3} \rangle \\
&\boxed{\mathrm{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash^{p} e : A]\!]_{p}} \\
\preceq\ &\langle \text{definition} \rangle \\
&\boxed{\mathrm{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash \mathsf{box}\,\boxed{e} : \Box A]\!]_{v}}
\end{aligned}$$

□

Lemma C.5. *If* $\Gamma \supseteq \Delta$ *and* $\Delta \vdash \theta : \Psi$, *then*

$$[\![\Gamma \vdash \theta : \Psi]\!] = \mathrm{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash \theta : \Psi]\!]$$

Proof. Assume $\Gamma \supseteq \Delta$. We proceed by induction on $\Delta \vdash \theta : \Psi$.

$$\diamond\quad \frac{}{\Gamma \vdash \langle\rangle : \cdot}\ \textsc{sub-id}$$

$$\begin{aligned}
&\boxed{[\![\Gamma \vdash \langle\rangle : \cdot]\!]} \\
\preceq\ &\langle \text{definition} \rangle \\
&\boxed{!_{\Gamma}} \\
\preceq\ &\langle \text{universal property of } 1 \rangle \\
&\boxed{\mathrm{Wk}(\Gamma \supseteq \Delta) \,;\, !_{\Delta}} \\
\preceq\ &\langle \text{definition} \rangle \\
&\boxed{\mathrm{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash \langle\rangle : \cdot]\!]}
\end{aligned}$$

$$\diamond\quad \frac{\Gamma \vdash \theta : \Delta \qquad \Gamma \vdash^{p} e : A}{\Gamma \vdash \langle \theta, e^{p}/x \rangle : \Delta, x : A^{p}}\ \textsc{sub-pure}$$

$$\begin{aligned}
&\boxed{[\![\Gamma \vdash \langle \theta, e^{p}/x \rangle : \Psi, x : A^{p}]\!]} \\
\preceq\ &\langle \text{definition} \rangle \\
&\boxed{\langle [\![\Gamma \vdash \theta : \psi]\!] \,,\, [\![\Gamma \vdash^{p} e : A]\!]_{p} \rangle} \\
\preceq\ &\langle \text{induction hypothesis} \rangle \\
&\boxed{\langle \mathrm{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash \theta : \psi]\!] \,,\, [\![\Gamma \vdash^{p} e : A]\!]_{p} \rangle} \\
\preceq\ &\langle \text{lemma C.3} \rangle
\end{aligned}$$

$$
\begin{aligned}
& \langle \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash \theta : \psi]\!] \,,\, \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash^{p} e : A]\!]_p \rangle \\
\preccurlyeq\; & \langle \text{universal property of products} \rangle \\
& \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, \langle [\![\Delta \vdash \theta : \psi]\!] \,,\, [\![\Delta \vdash^{p} e : A]\!]_p \rangle \\
\preccurlyeq\; & \langle \text{definition} \rangle \\
& \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash \langle \theta, e^{p}/x \rangle : \Psi, x : A^{p}]\!]
\end{aligned}
$$

$$
\diamond \;\frac{\Gamma \vdash \theta : \Delta \qquad \Gamma \vdash v : A}{\Gamma \vdash \langle \theta, v^{i}/x \rangle : \Delta, x : A^{i}}\;\textsc{sub-impure}
$$

$$
\begin{aligned}
& [\![\Gamma \vdash \langle \theta, v^{i}/x \rangle : \Psi, x : A^{i}]\!] \\
\preccurlyeq\; & \langle \text{definition} \rangle \\
& \langle [\![\Gamma \vdash \theta : \Psi]\!] \,,\, [\![\Gamma \vdash v : A]\!]_v \rangle \\
\preccurlyeq\; & \langle \text{induction hypothesis} \rangle \\
& \langle \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash \theta : \Psi]\!] \,,\, [\![\Gamma \vdash v : A]\!]_v \rangle \\
\preccurlyeq\; & \langle \text{lemma C.4} \rangle \\
& \langle \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash \theta : \Psi]\!] \,,\, \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash v : A]\!]_v \rangle \\
\preccurlyeq\; & \langle \text{universal property of products} \rangle \\
& \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, \langle [\![\Delta \vdash \theta : \Psi]\!] \,,\, [\![\Delta \vdash v : A]\!]_v \rangle \\
\preccurlyeq\; & \langle \text{definition} \rangle \\
& \mathsf{Wk}(\Gamma \supseteq \Delta) \,;\, [\![\Delta \vdash \langle \theta, v^{i}/x \rangle : \Psi, x : A^{i}]\!]
\end{aligned}
$$

□

Lemma C.6. *If* $\Gamma^{p} \vdash e : A^{p}$, *then*

$$
\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \Box [\![\Gamma^{p} \vdash^{p} e : A]\!]_p = [\![\Gamma \vdash^{p} e : A]\!]_p \,;\, \delta_A
$$

Proof.

$$
\begin{aligned}
& \rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \Box [\![\Gamma^{p} \vdash^{p} e : A]\!]_p \\
\preccurlyeq\; & \langle \text{definition} \rangle \\
& \rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \Box(\rho(\Gamma^{p}) \,;\, \mathcal{M}(\Gamma^{p}) \,;\, \Box [\![\Gamma^{p} \vdash e : A]\!] \,;\, \phi_A) \\
\preccurlyeq\; & \langle \Box \text{ preserves composition} \rangle \\
& \rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \Box \rho(\Gamma^{p}) \,;\, \Box \mathcal{M}(\Gamma^{p}) \,;\, \Box\Box [\![\Gamma^{p} \vdash e : A]\!] \,;\, \Box \phi_A \\
\preccurlyeq\; & \langle \text{definition} \rangle \\
& \rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \Box id_{\Gamma^{p}} \,;\, \delta_{\Gamma^{p}} \,;\, \delta^{-1}_{\Gamma^{p}} \,;\, \Box [\![\Gamma^{p} \vdash e : A]\!] \,;\, \phi_A \,;\, \delta_A \\
\preccurlyeq\; & \langle \text{simplification} \rangle \\
& \rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \Box [\![\Gamma^{p} \vdash e : A]\!] \,;\, \phi_A \,;\, \delta_A
\end{aligned}
$$

$\doteqdot\langle$ definition $\rangle$

$$[\![\Gamma \vdash^{p} e : A]\!]_{p} \,;\, \delta_A$$

$\square$

Lemma C.7. *If* $\Gamma \vdash \theta : \Delta$, *then*

$$\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \square[\![\Gamma^{p} \vdash \theta^{p} : \Delta^{p}]\!] = [\![\Gamma \vdash \theta : \Delta]\!] \,;\, \rho(\Delta) \,;\, \mathcal{M}(\Delta)$$

Proof. We do induction on $\Gamma \vdash \theta : \Delta$.

$\diamond$ $\dfrac{}{\Gamma \vdash \langle\rangle : \cdot}$ SUB-ID

$$\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \square[\![\Gamma^{p} \vdash \langle\rangle : \cdot]\!]$$

$\doteqdot\langle$ definition $\rangle$

$$\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \square !_{\Gamma^{p}}$$

$\doteqdot\langle$ definition $\rangle$

$$\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, !_{\square\Gamma^{p}}$$

$\doteqdot\langle$ universal property of 1 $\rangle$

$$!_{\Gamma}$$

$\doteqdot\langle$ identity law $\rangle$

$$!_{\Gamma} \,;\, id_1 \,;\, id_1$$

$\doteqdot\langle$ definition $\rangle$

$$[\![\Gamma \vdash \langle\rangle : \cdot]\!] \,;\, \rho(\cdot) \,;\, \mathcal{M}(\cdot)$$

$\diamond$ $\dfrac{\Gamma \vdash \theta : \Delta \qquad \Gamma \vdash^{p} e : A}{\Gamma \vdash \langle \theta, e^{p}/x\rangle : \Delta, x : A^{p}}$ SUB-PURE

$$\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \square[\![\Gamma^{p} \vdash \langle \theta^{p}, e^{p}/x\rangle : \Delta^{p}, x : A^{p}]\!]$$

$\doteqdot\langle$ definition $\rangle$

$$\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \square\langle [\![\Gamma^{p} \vdash \theta^{p} : \Delta^{p}]\!] \,,\, [\![\Gamma^{p} \vdash^{p} e : A]\!]_{p}\rangle$$

$\doteqdot\langle$ monoidal action of $\square$ $\rangle$

$$\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \langle \square[\![\Gamma^{p} \vdash \theta^{p} : \Delta^{p}]\!] \,,\, \square[\![\Gamma^{p} \vdash^{p} e : A]\!]_{p}\rangle \,;\, m^{\times}_{\Delta^{p},\square A}$$

$\doteqdot\langle$ universal property of products $\rangle$

$$\langle \rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \square[\![\Gamma^{p} \vdash \theta^{p} : \Delta^{p}]\!] \,,\, \rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \square[\![\Gamma^{p} \vdash^{p} e : A]\!]_{p}\rangle \,;\, m^{\times}_{\Delta^{p},\square A}$$

$\doteqdot\langle$ induction hypothesis $\rangle$

$$\langle [\![\Gamma \vdash \theta : \Delta]\!] \,;\, \rho(\Delta) \,;\, \mathcal{M}(\Delta) \,,\, \rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \square[\![\Gamma^{p} \vdash^{p} e : A]\!]_{p}\rangle \,;\, m^{\times}_{\Delta^{p},\square A}$$

$\doteqdot\langle$ lemma C.6 $\rangle$

$$
\begin{array}{ll}
& \boxed{\langle [\![\Gamma \vdash \theta : \Delta]\!] \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta) \,,\, [\![\Gamma \vdash^{p} e : A]\!]_p \mathbin{;} \delta_A \rangle \mathbin{;} m^{\times}_{\Delta^{p}, \Box A}} \\
\Bumpeq\!\langle & \text{identity law} \quad \rangle \\
& \boxed{\langle [\![\Gamma \vdash \theta : \Delta]\!] \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta) \,,\, [\![\Gamma \vdash^{p} e : A]\!]_p \mathbin{;} id_{\Box A} \mathbin{;} \delta_A \rangle \mathbin{;} m^{\times}_{\Delta^{p}, \Box A}} \\
\Bumpeq\!\langle & \text{universal property of products} \quad \rangle \\
& \boxed{\langle [\![\Gamma \vdash \theta : \Delta]\!] \,,\, [\![\Gamma \vdash^{p} e : A]\!]_p \rangle \mathbin{;} [\rho(\Delta) \mathbin{;} \mathcal{M}(\Delta) \times id_{\Box A} \mathbin{;} \delta_A] \mathbin{;} m^{\times}_{\Delta^{p}, \Box A}} \\
\Bumpeq\!\langle & \text{exchange law} \quad \rangle \\
& \boxed{\langle [\![\Gamma \vdash \theta : \Delta]\!] \,,\, [\![\Gamma \vdash^{p} e : A]\!]_p \rangle \mathbin{;} [\rho(\Delta) \times id_{\Box A}] \mathbin{;} [\mathcal{M}(\Delta) \times \delta_A] \mathbin{;} m^{\times}_{\Delta^{p}, \Box A}} \\
\Bumpeq\!\langle & \text{definition} \quad \rangle \\
& \boxed{[\![\Gamma \vdash \langle \theta, e^{p}/x \rangle : \Delta, x : A^{p}]\!] \mathbin{;} \rho(\Delta, x : A^{p}) \mathbin{;} \mathcal{M}(\Delta, x : A^{p})}
\end{array}
$$

$$
\diamond \; \frac{\Gamma \vdash \theta : \Delta \qquad \Gamma \vdash v : A}{\Gamma \vdash \langle \theta, v^{i}/x \rangle : \Delta, x : A^{i}} \; \textsc{sub-impure}
$$

$$
\begin{array}{ll}
& \boxed{\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \Box [\![\Gamma^{p} \vdash \langle \theta, v^{i}/x \rangle^{p} : (\Delta, x : A^{i})^{p}]\!]} \\
\Bumpeq\!\langle & \text{definition} \quad \rangle \\
& \boxed{\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \Box [\![\Gamma^{p} \vdash \theta^{p} : \Delta^{p}]\!]} \\
\Bumpeq\!\langle & \text{induction hypothesis} \quad \rangle \\
& \boxed{[\![\Gamma \vdash \theta : \Delta]\!] \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta)} \\
\Bumpeq\!\langle & \text{definition of } \pi_1 \quad \rangle \\
& \boxed{\langle [\![\Gamma \vdash \theta : \Delta]\!] \,,\, [\![\Gamma \vdash v : A]\!]_v \rangle \mathbin{;} \pi_1 \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta)} \\
\Bumpeq\!\langle & \text{definition} \quad \rangle \\
& \boxed{[\![\Gamma \vdash \langle \theta, v^{i}/x \rangle : \Delta, x : A^{i}]\!] \mathbin{;} \rho(\Delta, x : A^{i}) \mathbin{;} \mathcal{M}(\Delta, x : A^{i})}
\end{array}
$$

□

Lemma C.8. *For any context* Γ,

$$\mathsf{Wk}(\Gamma \supseteq \Gamma^{p}) = \rho(\Gamma)$$

Proof. We do induction on Γ.

$\diamond\ \Gamma = \cdot$

$$
\begin{array}{ll}
& \boxed{\mathsf{Wk}(\cdot \supseteq \cdot^{p})} \\
\Bumpeq\!\langle & \text{definition} \quad \rangle \\
& \boxed{\mathsf{Wk}(\cdot \supseteq \cdot)} \\
\Bumpeq\!\langle & \text{definition} \quad \rangle \\
& \boxed{id_1}
\end{array}
$$

$=\!\!\!\langle$ definition $\rangle$

$$\boxed{\rho(\cdot)}$$

$\diamond\ \Gamma = \Delta, x : A^q$

When $q = \boldsymbol{p}$,

$$\boxed{\mathsf{Wk}(\Delta, x : A^{\boldsymbol{p}} \supseteq \Delta^{\boldsymbol{p}}, x : A^{\boldsymbol{p}})}$$

$=\!\!\!\langle$ definition $\rangle$

$$\boxed{[\mathsf{Wk}(\Delta \supseteq \Delta^{\boldsymbol{p}}) \times id_{\Box A}]}$$

$=\!\!\!\langle$ induction hypothesis $\rangle$

$$\boxed{[\rho(\Delta) \times id_{\Box A}]}$$

$=\!\!\!\langle$ definition $\rangle$

$$\boxed{\rho(\Delta, x : A^{\boldsymbol{p}})}$$

When $q = \boldsymbol{i}$,

$$\boxed{\mathsf{Wk}(\Delta, x : A^{\boldsymbol{i}} \supseteq \Delta^{\boldsymbol{p}})}$$

$=\!\!\!\langle$ definition $\rangle$

$$\boxed{\pi_1 \,;\, \mathsf{Wk}(\Delta \supseteq \Delta^{\boldsymbol{p}})}$$

$=\!\!\!\langle$ induction hypothesis $\rangle$

$$\boxed{\pi_1 \,;\, \rho(\Delta)}$$

$=\!\!\!\langle$ definition $\rangle$

$$\boxed{\rho(\Delta, x : A^{\boldsymbol{i}})}$$

$\Box$

Lemma 5.9 Pure interpretation. *If $\Gamma \vdash^{\boldsymbol{p}} e : A$, then*

$$[\![\Gamma \vdash e : A]\!] = [\![\Gamma \vdash^{\boldsymbol{p}} e : A]\!]_p \,;\, \varepsilon_A \,;\, \eta_A.$$

Proof. Assume $\Gamma \vdash^{\boldsymbol{p}} e : A$. By inversion, we have $\Gamma^{\boldsymbol{p}} \vdash e : A$.

$$\boxed{[\![\Gamma \vdash e : A]\!]}$$

$=\!\!\!\langle$ semantic weakening lemma 5.6 $\rangle$

$$\boxed{\mathsf{Wk}(\Gamma \supseteq \Gamma^{\boldsymbol{p}}) \,;\, [\![\Gamma^{\boldsymbol{p}} \vdash e : A]\!]}$$

$=\!\!\!\langle$ lemma C.8 $\rangle$

$$\boxed{\rho(\Gamma) \,;\, [\![\Gamma^{\boldsymbol{p}} \vdash e : A]\!]}$$

$=\!\!\!\langle$ definition $\rangle$

$$\boxed{\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \Box[\![\Gamma^{\boldsymbol{p}} \vdash e : A]\!] \,;\, \varepsilon_{TA}}$$

$=\!\!\!\langle$ definition $\rangle$

$$\begin{array}{ll} & \boxed{\rho(\Gamma) \,;\, \mathcal{M}(\Gamma) \,;\, \Box[\![\Gamma^{p} \vdash e : A]\!] \,;\, \phi_A \,;\, \varepsilon_A \,;\, \eta_A} \\ \preceq\langle & \text{definition} \quad \rangle \\ & \boxed{[\![\Gamma \vdash^{p} e : A]\!]_p \,;\, \varepsilon_A \,;\, \eta_A} \end{array}$$

□

Lemma 5.10 Value interpretation. *If* $\Gamma \vdash v : A$, *then*

$$[\![\Gamma \vdash v : A]\!] = [\![\Gamma \vdash v : A]\!]_v \,;\, \eta_A.$$

Proof. We proceed by induction on $\Gamma \vdash v : A$.

⋄ $$\frac{}{\Gamma \vdash () : \mathsf{unit}}\ \mathsf{unitI}$$

$$\begin{array}{ll} & \boxed{[\![\Gamma \vdash () : \mathsf{unit}]\!]} \\ \preceq\langle & \text{definition} \quad \rangle \\ & \boxed{!_\Gamma \,;\, \eta_1} \\ \preceq\langle & \text{definition} \quad \rangle \\ & \boxed{[\![\Gamma \vdash () : \mathsf{unit}]\!]_v \,;\, \eta_1} \end{array}$$

⋄ $$\frac{\Gamma \vdash v_1 : A \qquad \Gamma \vdash v_2 : B}{\Gamma \vdash (v_1\,,v_2) : A \times B}\ \times\mathrm{I}$$

$$\begin{array}{ll} & \boxed{[\![\Gamma \vdash (v_1\,,v_2) : A \times B]\!]} \\ \preceq\langle & \text{definition} \quad \rangle \\ & \boxed{\langle [\![\Gamma \vdash v_1 : A]\!] \,,\, [\![\Gamma \vdash v_2 : B]\!] \rangle \,;\, \beta_{A,B}} \\ \preceq\langle & \text{induction hypothesis} \quad \rangle \\ & \boxed{\langle [\![\Gamma \vdash v_1 : A]\!]_v \,;\, \eta_A \,,\, [\![\Gamma \vdash v_2 : B]\!]_v \,;\, \eta_B \rangle \,;\, \beta_{A,B}} \\ \preceq\langle & \text{tensorial strength of } T \quad \rangle \\ & \boxed{\langle [\![\Gamma \vdash v_1 : A]\!]_v \,,\, [\![\Gamma \vdash v_2 : B]\!]_v \,;\, \eta_B \rangle \,;\, \sigma_{A,B}} \\ \preceq\langle & \text{tensorial strength of } T \quad \rangle \\ & \boxed{\langle [\![\Gamma \vdash v_1 : A]\!]_v \,,\, [\![\Gamma \vdash v_2 : B]\!]_v \rangle \,;\, \eta_{A\times B}} \\ \preceq\langle & \text{definition} \quad \rangle \\ & \boxed{[\![\Gamma \vdash (v_1\,,v_2) : A \times B]\!]_v \,;\, \eta_{A\times B}} \end{array}$$

⋄ $$\frac{x : A^{q} \in \Gamma}{\Gamma \vdash x : A}\ \textsc{Var}$$

$$\boxed{[\![\Gamma \vdash x : A]\!]}$$

$\quad \mathrel{=\!\!\!\langle}$ definition $\rangle$

$$[\![x : A^q \in \Gamma]\!] \mathbin{;} \eta_A$$

$\quad \mathrel{=\!\!\!\langle}$ definition $\rangle$

$$[\![\Gamma \vdash x : A]\!]_v \mathbin{;} \eta_A$$

$$\diamond \quad \frac{\Gamma, x : A^i \vdash e : B}{\Gamma \vdash \lambda x : A.\, e : A \Rightarrow B} \Rightarrow\mathrm{I}$$

$$[\![\Gamma \vdash \lambda x.\, e : A \Rightarrow B]\!]$$

$\quad \mathrel{=\!\!\!\langle}$ definition $\rangle$

$$\mathsf{curry}\,([\![\Gamma, x : A^i \vdash e : B]\!]) \mathbin{;} \eta_{A \to TB}$$

$\quad \mathrel{=\!\!\!\langle}$ definition $\rangle$

$$[\![\Gamma \vdash \lambda x.\, e : A \Rightarrow B]\!]_v \mathbin{;} \eta_{A \to TB}$$

$$\diamond \quad \frac{\Gamma \vdash^{p} e : A}{\Gamma \vdash \mathsf{box}\, \boxed{e} : \Box A} \Box\mathrm{I}$$

$$[\![\Gamma \vdash \mathsf{box}\, \boxed{e} : \Box A]\!]$$

$\quad \mathrel{=\!\!\!\langle}$ definition $\rangle$

$$[\![\Gamma \vdash^{p} e : A]\!]_p \mathbin{;} \eta_{\Box A}$$

$\quad \mathrel{=\!\!\!\langle}$ definition $\rangle$

$$[\![\Gamma \vdash \mathsf{box}\, \boxed{e} : \Box A]\!]_v \mathbin{;} \eta_{\Box A}$$

$\square$

Lemma C.9. *If* $\Gamma \vdash \theta : \Delta$ *and* $x : A^q \in \Delta$, *then*

$$[\![\Gamma \vdash \theta[x] : A]\!] = [\![\Gamma \vdash \theta : \Delta]\!] \mathbin{;} [\![x : A^q \in \Delta]\!] \mathbin{;} \eta_A$$

Proof. We proceed by induction on $x : A^q \in \Delta$.

$$\diamond \quad \frac{}{x : A^q \in (\Gamma, x : A^q)} \in\text{-ID}$$

When $q = \boldsymbol{p}$,

$$[\![\Gamma \vdash \langle \phi, e^{p}/x \rangle[x] : A]\!]$$

$\quad \mathrel{=\!\!\!\langle}$ definition $\rangle$

$$[\![\Gamma \vdash e : A]\!]$$

$\quad \mathrel{=\!\!\!\langle}$ pure interpretation lemma 5.9 $\rangle$

$$[\![\Gamma \vdash^{p} e : A]\!]_p \mathbin{;} \varepsilon_A \mathbin{;} \eta_A$$

$\preceq\langle$ definition of π_2 $\rangle$

$$\langle [\![\Gamma \vdash \phi : \Delta]\!] , [\![\Gamma \vdash^{\boldsymbol{p}} e : A]\!]_p \rangle \,;\, \pi_2 \,;\, \varepsilon_A \,;\, \eta_A$$

$\preceq\langle$ definition $\rangle$

$$[\![\Gamma \vdash \langle \phi, e^{\boldsymbol{p}}/x \rangle : \Delta, x : A^{\boldsymbol{p}}]\!] \,;\, [\![x : A^{\boldsymbol{p}} \in (\Delta, x : A^{\boldsymbol{p}})]\!] \,;\, \eta_A$$

When $q = \boldsymbol{i}$,

$$[\![\Gamma \vdash \langle \phi, v^{\boldsymbol{i}}/x \rangle[x] : A]\!]$$

$\preceq\langle$ definition $\rangle$

$$[\![\Gamma \vdash v : A]\!]$$

$\preceq\langle$ value interpretation lemma 5.10 $\rangle$

$$[\![\Gamma \vdash v : A]\!]_v \,;\, \eta_A$$

$\preceq\langle$ definition of π_2 $\rangle$

$$\langle [\![\Gamma \vdash \phi : \Delta]\!] , [\![\Gamma \vdash v : A]\!]_v \rangle \,;\, \pi_2 \,;\, \eta_A$$

$\preceq\langle$ definition $\rangle$

$$[\![\Gamma \vdash \langle \phi, v^{\boldsymbol{i}}/x \rangle : \Delta, x : A^{\boldsymbol{i}}]\!] \,;\, [\![x : A^{\boldsymbol{i}} \in (\Delta, x : A^{\boldsymbol{i}})]\!] \,;\, \eta_A$$

$$\diamond \quad \frac{x : A^q \in \Gamma \qquad (x \neq y)}{x : A^q \in (\Gamma, y : B^r)}\ \in\text{-EX}$$

When $r = \boldsymbol{p}$

$$[\![\Gamma \vdash \langle \phi, e^{\boldsymbol{p}}/y \rangle[x] : A]\!]$$

$\preceq\langle$ definition $\rangle$

$$[\![\Gamma \vdash \phi[x] : A]\!]$$

$\preceq\langle$ induction hypothesis $\rangle$

$$[\![\Gamma \vdash \phi : \Delta]\!] \,;\, [\![x : A^q \in \Delta]\!] \,;\, \eta_A$$

$\preceq\langle$ definition of π_1 $\rangle$

$$\langle [\![\Gamma \vdash \phi : \Delta]\!] , [\![\Gamma \vdash^{\boldsymbol{p}} e : B]\!] \rangle \,;\, \pi_1 \,;\, [\![x : A^q \in \Delta]\!] \,;\, \eta_A$$

$\preceq\langle$ definition $\rangle$

$$[\![\Gamma \vdash \langle \phi, e^{\boldsymbol{p}}/y \rangle : \Delta, y : B^{\boldsymbol{p}}]\!] \,;\, \pi_1 \,;\, [\![x : A^q \in \Delta]\!] \,;\, \eta_A$$

$\preceq\langle$ definition $\rangle$

$$[\![\Gamma \vdash \langle \phi, e^{\boldsymbol{p}}/y \rangle : \Delta, y : B^{\boldsymbol{p}}]\!] \,;\, [\![x : A^q \in (\Delta, y : B^{\boldsymbol{p}})]\!] \,;\, \eta_A$$

When $r = \boldsymbol{i}$,

$$\boxed{[\![\Gamma \vdash \langle \phi, v^i/y \rangle[x] : A]\!]}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash \phi[x] : A]\!]}$$

$\preceq\langle$ induction hypothesis $\rangle$

$$\boxed{[\![\Gamma \vdash \phi : \Delta]\!] \,;\, [\![x : A^q \in \Delta]\!] \,;\, \eta_A}$$

$\preceq\langle$ definition of π_1 $\rangle$

$$\boxed{\langle [\![\Gamma \vdash \phi : \Delta]\!] \,,\, [\![\Gamma \vdash v : B]\!] \rangle \,;\, \pi_1 \,;\, [\![x : A^q \in \Delta]\!] \,;\, \eta_A}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash \langle \phi, v^i/y \rangle : \Delta, y : B^i]\!] \,;\, \pi_1 \,;\, [\![x : A^q \in \Delta]\!] \,;\, \eta_A}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash \langle \phi, v^i/y \rangle : \Delta, y : B^i]\!] \,;\, [\![x : A^q \in (\Delta, y : B^i)]\!] \,;\, \eta_A}$$

□

Theorem 5.11 Semantic substitution. *If* $\Gamma \vdash \theta : \Delta$ *and* $\Delta \vdash e : A$, *then*

$$[\![\Gamma \vdash \theta(e) : A]\!] = [\![\Gamma \vdash \theta : \Delta]\!] \,;\, [\![\Delta \vdash e : A]\!]$$

Proof. Assume $\Gamma \vdash \theta : \Delta$. We proceed by induction on $\Delta \vdash e : A$.

◇ $$\dfrac{x : A^q \in \Gamma}{\Gamma \vdash x : A}\ \textsc{Var}$$

$$\boxed{[\![\Gamma \vdash \theta(x) : A]\!]}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash \theta[x] : A]\!]}$$

$\preceq\langle$ lemma C.9 $\rangle$

$$\boxed{[\![\Gamma \vdash \theta : \Delta]\!] \,;\, [\![x : A^q \in \Delta]\!] \,;\, \eta_A}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash \theta : \Delta]\!] \,;\, [\![\Delta \vdash x : A]\!]}$$

◇ $$\dfrac{}{\Gamma \vdash () : \mathsf{unit}}\ \mathsf{unitI}$$

$$\boxed{[\![\Gamma \vdash \theta(()) : \mathsf{unit}]\!]}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash () : \mathsf{unit}]\!]}$$

$\preceq\langle$ definition $\rangle$

$$\boxed{!_\Gamma \,;\, \eta_1}$$

$\equiv$ ⟨ universal property of 1 ⟩

$$\boxed{[\![\Gamma \vdash \theta : \Delta]\!] \,;\, !_{\Delta} \,;\, \eta_1}$$

$\equiv$ ⟨ definition ⟩

$$\boxed{[\![\Gamma \vdash \theta : \Delta]\!] \,;\, [\![\Delta \vdash () : \mathsf{unit}]\!]}$$

$$\diamond \;\; \frac{\Gamma \vdash e_1 : A \qquad \Gamma \vdash e_2 : B}{\Gamma \vdash (e_1\,, e_2) : A \times B} \times\mathrm{I}$$

$$\boxed{[\![\Gamma \vdash \theta((e_1\,, e_2)) : A \times B]\!]}$$

$\equiv$ ⟨ definition ⟩

$$\boxed{[\![\Gamma \vdash (\theta(e_1)\,, \theta(e_2)) : A \times B]\!]}$$

$\equiv$ ⟨ definition ⟩

$$\boxed{\langle [\![\Gamma \vdash \theta(e_1) : A]\!] \,, [\![\Gamma \vdash \theta(e_2) : B]\!] \rangle \,;\, \beta_{A,B}}$$

$\equiv$ ⟨ induction hypothesis ⟩

$$\boxed{\langle [\![\Gamma \vdash \theta : \Delta]\!] \,;\, [\![\Delta \vdash e_1 : A]\!] \,, [\![\Gamma \vdash \theta : \Delta]\!] \,;\, [\![\Delta \vdash e_2 : B]\!] \rangle \,;\, \beta_{A,B}}$$

$\equiv$ ⟨ universal property of products ⟩

$$\boxed{[\![\Gamma \vdash \theta : \Delta]\!] \,;\, \langle [\![\Delta \vdash e_1 : A]\!] \,, [\![\Delta \vdash e_2 : B]\!] \rangle \,;\, \beta_{A,B}}$$

$\equiv$ ⟨ definition ⟩

$$\boxed{[\![\Gamma \vdash \theta : \Delta]\!] \,;\, [\![\Delta \vdash (e_1\,, e_2) : A \times B]\!]}$$

$$\diamond \;\; \frac{\Gamma \vdash e : A \times B}{\Gamma \vdash \mathsf{fst}\, e : A} \times\mathrm{E}_1$$

$$\boxed{[\![\Gamma \vdash \theta(\mathsf{fst}\, e) : A]\!]}$$

$\equiv$ ⟨ definition ⟩

$$\boxed{[\![\Gamma \vdash \mathsf{fst}\, \theta(e) : A]\!]}$$

$\equiv$ ⟨ definition ⟩

$$\boxed{[\![\Gamma \vdash \theta(e) : A \times B]\!] \,;\, T\pi_1}$$

$\equiv$ ⟨ induction hypothesis ⟩

$$\boxed{[\![\Gamma \vdash \theta : \Delta]\!] \,;\, [\![\Delta \vdash e : A \times B]\!] \,;\, T\pi_1}$$

$\equiv$ ⟨ definition ⟩

$$\boxed{[\![\Gamma \vdash \theta : \Delta]\!] \,;\, [\![\Delta \vdash \mathsf{fst}\, e : A]\!]}$$

$$\diamond \;\; \frac{\Gamma \vdash e : A \times B}{\Gamma \vdash \mathsf{snd}\, e : B} \times\mathrm{E}_2$$

$$\llbracket \Gamma \vdash \theta(\mathsf{snd}\, e) : B \rrbracket$$
$\curlyeqprec$ ⟨ definition ⟩
$$\llbracket \Gamma \vdash \mathsf{snd}\, \theta(e) : B \rrbracket$$
$\curlyeqprec$ ⟨ definition ⟩
$$\llbracket \Gamma \vdash \theta(e) : A \times B \rrbracket \mathbin{;} T\pi_2$$
$\curlyeqprec$ ⟨ induction hypothesis ⟩
$$\llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash e : A \times B \rrbracket \mathbin{;} T\pi_2$$
$\curlyeqprec$ ⟨ definition ⟩
$$\llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash \mathsf{snd}\, e : B \rrbracket$$

$$\diamond\ \frac{\Gamma \vdash^{p} e : A}{\Gamma \vdash \mathsf{box}\,\boxed{e} : \square A}\ \square\mathrm{I}$$

$$\llbracket \Gamma \vdash \theta(\mathsf{box}\,\boxed{e}) : \square A \rrbracket$$
$\curlyeqprec$ ⟨ definition ⟩
$$\llbracket \Gamma \vdash \mathsf{box}\,\boxed{\theta^{p}(e)} : \square A \rrbracket$$
$\curlyeqprec$ ⟨ definition ⟩
$$\llbracket \Gamma \vdash^{p} \theta^{p}(e) : A \rrbracket_{p} \mathbin{;} \eta_{\square A}$$
$\curlyeqprec$ ⟨ definition ⟩
$$\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \square\llbracket \Gamma^{p} \vdash \theta^{p}(e) : A \rrbracket \mathbin{;} \phi_A \mathbin{;} \eta_{\square A}$$
$\curlyeqprec$ ⟨ induction hypothesis ⟩
$$\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \square(\llbracket \Gamma^{p} \vdash \theta^{p} : \Delta^{p} \rrbracket \mathbin{;} \llbracket \Delta^{p} \vdash e : A \rrbracket) \mathbin{;} \phi_A \mathbin{;} \eta_{\square A}$$
$\curlyeqprec$ ⟨ $\square$ preserves composition ⟩
$$\rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \square\llbracket \Gamma^{p} \vdash \theta^{p} : \Delta^{p} \rrbracket \mathbin{;} \square\llbracket \Delta^{p} \vdash e : A \rrbracket \mathbin{;} \phi_A \mathbin{;} \eta_{\square A}$$
$\curlyeqprec$ ⟨ lemma C.7 ⟩
$$\llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \rho(\Delta) \mathbin{;} \mathcal{M}(\Delta) \mathbin{;} \square\llbracket \Delta^{p} \vdash e : A \rrbracket \mathbin{;} \phi_A \mathbin{;} \eta_{\square A}$$
$\curlyeqprec$ ⟨ definition ⟩
$$\llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash^{p} e : A \rrbracket_{p} \mathbin{;} \eta_{\square A}$$
$\curlyeqprec$ ⟨ definition ⟩
$$\llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash \mathsf{box}\,\boxed{e} : \square A \rrbracket$$

$$\diamond\ \frac{\Gamma \vdash e_1 : \square A \qquad \Gamma, x : A^{p} \vdash e_2 : B}{\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e_1\ \mathsf{in}\ e_2 : B}\ \square\mathrm{E}$$

$$
\begin{aligned}
& \llbracket \Gamma \vdash \theta(\text{let box}\ \boxed{x} = e_1\ \text{in}\ e_2) : B \rrbracket \\
\curlyeqprec\ & \langle\ \text{definition}\ \rangle \\
& \llbracket \Gamma \vdash \text{let box}\ \boxed{y} = \theta(e_1)\ \text{in}\ \langle \theta, y^p/x \rangle(e_2) : B \rrbracket \\
\curlyeqprec\ & \langle\ \text{definition}\ \rangle \\
& \langle id_\Gamma , \llbracket \Gamma \vdash \theta(e_1) : A \rrbracket \rangle \,;\, \tau_{\Gamma,\Box A} \,;\, T\llbracket \Gamma, y : A^p \vdash \langle \theta, y^p/x \rangle(e_2) : B \rrbracket \,;\, \mu_B \\
\curlyeqprec\ & \langle\ \text{induction hypothesis}\ \rangle \\
& \langle id_\Gamma , \llbracket \Gamma \vdash \theta : \Delta \rrbracket \,;\, \llbracket \Delta \vdash e_1 : A \rrbracket \rangle \,;\, \tau_{\Gamma,\Box A} \\
& \;;\, T(\llbracket \Gamma, y : A^p \vdash \langle \theta, y^p/x \rangle : \Delta, x : A^p \rrbracket \,;\, \llbracket \Delta, x : A^p \vdash e_2 : B \rrbracket) \,;\, \mu_B \\
\curlyeqprec\ & \langle\ T\ \text{preserves composition}\ \rangle \\
& \langle id_\Gamma , \llbracket \Gamma \vdash \theta : \Delta \rrbracket \,;\, \llbracket \Delta \vdash e_1 : A \rrbracket \rangle \,;\, \tau_{\Gamma,\Box A} \\
& \;;\, T\llbracket \Gamma, y : A^p \vdash \langle \theta, y^p/x \rangle : \Delta, x : A^p \rrbracket \,;\, T\llbracket \Delta, x : A^p \vdash e_2 : B \rrbracket \,;\, \mu_B \\
\curlyeqprec\ & \langle\ \text{definition}\ \rangle \\
& \langle id_\Gamma , \llbracket \Gamma \vdash \theta : \Delta \rrbracket \,;\, \llbracket \Delta \vdash e_1 : A \rrbracket \rangle \,;\, \tau_{\Gamma,\Box A} \\
& \;;\, T\langle \llbracket \Gamma, y : A^p \vdash \theta : \Delta \rrbracket , \llbracket \Gamma, y : A^p \vdash^p y : A \rrbracket_p \rangle \\
& \;;\, T\llbracket \Delta, x : A^p \vdash e_2 : B \rrbracket \,;\, \mu_B \\
\curlyeqprec\ & \langle\ \text{lemma C.5}\ \rangle \\
& \langle id_\Gamma , \llbracket \Gamma \vdash \theta : \Delta \rrbracket \,;\, \llbracket \Delta \vdash e_1 : A \rrbracket \rangle \,;\, \tau_{\Gamma,\Box A} \\
& \;;\, T\langle \mathsf{Wk}(\Gamma, y : A^p \supseteq \Gamma) \,;\, \llbracket \Gamma \vdash \theta : \Delta \rrbracket , \llbracket \Gamma, y : A^p \vdash^p y : A \rrbracket_p \rangle \\
& \;;\, T\llbracket \Delta, x : A^p \vdash e_2 : B \rrbracket \,;\, \mu_B \\
\curlyeqprec\ & \langle\ \text{definition}\ \rangle \\
& \langle id_\Gamma , \llbracket \Gamma \vdash \theta : \Delta \rrbracket \,;\, \llbracket \Delta \vdash e_1 : A \rrbracket \rangle \,;\, \tau_{\Gamma,\Box A} \\
& \;;\, T\langle \pi_1 \,;\, \llbracket \Gamma \vdash \theta : \Delta \rrbracket , \pi_2 \rangle \,;\, T\llbracket \Delta, x : A^p \vdash e_2 : B \rrbracket \,;\, \mu_B \\
\curlyeqprec\ & \langle\ \text{universal property of products}\ \rangle \\
& \langle id_\Gamma , \llbracket \Gamma \vdash \theta : \Delta \rrbracket \,;\, \llbracket \Delta \vdash e_1 : A \rrbracket \rangle \,;\, \tau_{\Gamma,\Box A} \\
& \;;\, T[\llbracket \Gamma \vdash \theta : \Delta \rrbracket \times id_{\Box A}] \,;\, T\llbracket \Delta, x : A^p \vdash e_2 : B \rrbracket \,;\, \mu_B \\
\curlyeqprec\ & \langle\ \text{tensorial strength of}\ T\ \rangle \\
& \langle \llbracket \Gamma \vdash \theta : \Delta \rrbracket \,;\, id_\Delta , \llbracket \Gamma \vdash \theta : \Delta \rrbracket \,;\, \llbracket \Delta \vdash e_1 : A \rrbracket \rangle \,;\, \tau_{\Delta,\Box A} \\
& \;;\, T\llbracket \Delta, x : A^p \vdash e_2 : B \rrbracket \,;\, \mu_B \\
\curlyeqprec\ & \langle\ \text{universal property of products}\ \rangle \\
& \llbracket \Gamma \vdash \theta : \Delta \rrbracket \,;\, \langle id_\Delta , \llbracket \Delta \vdash e_1 : A \rrbracket \rangle \,;\, \tau_{\Delta,\Box A} \,;\, T\llbracket \Delta, x : A^p \vdash e_2 : B \rrbracket \,;\, \mu_B \\
\curlyeqprec\ & \langle\ \text{definition}\ \rangle \\
& \llbracket \Gamma \vdash \theta : \Delta \rrbracket \,;\, \llbracket \Delta \vdash \text{let box}\ \boxed{x} = e_1\ \text{in}\ e_2 : B \rrbracket
\end{aligned}
$$

$$\diamond\ \frac{\Gamma, x : A^{i} \vdash e : B}{\Gamma \vdash \lambda x : A.\, e : A \Rightarrow B}\ \Rightarrow\mathrm{I}$$

$$\begin{aligned}
& \boxed{[\![\Gamma \vdash \theta(\lambda x.\, e) : A \Rightarrow B]\!]} \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \boxed{[\![\Gamma \vdash \lambda y.\, \langle \theta, y^{i}/x \rangle(e) : A \Rightarrow B]\!]} \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \boxed{\mathsf{curry}\,([\![\Gamma, y : A^{i} \vdash \langle \theta, y^{i}/x \rangle(e) : B]\!]) \mathbin{;} \eta_{A \to TB}} \\
\preceq\ & \langle\ \text{induction hypothesis}\ \rangle \\
& \boxed{\begin{array}{l}\mathsf{curry}\,([\![\Gamma, y : A^{i} \vdash \langle \theta, y^{i}/x \rangle : \Delta, x : A^{i}]\!] \mathbin{;} [\![\Delta, x : A^{i} \vdash e : B]\!]) \\ \mathbin{;} \eta_{A \to TB}\end{array}} \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \boxed{\begin{array}{l}\mathsf{curry}\,(\langle [\![\Gamma, y : A^{i} \vdash \theta : \Delta]\!] , [\![\Gamma, y : A^{i} \vdash y : A]\!]_{v} \rangle \mathbin{;} [\![\Delta, x : A^{i} \vdash e : B]\!]) \\ \mathbin{;} \eta_{A \to TB}\end{array}} \\
\preceq\ & \langle\ \text{lemma C.5}\ \rangle \\
& \boxed{\begin{array}{l}\mathsf{curry}\,(\langle \mathsf{Wk}(\Gamma, y : A^{i} \supseteq \Gamma) \mathbin{;} [\![\Gamma \vdash \theta : \Delta]\!] , \pi_2 \rangle \mathbin{;} [\![\Delta, x : A^{i} \vdash e : B]\!]) \\ \mathbin{;} \eta_{A \to TB}\end{array}} \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \boxed{\mathsf{curry}\,(\langle \pi_1 \mathbin{;} [\![\Gamma \vdash \theta : \Delta]\!] , \pi_2 \rangle \mathbin{;} [\![\Delta, x : A^{i} \vdash e : B]\!]) \mathbin{;} \eta_{A \to TB}} \\
\preceq\ & \langle\ \text{universal property of products}\ \rangle \\
& \boxed{\mathsf{curry}\,([\,[\![\Gamma \vdash \theta : \Delta]\!] \times id_{A}\,] \mathbin{;} [\![\Delta, x : A^{i} \vdash e : B]\!]) ; \eta_{A \to TB}} \\
\preceq\ & \langle\ \text{universal property of exponential}\ \rangle \\
& \boxed{[\![\Gamma \vdash \theta : \Delta]\!] \mathbin{;} \mathsf{curry}\,([\![\Delta, x : A^{i} \vdash e : B]\!]) \mathbin{;} \eta_{A \to TB}} \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \boxed{[\![\Gamma \vdash \theta : \Delta]\!] \mathbin{;} [\![\Delta \vdash \lambda x.\, e : A \Rightarrow B]\!]}
\end{aligned}$$

$$\diamond\ \frac{\Gamma \vdash e_1 : A \Rightarrow B \qquad \Gamma \vdash e_2 : A}{\Gamma \vdash e_1\, e_2 : B}\ \Rightarrow\mathrm{E}$$

$$\begin{aligned}
& \boxed{[\![\Gamma \vdash \theta(e_1\, e_2) : B]\!]} \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \boxed{[\![\Gamma \vdash \theta(e_1)\, \theta(e_2) : B]\!]} \\
\preceq\ & \langle\ \text{definition}\ \rangle \\
& \boxed{\langle [\![\Gamma \vdash \theta(e_1) : A \Rightarrow B]\!] , [\![\Gamma \vdash \theta(e_2) : A]\!] \rangle \mathbin{;} \beta_{A \to TB, A} \mathbin{;} T\,\mathsf{ev}_{A, TB} \mathbin{;} \mu_{B}}
\end{aligned}$$

$\preccurlyeq$ ⟨ induction hypothesis ⟩

$$\langle \llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash e_1 : A \Rightarrow B \rrbracket , \llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash e_2 : A \rrbracket \rangle \mathbin{;} \beta_{A \to TB, A} \mathbin{;} T\,\mathsf{ev}_{A,TB} \mathbin{;} \mu_B$$

$\preccurlyeq$ ⟨ universal property of products ⟩

$$\llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \langle \llbracket \Delta \vdash e_1 : A \Rightarrow B \rrbracket , \llbracket \Delta \vdash e_2 : A \rrbracket \rangle \mathbin{;} \beta_{A \to TB, A} \mathbin{;} T\,\mathsf{ev}_{A,TB} \mathbin{;} \mu_B$$

$\preccurlyeq$ ⟨ definition ⟩

$$\llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash e_1\, e_2 : B \rrbracket$$

$$\diamond \quad \frac{}{\Gamma \vdash s : \mathsf{str}}\ \mathsf{strI}$$

$$\llbracket \Gamma \vdash \theta(s) : \mathsf{str} \rrbracket$$

$\preccurlyeq$ ⟨ definition ⟩

$$\llbracket \Gamma \vdash s : \mathsf{str} \rrbracket$$

$\preccurlyeq$ ⟨ definition ⟩

$$\llbracket \Gamma \vdash \langle\rangle : \cdot \rrbracket \mathbin{;} \llbracket \cdot \vdash s : \mathsf{str} \rrbracket$$

$\preccurlyeq$ ⟨ universal property of 1 ⟩

$$\llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash \langle\rangle : \cdot \rrbracket \mathbin{;} \llbracket \cdot \vdash s : \mathsf{str} \rrbracket$$

$\preccurlyeq$ ⟨ definition ⟩

$$\llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash s : \mathsf{str} \rrbracket$$

$$\diamond \quad \frac{\Gamma \vdash e_1 : \mathsf{cap} \qquad \Gamma \vdash e_2 : \mathsf{str}}{\Gamma \vdash e_1 \cdot \mathsf{print}(e_2) : \mathsf{unit}}\ \textsc{Print}$$

$$\llbracket \Gamma \vdash \theta(e_1 \cdot \mathsf{print}(e_2)) : \mathsf{unit} \rrbracket$$

$\preccurlyeq$ ⟨ definition ⟩

$$\llbracket \Gamma \vdash \theta(e_1) \cdot \mathsf{print}(\theta(e_2)) : \mathsf{unit} \rrbracket$$

$\preccurlyeq$ ⟨ definition ⟩

$$\langle \llbracket \Gamma \vdash \theta(e_1) : \mathsf{cap} \rrbracket , \llbracket \Gamma \vdash \theta(e_2) : \mathsf{str} \rrbracket \rangle \mathbin{;} \beta_{C,\Sigma^*} \mathbin{;} Tp \mathbin{;} \mu_1$$

$\preccurlyeq$ ⟨ induction hypothesis ⟩

$$\langle \llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash e_1 : \mathsf{cap} \rrbracket , \llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash e_2 : \mathsf{str} \rrbracket \rangle \mathbin{;} \beta_{C,\Sigma^*} \mathbin{;} Tp \mathbin{;} \mu_1$$

$\preccurlyeq$ ⟨ universal property of products ⟩

$$\llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \langle \llbracket \Delta \vdash e_1 : \mathsf{cap} \rrbracket , \llbracket \Delta \vdash e_2 : \mathsf{str} \rrbracket \rangle \mathbin{;} \beta_{C,\Sigma^*} \mathbin{;} Tp \mathbin{;} \mu_1$$

$\preccurlyeq$ ⟨ definition ⟩

$$\llbracket \Gamma \vdash \theta : \Delta \rrbracket \mathbin{;} \llbracket \Delta \vdash e_1 \cdot \mathsf{print}(e_2) : \mathsf{unit} \rrbracket$$

□

D PROOFS FOR SECTION 6 (EQUATIONAL THEORY)

Theorem 6.1 Soundness of $\approx$. *If* $\Gamma \vdash e_1 \approx e_2 : A$, *then* $[\![\Gamma \vdash e_1 : A]\!] = [\![\Gamma \vdash e_2 : A]\!]$.

Proof. We proceed by induction on $\Gamma \vdash e_1 \approx e_2 : A$.

◇ $$\frac{\Gamma \vdash e : A}{\Gamma \vdash e \approx e : A}\ \textsc{refl}$$

$$\boxed{[\![\Gamma \vdash e : A]\!]}$$
$\eqsim\langle$ reflexivity $\rangle$
$$\boxed{[\![\Gamma \vdash e : A]\!]}$$

◇ $$\frac{\Gamma \vdash e_1 \approx e_2 : A}{\Gamma \vdash e_2 \approx e_1 : A}\ \textsc{sym}$$

$$\boxed{[\![\Gamma \vdash e_2 : A]\!]}$$
$\eqsim\langle$ induction hypothesis $\rangle$
$$\boxed{[\![\Gamma \vdash e_1 : A]\!]}$$

◇ $$\frac{\Gamma \vdash e_1 \approx e_2 : A \qquad \Gamma \vdash e_2 \approx e_3 : A}{\Gamma \vdash e_1 \approx e_3 : A}\ \textsc{trans}$$

$$\boxed{[\![\Gamma \vdash e_1 : A]\!]}$$
$\eqsim\langle$ induction hypothesis $\rangle$
$$\boxed{[\![\Gamma \vdash e_2 : A]\!]}$$
$\eqsim\langle$ induction hypothesis $\rangle$
$$\boxed{[\![\Gamma \vdash e_3 : A]\!]}$$

◇ $$\frac{\Gamma \vdash e_1 \approx e_2 : A \times B}{\Gamma \vdash \mathsf{fst}\, e_1 \approx \mathsf{fst}\, e_2 : A}\ \mathsf{fst}\text{-}\textsc{cong}$$

$$\boxed{[\![\Gamma \vdash \mathsf{fst}\, e_1 : A]\!]}$$
$\eqsim\langle$ definition $\rangle$
$$\boxed{[\![\Gamma \vdash e_1 : A \times B]\!] \,;\, T\pi_1}$$
$\eqsim\langle$ induction hypothesis $\rangle$
$$\boxed{[\![\Gamma \vdash e_2 : A \times B]\!] \,;\, T\pi_1}$$
$\eqsim\langle$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash \mathsf{fst}\, e_2 : A]\!]}$$

$$\diamond \frac{\Gamma \vdash e_1 \approx e_2 : A \times B}{\Gamma \vdash \mathsf{snd}\, e_1 \approx \mathsf{snd}\, e_2 : B}\ \mathsf{snd}\text{-}\textsc{cong}$$

$$\begin{aligned}
& \boxed{[\![\Gamma \vdash \mathsf{snd}\, e_1 : B]\!]} \\
\eqsim\langle\ & \text{definition} \ \rangle \\
& \boxed{[\![\Gamma \vdash e_1 : A \times B]\!] \mathbin{;} T\pi_2} \\
\eqsim\langle\ & \text{induction hypothesis} \ \rangle \\
& \boxed{[\![\Gamma \vdash e_2 : A \times B]\!] \mathbin{;} T\pi_2} \\
\eqsim\langle\ & \text{definition} \ \rangle \\
& \boxed{[\![\Gamma \vdash \mathsf{snd}\, e_2 : B]\!]}
\end{aligned}$$

$$\diamond \frac{\Gamma \vdash e_1 \approx e_2 : A \qquad \Gamma \vdash e_3 \approx e_4 : B}{\Gamma \vdash (e_1 , e_3) \approx (e_2 , e_4) : A \times B}\ \textsc{pair-cong}$$

$$\begin{aligned}
& \boxed{[\![\Gamma \vdash (e_1 , e_3) : A \times B]\!]} \\
\eqsim\langle\ & \text{definition} \ \rangle \\
& \boxed{\langle [\![\Gamma \vdash e_1 : A]\!] , [\![\Gamma \vdash e_3 : B]\!] \rangle \mathbin{;} \beta_{A,B}} \\
\eqsim\langle\ & \text{induction hypothesis} \ \rangle \\
& \boxed{\langle [\![\Gamma \vdash e_2 : A]\!] , [\![\Gamma \vdash e_4 : B]\!] \rangle \mathbin{;} \beta_{A,B}} \\
\eqsim\langle\ & \text{definition} \ \rangle \\
& \boxed{[\![\Gamma \vdash (e_2 , e_4) : A \times B]\!]}
\end{aligned}$$

$$\diamond \frac{\Gamma, x : A^{i} \vdash e_1 \approx e_2 : B}{\Gamma \vdash \lambda x : A.\, e_1 \approx \lambda x : A.\, e_2 : A \Rightarrow B}\ \lambda\text{-}\textsc{cong}$$

$$\begin{aligned}
& \boxed{[\![\Gamma \vdash \lambda x.\, e_1 : A \Rightarrow B]\!]} \\
\eqsim\langle\ & \text{definition} \ \rangle \\
& \boxed{\mathsf{curry}\,([\![\Gamma, x : A^{i} \vdash e_1 : B]\!]) \mathbin{;} \eta_{A \to TB}} \\
\eqsim\langle\ & \text{induction hypothesis} \ \rangle \\
& \boxed{\mathsf{curry}\,([\![\Gamma, x : A^{i} \vdash e_2 : B]\!]) \mathbin{;} \eta_{A \to TB}} \\
\eqsim\langle\ & \text{definition} \ \rangle \\
& \boxed{[\![\Gamma \vdash \lambda x.\, e_2 : A \Rightarrow B]\!]}
\end{aligned}$$

$$\diamond\quad \frac{\Gamma \vdash e_1 \approx e_2 : A \Rightarrow B \qquad \Gamma \vdash e_3 \approx e_4 : A}{\Gamma \vdash e_1\, e_3 \approx e_2\, e_4 : B}\ \textsc{app-cong}$$

$$\begin{array}{ll}
& [\![\Gamma \vdash e_1\, e_3 : B]\!] \\
\preccurlyeq & \langle\ \text{definition}\ \rangle \\
& \langle [\![\Gamma \vdash e_1 : A \Rightarrow B]\!] , [\![\Gamma \vdash e_3 : A]\!] \rangle \mathbin{;} \beta_{A \to TB, A} \mathbin{;} T\,\mathrm{ev}_{A,TB} \mathbin{;} \mu_B \\
\preccurlyeq & \langle\ \text{induction hypothesis}\ \rangle \\
& \langle [\![\Gamma \vdash e_2 : A \Rightarrow B]\!] , [\![\Gamma \vdash e_4 : A]\!] \rangle \mathbin{;} \beta_{A \to TB, A} \mathbin{;} T\,\mathrm{ev}_{A,TB} \mathbin{;} \mu_B \\
\preccurlyeq & \langle\ \text{definition}\ \rangle \\
& [\![\Gamma \vdash e_2\, e_4 : B]\!]
\end{array}$$

$$\diamond\quad \frac{\Gamma^{p} \vdash e_1 \approx e_2 : A}{\Gamma \vdash \mathsf{box}\,\boxed{e_1} \approx \mathsf{box}\,\boxed{e_2} : \Box A}\ \textsc{box-cong}$$

$$\begin{array}{ll}
& [\![\Gamma \vdash \mathsf{box}\,\boxed{e_1} : \Box A]\!] \\
\preccurlyeq & \langle\ \text{definition}\ \rangle \\
& [\![\Gamma \vdash^{p} e_1 : A]\!]_p \mathbin{;} \eta_{\Box A} \\
\preccurlyeq & \langle\ \text{definition}\ \rangle \\
& \rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \Box[\![\Gamma^{p} \vdash e_1 : A]\!] \mathbin{;} \phi_A \mathbin{;} \eta_{\Box A} \\
\preccurlyeq & \langle\ \text{induction hypothesis}\ \rangle \\
& \rho(\Gamma) \mathbin{;} \mathcal{M}(\Gamma) \mathbin{;} \Box[\![\Gamma^{p} \vdash e_2 : A]\!] \mathbin{;} \phi_A \mathbin{;} \eta_{\Box A} \\
\preccurlyeq & \langle\ \text{definition}\ \rangle \\
& [\![\Gamma \vdash^{p} e_2 : A]\!]_p \mathbin{;} \eta_{\Box A} \\
\preccurlyeq & \langle\ \text{definition}\ \rangle \\
& [\![\Gamma \vdash \mathsf{box}\,\boxed{e_2} : \Box A]\!]
\end{array}$$

$$\diamond\quad \frac{\Gamma \vdash e_1 \approx e_2 : \Box A \qquad \Gamma, x : A^{p} \vdash e_3 \approx e_4 : B}{\Gamma \vdash (\mathsf{let\ box}\,\boxed{x} = e_1\ \mathsf{in}\ e_3) \approx (\mathsf{let\ box}\,\boxed{x} = e_2\ \mathsf{in}\ e_4) : B}\ \text{let box-}\textsc{cong}$$

$$\begin{array}{ll}
& [\![\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e_1\ \mathsf{in}\ e_3 : B]\!] \\
\preccurlyeq & \langle\ \text{definition}\ \rangle \\
& \langle id_\Gamma , [\![\Gamma \vdash e_1 : \Box A]\!] \rangle \mathbin{;} \tau_{\Gamma, \Box A} \mathbin{;} T[\![\Gamma, x : A^{p} \vdash e_3 : B]\!] \mathbin{;} \mu_B \\
\preccurlyeq & \langle\ \text{induction hypothesis}\ \rangle \\
& \langle id_\Gamma , [\![\Gamma \vdash e_2 : \Box A]\!] \rangle \mathbin{;} \tau_{\Gamma, \Box A} \mathbin{;} T[\![\Gamma, x : A^{p} \vdash e_4 : B]\!] \mathbin{;} \mu_B \\
\preccurlyeq & \langle\ \text{definition}\ \rangle
\end{array}$$

$$[\![\Gamma \vdash \mathsf{let\ box}\ \boxed{x} = e_2\ \mathsf{in}\ e_4 : B]\!]$$

$$\diamond\ \frac{\Gamma \vdash e_1 \approx e_2 : \mathsf{cap} \qquad \Gamma \vdash e_3 \approx e_4 : \mathsf{str}}{\Gamma \vdash e_1 \cdot \mathsf{print}(e_3) \approx e_2 \cdot \mathsf{print}(e_4) : \mathsf{unit}}\ \text{print-CONG}$$

$$\begin{aligned}
& [\![\Gamma \vdash e_1 \cdot \mathsf{print}(e_3) : \mathsf{unit}]\!] \\
\preceq\!\!\!\!= \langle\ & \text{definition}\ \rangle \\
& \langle [\![\Gamma \vdash e_1 : \mathsf{cap}]\!] , [\![\Gamma \vdash e_3 : \mathsf{str}]\!] \rangle ; \beta_{C,\Sigma^*} ; Tp ; \mu_1 \\
\preceq\!\!\!\!= \langle\ & \text{induction hypothesis}\ \rangle \\
& \langle [\![\Gamma \vdash e_2 : \mathsf{cap}]\!] , [\![\Gamma \vdash e_4 : \mathsf{str}]\!] \rangle ; \beta_{C,\Sigma^*} ; Tp ; \mu_1 \\
\preceq\!\!\!\!= \langle\ & \text{definition}\ \rangle \\
& [\![\Gamma \vdash e_2 \cdot \mathsf{print}(e_4) : \mathsf{unit}]\!]
\end{aligned}$$

$$\diamond\ \frac{\Gamma \vdash v_1 : A \qquad \Gamma \vdash v_2 : B}{\Gamma \vdash \mathsf{fst}\ (v_1 , v_2) \approx v_1 : A}\ \times_1\beta$$

$$\begin{aligned}
& [\![\Gamma \vdash \mathsf{fst}\ (v_1 , v_2) : A]\!] \\
\preceq\!\!\!\!= \langle\ & \text{definition}\ \rangle \\
& [\![\Gamma \vdash (v_1 , v_2) : A \times B]\!] ; T\pi_1 \\
\preceq\!\!\!\!= \langle\ & \text{value interpretation lemma 5.10}\ \rangle \\
& [\![\Gamma \vdash (v_1 , v_2) : A \times B]\!]_v ; \eta_{A\times B} ; T\pi_1 \\
\preceq\!\!\!\!= \langle\ & \text{monad laws}\ \rangle \\
& [\![\Gamma \vdash (v_1 , v_2) : A \times B]\!]_v ; \pi_1 ; \eta_A \\
\preceq\!\!\!\!= \langle\ & \text{definition}\ \rangle \\
& \langle [\![\Gamma \vdash v_1 : A]\!]_v , [\![\Gamma \vdash v_2 : B]\!]_v \rangle ; \pi_1 ; \eta_A \\
\preceq\!\!\!\!= \langle\ & \text{definition of } \pi_1\ \rangle \\
& [\![\Gamma \vdash v_1 : A]\!]_v ; \eta_A \\
\preceq\!\!\!\!= \langle\ & \text{value interpretation lemma 5.10}\ \rangle \\
& [\![\Gamma \vdash v_1 : A]\!]
\end{aligned}$$

$$\diamond\ \frac{\Gamma \vdash v_1 : A \qquad \Gamma \vdash v_2 : B}{\Gamma \vdash \mathsf{snd}\ (v_1 , v_2) \approx v_2 : B}\ \times_2\beta$$

$$\begin{aligned}
& [\![\Gamma \vdash \mathsf{snd}\ (v_1 , v_2) : B]\!] \\
\preceq\!\!\!\!= \langle\ & \text{definition}\ \rangle \\
& [\![\Gamma \vdash (v_1 , v_2) : A \times B]\!] ; T\pi_2
\end{aligned}$$

$\preceq\langle$ value interpretation lemma 5.10 $\rangle$

$\boxed{[\![\Gamma \vdash (v_1\,, v_2) : A \times B]\!]_v \mathbin{;} \eta_{A\times B} \mathbin{;} T\pi_2}$

$\preceq\langle$ monad laws $\rangle$

$\boxed{[\![\Gamma \vdash (v_1\,, v_2) : A \times B]\!]_v \mathbin{;} \pi_2 \mathbin{;} \eta_B}$

$\preceq\langle$ definition $\rangle$

$\boxed{\langle [\![\Gamma \vdash v_1 : A]\!]_v\,, [\![\Gamma \vdash v_2 : B]\!]_v \rangle \mathbin{;} \pi_2 \mathbin{;} \eta_B}$

$\preceq\langle$ definition of π_2 $\rangle$

$\boxed{[\![\Gamma \vdash v_2 : B]\!]_v \mathbin{;} \eta_B}$

$\preceq\langle$ value interpretation lemma 5.10 $\rangle$

$\boxed{[\![\Gamma \vdash v_2 : B]\!]}$

$$\diamond\ \dfrac{\Gamma \vdash v : A \times B}{\Gamma \vdash v \approx (\mathsf{fst}\, v\,, \mathsf{snd}\, v) : A \times B}\ \times\eta$$

$\boxed{[\![\Gamma \vdash (\mathsf{fst}\, v\,, \mathsf{snd}\, v) : A \times B]\!]}$

$\preceq\langle$ definition $\rangle$

$\boxed{\langle [\![\Gamma \vdash \mathsf{fst}\, v : A]\!]\,, [\![\Gamma \vdash \mathsf{snd}\, v : B]\!] \rangle \mathbin{;} \beta_{A,B}}$

$\preceq\langle$ definition $\rangle$

$\boxed{\langle [\![\Gamma \vdash v : A \times B]\!] \mathbin{;} T\pi_1\,, [\![\Gamma \vdash v : A \times B]\!] \mathbin{;} T\pi_2 \rangle \mathbin{;} \beta_{A,B}}$

$\preceq\langle$ value interpretation lemma 5.10 $\rangle$

$\boxed{\langle [\![\Gamma \vdash v : A \times B]\!]_v \mathbin{;} \eta_{A\times B} \mathbin{;} T\pi_1\,, [\![\Gamma \vdash v : A \times B]\!]_v \mathbin{;} \eta_{A\times B} \mathbin{;} T\pi_2 \rangle \mathbin{;} \beta_{A,B}}$

$\preceq\langle$ monad laws $\rangle$

$\boxed{\langle [\![\Gamma \vdash v : A \times B]\!]_v \mathbin{;} \pi_1 \mathbin{;} \eta_A\,, [\![\Gamma \vdash v : A \times B]\!]_v \mathbin{;} \pi_2 \mathbin{;} \eta_B \rangle \mathbin{;} \beta_{A,B}}$

$\preceq\langle$ universal property of products $\rangle$

$\boxed{[\![\Gamma \vdash v : A \times B]\!]_v \mathbin{;} \langle \pi_1 \mathbin{;} \eta_A\,, \pi_2 \mathbin{;} \eta_B \rangle \mathbin{;} \beta_{A,B}}$

$\preceq\langle$ universal property of products $\rangle$

$\boxed{[\![\Gamma \vdash v : A \times B]\!]_v \mathbin{;} [\eta_A \times \eta_B] \mathbin{;} \beta_{A,B}}$

$\preceq\langle$ diagram $\rangle$

$\boxed{[\![\Gamma \vdash v : A \times B]\!]_v \mathbin{;} \eta_{A\times B}}$

$\preceq\langle$ value interpretation lemma 5.10 $\rangle$

$\boxed{[\![\Gamma \vdash v : A \times B]\!]}$

$$\diamond\ \dfrac{\Gamma, x : A^{i} \vdash e : B \qquad \Gamma \vdash v : A}{\Gamma \vdash (\lambda x : A.\, e)\, v \approx [v/x]e : B}\ \Rightarrow\beta$$

$$\boxed{[\![\Gamma \vdash (\lambda x.\, e)\, v : B]\!]}$$

$\sqsupseteq\langle$ definition $\rangle$

$$\boxed{\langle [\![\Gamma \vdash \lambda x.\, e : A \Rightarrow B]\!] \,,\, [\![\Gamma \vdash v : A]\!] \rangle \,;\, \beta_{A \to TB, A} \,;\, T\,\mathsf{ev}_{A,TB} \,;\, \mu_B}$$

$\sqsupseteq\langle$ definition $\rangle$

$$\boxed{\begin{array}{l}\langle \mathsf{curry}\,([\![\Gamma, x : A^{i} \vdash e : B]\!]) \,;\, \eta_{A \to TB} \,,\, [\![\Gamma \vdash v : A]\!] \rangle \\ \;;\, \beta_{A \to TB, A} \,;\, T\,\mathsf{ev}_{A,TB} \,;\, \mu_B\end{array}}$$

$\sqsupseteq\langle$ value interpretation lemma 5.10 $\rangle$

$$\boxed{\begin{array}{l}\langle \mathsf{curry}\,([\![\Gamma, x : A^{i} \vdash e : B]\!]) \,;\, \eta_{A \to TB} \,,\, [\![\Gamma \vdash v : A]\!]_{v} \,;\, \eta_A \rangle \\ \;;\, \beta_{A \to TB, A} \,;\, T\,\mathsf{ev}_{A,TB} \,;\, \mu_B\end{array}}$$

$\sqsupseteq\langle$ universal property of products $\rangle$

$$\boxed{\begin{array}{l}\langle \mathsf{curry}\,([\![\Gamma, x : A^{i} \vdash e : B]\!]) \,,\, [\![\Gamma \vdash v : A]\!]_{v} \rangle \\ \;;\, [\eta_{A \to TB} \times \eta_A] \,;\, \beta_{A \to TB, A} \,;\, T\,\mathsf{ev}_{A,TB} \,;\, \mu_B\end{array}}$$

$\sqsupseteq\langle$ diagram $\rangle$

$$\boxed{\begin{array}{l}\langle \mathsf{curry}\,([\![\Gamma, x : A^{i} \vdash e : B]\!]) \,,\, [\![\Gamma \vdash v : A]\!]_{v} \rangle \\ \;;\, \eta_{(A \to TB) \times A} \,;\, T\,\mathsf{ev}_{A,TB} \,;\, \mu_B\end{array}}$$

$\sqsupseteq\langle$ monad laws $\rangle$

$$\boxed{\langle \mathsf{curry}\,([\![\Gamma, x : A^{i} \vdash e : B]\!]) \,,\, [\![\Gamma \vdash v : A]\!]_{v} \rangle \,;\, \mathsf{ev}_{A,TB}}$$

$\sqsupseteq\langle$ universal property of exponential $\rangle$

$$\boxed{\langle id_{\Gamma} \,,\, [\![\Gamma \vdash v : A]\!]_{v} \rangle \,;\, [\![\Gamma, x : A^{i} \vdash e : B]\!]}$$

$\sqsupseteq\langle$ definition $\rangle$

$$\boxed{\langle [\![\Gamma \vdash \langle \Gamma \rangle : \Gamma]\!] \,,\, [\![\Gamma \vdash v : A]\!]_{v} \rangle \,;\, [\![\Gamma, x : A^{i} \vdash e : B]\!]}$$

$\sqsupseteq\langle$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash \langle \langle \Gamma \rangle, v^{i}/x \rangle : \Gamma, x : A^{i}]\!] \,;\, [\![\Gamma, x : A^{i} \vdash e : B]\!]}$$

$\sqsupseteq\langle$ semantic substitution theorem 5.11 $\rangle$

$$\boxed{[\![\Gamma \vdash \langle \langle \Gamma \rangle, v^{i}/x \rangle (e) : B]\!]}$$

$\sqsupseteq\langle$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash [v/x]e : B]\!]}$$

$$\diamond\; \frac{\Gamma \vdash v : A \Rightarrow B}{\Gamma \vdash v \approx \lambda x : A.\, v\, x : A \Rightarrow B} \Rightarrow\eta\text{-}\textsc{impure}$$

$$\boxed{[\![\Gamma \vdash \lambda x.\, v\, x : A \Rightarrow B]\!]}$$

$\sqsupseteq\langle$ definition $\rangle$

$$\boxed{\mathsf{curry}\,([\![\Gamma, x : A^{i} \vdash v\, x : B]\!]) \,;\, \eta_{A \to TB}}$$

$= \langle$ definition $\rangle$

$$\begin{array}{lll} \textit{let} & h = & \beta_{A\to TB,A} \mathbin{;} T\,\mathsf{ev}_{A,TB} \mathbin{;} \mu_B \\ \textit{in} & \multicolumn{2}{l}{\mathsf{curry}\,(\langle [\![\Gamma, x : A^i \vdash v : A \Rightarrow B]\!] \,,\, [\![\Gamma, x : A^i \vdash x : A]\!]\rangle \mathbin{;} h) \mathbin{;} \eta_{A\to TB}} \end{array}$$

$= \langle$ semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{lll} & f = & \mathsf{Wk}(\Gamma, x : A^i \supseteq \Gamma) \\ \textit{let} & g = & [\![x : A^i \in \Gamma, x : A^i]\!] \\ & h = & \beta_{A\to TB,A} \mathbin{;} T\,\mathsf{ev}_{A,TB} \mathbin{;} \mu_B \\ \textit{in} & \multicolumn{2}{l}{\mathsf{curry}\,(\langle f \mathbin{;} [\![\Gamma \vdash v : A \Rightarrow B]\!] \,,\, g \mathbin{;} \eta_A\rangle \mathbin{;} h) \mathbin{;} \eta_{A\to TB}} \end{array}$$

$= \langle$ definition $\rangle$

$$\begin{array}{lll} \textit{let} & h = & \beta_{A\to TB,A} \mathbin{;} T\,\mathsf{ev}_{A,TB} \mathbin{;} \mu_B \\ \textit{in} & \multicolumn{2}{l}{\mathsf{curry}\,(\langle \pi_1 \mathbin{;} [\![\Gamma \vdash v : A \Rightarrow B]\!] \,,\, \pi_2 \mathbin{;} \eta_A\rangle \mathbin{;} h) \mathbin{;} \eta_{A\to TB}} \end{array}$$

$= \langle$ value interpretation lemma 5.10 $\rangle$

$$\begin{array}{lll} \textit{let} & h = & \beta_{A\to TB,A} \mathbin{;} T\,\mathsf{ev}_{A,TB} \mathbin{;} \mu_B \\ \textit{in} & \multicolumn{2}{l}{\mathsf{curry}\,(\langle \pi_1 \mathbin{;} [\![\Gamma \vdash v : A \Rightarrow B]\!]_v \mathbin{;} \eta_{A\to TB} \,,\, \pi_2 \mathbin{;} \eta_A\rangle \mathbin{;} h) \mathbin{;} \eta_{A\to TB}} \end{array}$$

$= \langle$ strength diagram and monad laws $\rangle$

$$\mathsf{curry}\,(\langle \pi_1 \mathbin{;} [\![\Gamma \vdash v : A \Rightarrow B]\!]_v \,,\, \pi_2\rangle \mathbin{;} \mathsf{ev}_{A,TB}) \mathbin{;} \eta_{A\to TB}$$

$= \langle$ universal property of products $\rangle$

$$\mathsf{curry}\,([\,[\![\Gamma \vdash v : A \Rightarrow B]\!]_v \times id_A\,] \mathbin{;} \mathsf{ev}_{A,TB}) \mathbin{;} \eta_{A\to TB}$$

$= \langle$ universal property of exponential $\rangle$

$$[\![\Gamma \vdash v : A \Rightarrow B]\!]_v \mathbin{;} \eta_{A\to TB}$$

$= \langle$ value interpretation lemma 5.10 $\rangle$

$$[\![\Gamma \vdash v : A \Rightarrow B]\!]$$

$$\diamond\ \frac{\Gamma \vdash^{p} e : A \Rightarrow B}{\Gamma \vdash e \approx \lambda x : A.\, e\, x : A \Rightarrow B} \Rightarrow\eta\text{-}\textsc{pure}$$

$$[\![\Gamma \vdash \lambda x.\, e\, x : A \Rightarrow B]\!]$$

$= \langle$ definition $\rangle$

$$\mathsf{curry}\,([\![\Gamma, x : A^i \vdash e\, x : B]\!]) \mathbin{;} \eta_{A\to TB}$$

$= \langle$ definition $\rangle$

$$\begin{array}{lll} \textit{let} & h = & \beta_{A\to TB,A} \mathbin{;} T\,\mathsf{ev}_{A,TB} \mathbin{;} \mu_B \\ \textit{in} & \multicolumn{2}{l}{\mathsf{curry}\,(\langle [\![\Gamma, x : A^i \vdash e : A \Rightarrow B]\!] \,,\, [\![\Gamma, x : A^i \vdash x : A]\!]\rangle \mathbin{;} h) \mathbin{;} \eta_{A\to TB}} \end{array}$$

$= \langle$ semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{l} let \quad \begin{array}{lcl} f & = & \mathsf{Wk}(\Gamma, x : A^{i} \supseteq \Gamma) \\ g & = & [\![x : A^{i} \in \Gamma, x : A^{i}]\!] \\ h & = & \beta_{A \to TB, A} ; T\,\mathsf{ev}_{A,TB} ; \mu_B \end{array} \\ in \quad \mathsf{curry}\,(\langle f ; [\![\Gamma \vdash e : A \Rightarrow B]\!] , g ; \eta_A \rangle ; h) ; \eta_{A \to TB} \end{array}$$

$\preceq\langle$ definition $\rangle$

$$\begin{array}{l} let \quad h = \beta_{A \to TB, A} ; T\,\mathsf{ev}_{A,TB} ; \mu_B \\ in \quad \mathsf{curry}\,(\langle \pi_1 ; [\![\Gamma \vdash e : A \Rightarrow B]\!] , \pi_2 ; \eta_A \rangle ; h) ; \eta_{A \to TB} \end{array}$$

$\preceq\langle$ pure interpretation lemma 5.9 $\rangle$

$$\begin{array}{l} let \quad h = \beta_{A \to TB, A} ; T\,\mathsf{ev}_{A,TB} ; \mu_B \\ in \quad \mathsf{curry}\,(\langle \pi_1 ; [\![\Gamma \vdash^{p} e : A \Rightarrow B]\!]_p ; \varepsilon_{A \to TB} ; \eta_{A \to TB} , \pi_2 ; \eta_A \rangle ; h) ; \eta_{A \to TB} \end{array}$$

$\preceq\langle$ diagram and monad laws $\rangle$

$$\mathsf{curry}\,(\langle \pi_1 ; [\![\Gamma \vdash^{p} e : A \Rightarrow B]\!]_p ; \varepsilon_{A \to TB} , \pi_2 \rangle ; \mathsf{ev}_{A,TB}) ; \eta_{A \to TB}$$

$\preceq\langle$ universal property of products $\rangle$

$$\mathsf{curry}\,([\,[\![\Gamma \vdash^{p} e : A \Rightarrow B]\!]_p ; \varepsilon_{A \to TB} \times id_A\,] ; \mathsf{ev}_{A,TB}) ; \eta_{A \to TB}$$

$\preceq\langle$ universal property of exponential $\rangle$

$$[\![\Gamma \vdash^{p} e : A \Rightarrow B]\!]_p ; \varepsilon_{A \to TB} ; \eta_{A \to TB}$$

$\preceq\langle$ pure interpretation lemma 5.9 $\rangle$

$$[\![\Gamma \vdash e : A \Rightarrow B]\!]$$

$$\diamond \quad \frac{\Gamma^{p} \vdash e_1 : A \qquad \Gamma, x : A^{p} \vdash e_2 : B}{\Gamma \vdash \mathsf{let\,box}\,\boxed{x} = \mathsf{box}\,\boxed{e_1}\,\mathsf{in}\,e_2 \approx [e_1/x]e_2 : B}\ \square\beta$$

$$[\![\Gamma \vdash \mathsf{let\,box}\,\boxed{x} = \mathsf{box}\,\boxed{e_1}\,\mathsf{in}\,e_2 : B]\!]$$

$\preceq\langle$ definition $\rangle$

$$\langle id_\Gamma , [\![\Gamma \vdash \mathsf{box}\,\boxed{e_1} : \square A]\!] \rangle ; \tau_{\Gamma, \square A} ; T[\![\Gamma, x : A^{p} \vdash e_2 : B]\!] ; \mu_B$$

$\preceq\langle$ definition $\rangle$

$$\langle id_\Gamma , [\![\Gamma \vdash^{p} e_1 : A]\!]_p ; \eta_{\square A} \rangle ; \tau_{\Gamma, \square A} ; T[\![\Gamma, x : A^{p} \vdash e_2 : B]\!] ; \mu_B$$

$\preceq\langle$ strength commutes with unit $\rangle$

$$\langle id_\Gamma , [\![\Gamma \vdash^{p} e_1 : A]\!]_p \rangle ; \eta_{\Gamma \times \square A} ; T[\![\Gamma, x : A^{p} \vdash e_2 : B]\!] ; \mu_B$$

$\preceq\langle$ monad laws $\rangle$

$$\langle id_\Gamma , [\![\Gamma \vdash^{p} e_1 : A]\!]_p \rangle ; [\![\Gamma, x : A^{p} \vdash e_2 : B]\!] ; \eta_{TB} ; \mu_B$$

$\preceq\langle$ monad laws $\rangle$

$$\langle id_\Gamma , [\![\Gamma \vdash^{p} e_1 : A]\!]_p \rangle ; [\![\Gamma, x : A^{p} \vdash e_2 : B]\!]$$

$\preceq\langle$ definition $\rangle$

$$\langle [\![\Gamma \vdash \langle \Gamma \rangle : \Gamma]\!] , [\![\Gamma \vdash^{p} e_1 : A]\!]_p \rangle \,;\, [\![\Gamma, x : A^{p} \vdash e_2 : B]\!]$$

$\mathrel{=\!\!\prec}$ definition $\rangle$

$$[\![\Gamma \vdash \langle \langle \Gamma \rangle, e_1{}^{p}/x \rangle : \Gamma, x : A^{p}]\!] \,;\, [\![\Gamma, x : A^{p} \vdash e_2 : B]\!]$$

$\mathrel{=\!\!\prec}$ semantic substitution theorem 5.11 $\rangle$

$$[\![\Gamma \vdash \langle \langle \Gamma \rangle, e_1{}^{p}/x \rangle (e_2) : B]\!]$$

$\mathrel{=\!\!\prec}$ definition $\rangle$

$$\Gamma \vdash [e_1/x]\, e_2 : B$$

$$\diamond\ \frac{\Gamma \vdash^{p} e : \Box A \qquad \Gamma \vdash C\langle\!\langle e \rangle\!\rangle : B \qquad \Gamma \vdash \text{let box}\,\boxed{x} = e \text{ in } C\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B}{\Gamma \vdash C\langle\!\langle e \rangle\!\rangle \approx \text{let box}\,\boxed{x} = e \text{ in } C\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B}\ \Box\eta\text{-PURE}$$

We first make the following observation.

Observation.

$$[\![\Gamma \vdash \text{let box}\,\boxed{x} = e \text{ in } C\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B]\!]$$

$\mathrel{=\!\!\prec}$ definition $\rangle$

$$\begin{array}{lll} \textit{let} & f & = [\![\Gamma \vdash e : \Box A]\!] \\ & g & = [\![\Gamma, x : A^{p} \vdash C\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ \textit{in} & \multicolumn{2}{l}{\langle id_\Gamma , f \rangle \,;\, \tau_{\Gamma,\Box A} \,;\, Tg \,;\, \mu_B} \end{array}$$

$\mathrel{=\!\!\prec}$ pure interpretation lemma 5.9 $\rangle$

$$\begin{array}{lll} \textit{let} & f & = [\![\Gamma \vdash^{p} e : \Box A]\!]_p \,;\, \varepsilon_{\Box A} \,;\, \eta_{\Box A} \\ & g & = [\![\Gamma, x : A^{p} \vdash C\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ \textit{in} & \multicolumn{2}{l}{\langle id_\Gamma , f \rangle \,;\, \tau_{\Gamma,\Box A} \,;\, Tg \,;\, \mu_B} \end{array}$$

$\mathrel{=\!\!\prec}$ simplification $\rangle$

$$\begin{array}{lll} \textit{let} & f & = [\![\Gamma \vdash^{p} e : \Box A]\!]_p \,;\, \varepsilon_{\Box A} \\ & g & = [\![\Gamma, x : A^{p} \vdash C\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ \textit{in} & \multicolumn{2}{l}{\langle id_\Gamma , f \,;\, \eta_{\Box A} \rangle \,;\, \tau_{\Gamma,\Box A} \,;\, Tg \,;\, \mu_B} \end{array}$$

$\mathrel{=\!\!\prec}$ strength commutes with unit $\rangle$

$$\begin{array}{lll} \textit{let} & f & = [\![\Gamma \vdash^{p} e : \Box A]\!]_p \,;\, \varepsilon_{\Box A} \\ & g & = [\![\Gamma, x : A^{p} \vdash C\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ \textit{in} & \multicolumn{2}{l}{\langle id_\Gamma , f \rangle \,;\, \eta_{\Gamma \times \Box A} \,;\, Tg \,;\, \mu_B} \end{array}$$

$\mathrel{=\!\!\prec}$ monad laws $\rangle$

$$\begin{array}{lll} \textit{let} & f & = [\![\Gamma \vdash^{p} e : \Box A]\!]_p \,;\, \varepsilon_{\Box A} \\ & g & = [\![\Gamma, x : A^{p} \vdash C\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ \textit{in} & \multicolumn{2}{l}{\langle id_\Gamma , f \rangle \,;\, g \,;\, T\eta_B \,;\, \mu_B} \end{array}$$

$\mathrel{=\!\!\prec}$ monad laws $\rangle$

$$\begin{array}{ll} let & f = [\![\Gamma \vdash^{p} e : \square A]\!]_p \,;\, \varepsilon_{\square A} \\ & g = [\![\Gamma, x : A^{p} \vdash C\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ in & \langle id_\Gamma , f \rangle \,;\, g \end{array}$$

Fixing f, we proceed by cases on C.

◇ $C = [\cdot]$

$$\begin{aligned} & [\![\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e\ \mathsf{in\ box}\,\boxed{x} : \square A]\!] \\ \preceq\!\!\!= \langle & \text{ observation } \rangle \\ & \langle id_\Gamma , f \rangle \,;\, [\![\Gamma, x : A^{p} \vdash \mathsf{box}\,\boxed{x} : \square A]\!] \\ \preceq\!\!\!= \langle & \text{ definition } \rangle \\ & \langle id_\Gamma , f \rangle \,;\, [\![\Gamma, x : A^{p} \vdash^{p} x : A]\!]_p \,;\, \eta_{\square A} \\ \preceq\!\!\!= \langle & \text{ definition } \rangle \\ & \langle id_\Gamma , f \rangle \,;\, \pi_2 \,;\, \eta_{\square A} \\ \preceq\!\!\!= \langle & \text{ applying } \pi_2 \rangle \\ & f \,;\, \eta_{\square A} \\ \preceq\!\!\!= \langle & \text{ definition } \rangle \\ & [\![\Gamma \vdash e : \square A]\!] \end{aligned}$$

◇ $C = e_1\ C_1$

$$\begin{aligned} & [\![\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e\ \mathsf{in}\ e_1\ C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ \preceq\!\!\!= \langle & \text{ observation } \rangle \\ & \langle id_\Gamma , f \rangle \,;\, [\![\Gamma, x : A^{p} \vdash e_1\ C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ \preceq\!\!\!= \langle & \text{ definition } \rangle \\ & \begin{array}{ll} let & h_1 = [\![\Gamma, x : A^{p} \vdash e_1 : C \Rightarrow B]\!] \\ & h_2 = [\![\Gamma, x : A^{p} \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C]\!] \\ in & \langle id_\Gamma , f \rangle \,;\, \langle h_1 , h_2 \rangle \,;\, \beta_{C \to TB, C} \,;\, T\,\mathsf{ev}_{C,TB} \,;\, \mu_B \end{array} \\ \preceq\!\!\!= \langle & \text{ semantic weakening lemma 5.6 } \rangle \\ & \begin{array}{ll} let & h_1 = [\![\Gamma \vdash e_1 : C \Rightarrow B]\!] \\ & h_2 = [\![\Gamma, x : A^{p} \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C]\!] \\ in & \langle id_\Gamma , f \rangle \,;\, \langle \pi_1 \,;\, h_1 , h_2 \rangle \,;\, \beta_{C \to TB, C} \,;\, T\,\mathsf{ev}_{C,TB} \,;\, \mu_B \end{array} \\ \preceq\!\!\!= \langle & \text{ simplification } \rangle \end{aligned}$$

$$\begin{array}{ll} \mathit{let} & h_1 = [\![\Gamma \vdash e_1 : C \Rightarrow B]\!] \\ & h_2 = [\![\Gamma, x : C^p \vdash \mathcal{C}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C]\!] \\ \mathit{in} & \langle\langle id_\Gamma\ , f\rangle \mathbin{;} \pi_1 \mathbin{;} h_1\ , \langle id_\Gamma\ , f\rangle \mathbin{;} h_2\rangle \mathbin{;} \beta_{C\to TB,C} \mathbin{;} T\ \mathsf{ev}_{C,TB} \mathbin{;} \mu_B \end{array}$$

$\eqslantless\langle$ simplification $\rangle$

$$\begin{array}{ll} \mathit{let} & h_1 = [\![\Gamma \vdash e_1 : C \Rightarrow B]\!] \\ & h_2 = [\![\Gamma, x : A^p \vdash \mathcal{C}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C]\!] \\ \mathit{in} & \langle h_1\ , \langle id_\Gamma\ , f\rangle \mathbin{;} h_2\rangle \mathbin{;} \beta_{C\to TB,C} \mathbin{;} T\ \mathsf{ev}_{C,TB} \mathbin{;} \mu_B \end{array}$$

$\eqslantless\langle$ observation $\rangle$

$$\begin{array}{ll} \mathit{let} & h_1 = [\![\Gamma \vdash e_1 : C \Rightarrow B]\!] \\ & h_2 = [\![\Gamma \vdash \mathsf{let\ box}\ \boxed{x} = e\ \mathsf{in}\ \mathcal{C}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C]\!] \\ \mathit{in} & \langle h_1\ , h_2\rangle \mathbin{;} \beta_{C\to TB,C} \mathbin{;} T\ \mathsf{ev}_{C,TB} \mathbin{;} \mu_B \end{array}$$

$\eqslantless\langle$ induction hypothesis $\rangle$

$$\begin{array}{ll} \mathit{let} & h_1 = [\![\Gamma \vdash e_1 : C \Rightarrow B]\!] \\ & h_2 = [\![\Gamma \vdash \mathcal{C}_1\langle\!\langle e \rangle\!\rangle : C]\!] \\ \mathit{in} & \langle h_1\ , h_2\rangle \mathbin{;} \beta_{C\to TB,C} \mathbin{;} T\ \mathsf{ev}_{C,TB} \mathbin{;} \mu_B \end{array}$$

$\eqslantless\langle$ definition $\rangle$

$$[\![\Gamma \vdash e_1\ \mathcal{C}_1\langle\!\langle e \rangle\!\rangle : B]\!]$$

◇ $\mathcal{C} = \mathcal{C}_1\ e_1$

$$[\![\Gamma \vdash \mathsf{let\ box}\ \boxed{x} = e\ \mathsf{in}\ \mathcal{C}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle\ e_1 : B]\!]$$

$\eqslantless\langle$ observation $\rangle$

$$\langle id_\Gamma\ , f\rangle \mathbin{;} [\![\Gamma, x : A^p \vdash \mathcal{C}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle\ e_1 : B]\!]$$

$\eqslantless\langle$ definition $\rangle$

$$\begin{array}{ll} \mathit{let} & h_1 = [\![\Gamma, x : A^p \vdash \mathcal{C}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C \Rightarrow B]\!] \\ & h_2 = [\![\Gamma, x : A^p \vdash e_1 : C]\!] \\ \mathit{in} & \langle id_\Gamma\ , f\rangle \mathbin{;} \langle h_1\ , h_2\rangle \mathbin{;} \beta_{C\to TB,C} \mathbin{;} T\ \mathsf{ev}_{C,TB} \mathbin{;} \mu_B \end{array}$$

$\eqslantless\langle$ semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{ll} \mathit{let} & h_1 = [\![\Gamma, x : A^p \vdash \mathcal{C}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C \Rightarrow B]\!] \\ & h_2 = [\![\Gamma \vdash e_1 : C]\!] \\ \mathit{in} & \langle id_\Gamma\ , f\rangle \mathbin{;} \langle h_1\ , \pi_1 \mathbin{;} h_2\rangle \mathbin{;} \beta_{C\to TB,C} \mathbin{;} T\ \mathsf{ev}_{C,TB} \mathbin{;} \mu_B \end{array}$$

$\eqslantless\langle$ simplification $\rangle$

$$\begin{array}{ll} \mathit{let} & h_1 = [\![\Gamma, x : A^p \vdash \mathcal{C}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C \Rightarrow B]\!] \\ & h_2 = [\![\Gamma \vdash e_1 : C]\!] \\ \mathit{in} & \langle\langle id_\Gamma\ , f\rangle \mathbin{;} h_1\ , \langle id_\Gamma\ , f\rangle \mathbin{;} \pi_1 \mathbin{;} h_2\rangle \mathbin{;} \beta_{C\to TB,C} \mathbin{;} T\ \mathsf{ev}_{C,TB} \mathbin{;} \mu_B \end{array}$$

$\eqsim\langle$ simplification $\rangle$

$$\begin{array}{ll} let & \begin{array}{lcl} h_1 & = & [\![\Gamma, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \Rightarrow B]\!] \\ h_2 & = & [\![\Gamma \vdash e_1 : C]\!] \end{array} \\ in & \langle \langle id_\Gamma \,, f \rangle \,;\, h_1 \,, h_2 \rangle \,;\, \beta_{C \to TB, C} \,;\, T\,\mathsf{ev}_{C,TB} \,;\, \mu_B \end{array}$$

$\eqsim\langle$ observation $\rangle$

$$\begin{array}{ll} let & \begin{array}{lcl} h_1 & = & [\![\Gamma \vdash \mathsf{let\,box}\,\boxed{x} = e \,\mathsf{in}\, C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \Rightarrow B]\!] \\ h_2 & = & [\![\Gamma \vdash e_1 : C]\!] \end{array} \\ in & \langle h_1 \,, h_2 \rangle \,;\, \beta_{C \to TB, C} \,;\, T\,\mathsf{ev}_{C,TB} \,;\, \mu_B \end{array}$$

$\eqsim\langle$ induction hypothesis $\rangle$

$$\begin{array}{ll} let & \begin{array}{lcl} h_1 & = & [\![\Gamma \vdash C_1\langle\!\langle e \rangle\!\rangle : C \Rightarrow B]\!] \\ h_2 & = & [\![\Gamma \vdash e_1 : C]\!] \end{array} \\ in & \langle h_1 \,, h_2 \rangle \,;\, \beta_{C \to TB, C} \,;\, T\,\mathsf{ev}_{C,TB} \,;\, \mu_B \end{array}$$

$\eqsim\langle$ definition $\rangle$

$$[\![\Gamma \vdash C_1\langle\!\langle e \rangle\!\rangle\, e_1 : B]\!]$$

$\diamond\ C = \lambda z : C.\ C_1$

$$[\![\Gamma \vdash \mathsf{let\,box}\,\boxed{x} = e \,\mathsf{in}\, \lambda z : C.\ C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \Rightarrow B]\!]$$

$\eqsim\langle$ observation $\rangle$

$$\langle id_\Gamma \,, f \rangle \,;\, [\![\Gamma, x : A^p \vdash \lambda z : C.\ C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \Rightarrow B]\!]$$

$\eqsim\langle$ definition $\rangle$

$$\begin{array}{ll} let & h \ = \ [\![\Gamma, x : A^p, z : C^i \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ in & \langle id_\Gamma \,, f \rangle \,;\, \mathsf{curry}\,(h) \,;\, \eta_{C \to TB} \end{array}$$

$\eqsim\langle$ semantic substitution theorem 5.11 and semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{ll} let & \begin{array}{lcl} s & = & [\![\Gamma, x : A^p, z : C^i \vdash \theta : \Gamma, z : C^i, x : A^p]\!] \\ h & = & s \,;\, [\![\Gamma, z : C^i, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B]\!] \end{array} \\ in & \langle id_\Gamma \,, f \rangle \,;\, \mathsf{curry}\,(h) \,;\, \eta_{C \to TB} \end{array}$$

$\eqsim\langle$ simplification $\rangle$

$$\begin{array}{ll} let & h \ = \ [\![\Gamma, z : C^i, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ in & \langle id_\Gamma \,, f \rangle \,;\, \mathsf{curry}\,(\langle \pi_1 \,;\, \pi_1 \,, \pi_2 \,, \pi_1 \,;\, \pi_2 \rangle \,;\, h) \,;\, \eta_{C \to TB} \end{array}$$

$\eqsim\langle$ universal property of exponential $\rangle$

$$\begin{array}{ll} let & h \ = \ [\![\Gamma, z : C^i, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ in & \mathsf{curry}\,(\langle id_{\Gamma \times C} \,, \pi_1 \,;\, f \rangle \,;\, h) \,;\, \eta_{C \to TB} \end{array}$$

$\eqsim\langle$ observation $\rangle$

$$\begin{array}{lll} let & h & = \llbracket \Gamma, z : C^i \vdash \text{let box } \boxed{x} = e \text{ in } C_1\langle\!\langle \text{box } \boxed{x} \rangle\!\rangle : B \rrbracket \\ in & \text{curry } (h) \mathbin{;} \eta_{C \to TB} & \end{array}$$

$\eqsim\langle$ induction hypothesis $\rangle$

$$\begin{array}{lll} let & h & = \llbracket \Gamma, z : C^i \vdash C_1\langle\!\langle e \rangle\!\rangle : B \rrbracket \\ in & \text{curry } (h) \mathbin{;} \eta_{C \to TB} & \end{array}$$

$\eqsim\langle$ definition $\rangle$

$$\llbracket \Gamma \vdash \lambda z.\ C_1\langle\!\langle e \rangle\!\rangle : C \Rightarrow B \rrbracket$$

◇ $C = \text{fst } C_1$

$$\llbracket \Gamma \vdash \text{let box } \boxed{x} = e \text{ in fst } C_1\langle\!\langle \text{box } \boxed{x} \rangle\!\rangle : B \rrbracket$$

$\eqsim\langle$ observation $\rangle$

$$\langle id_\Gamma, f \rangle \mathbin{;} \llbracket \Gamma, x : A^p \vdash \text{fst } C_1\langle\!\langle \text{box } \boxed{x} \rangle\!\rangle : B \rrbracket$$

$\eqsim\langle$ definition $\rangle$

$$\langle id_\Gamma, f \rangle \mathbin{;} \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \text{box } \boxed{x} \rangle\!\rangle : B \times C \rrbracket \mathbin{;} T\pi_1$$

$\eqsim\langle$ observation $\rangle$

$$\llbracket \Gamma \vdash \text{let box } \boxed{x} = e \text{ in } C_1\langle\!\langle \text{box } \boxed{x} \rangle\!\rangle : B \times C \rrbracket \mathbin{;} T\pi_1$$

$\eqsim\langle$ induction hypothesis $\rangle$

$$\llbracket \Gamma \vdash C_1\langle\!\langle e \rangle\!\rangle : B \times C \rrbracket \mathbin{;} T\pi_1$$

$\eqsim\langle$ definition $\rangle$

$$\llbracket \Gamma \vdash \text{fst } C_1\langle\!\langle e \rangle\!\rangle : B \rrbracket$$

◇ $C = \text{snd } C_1$

$$\llbracket \Gamma \vdash \text{let box } \boxed{x} = e \text{ in snd } C_1\langle\!\langle \text{box } \boxed{x} \rangle\!\rangle : B \rrbracket$$

$\eqsim\langle$ observation $\rangle$

$$\langle id_\Gamma, f \rangle \mathbin{;} \llbracket \Gamma, x : A^p \vdash \text{snd } C_1\langle\!\langle \text{box } \boxed{x} \rangle\!\rangle : B \rrbracket$$

$\eqsim\langle$ definition $\rangle$

$$\langle id_\Gamma, f \rangle \mathbin{;} \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \text{box } \boxed{x} \rangle\!\rangle : C \times B \rrbracket \mathbin{;} T\pi_2$$

$\eqsim\langle$ observation $\rangle$

$$\llbracket \Gamma \vdash \text{let box } \boxed{x} = e \text{ in } C_1\langle\!\langle \text{box } \boxed{x} \rangle\!\rangle : C \times B \rrbracket \mathbin{;} T\pi_2$$

$\eqsim\langle$ induction hypothesis $\rangle$

$$\llbracket \Gamma \vdash C_1\langle\!\langle e \rangle\!\rangle : C \times B \rrbracket \mathbin{;} T\pi_2$$

$= \langle$ definition $\rangle$

$\llbracket \Gamma \vdash \mathsf{snd}\; C_1\langle\!\langle e \rangle\!\rangle : B \rrbracket$

◇ $C = (e_1\,, C_1)$

$\llbracket \Gamma \vdash \mathsf{let\ box}\, \boxed{x} = e\; \mathsf{in}\; (e_1\,, C_1\langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle) : B \times C \rrbracket$

$= \langle$ observation $\rangle$

$\langle id_\Gamma\,, f \rangle \,;\, \llbracket \Gamma, x : A^p \vdash (e_1\,, C_1\langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle) : B \times C \rrbracket$

$= \langle$ definition $\rangle$

$\langle id_\Gamma\,, f \rangle \,;\, \langle \llbracket \Gamma, x : A^p \vdash e_1 : B \rrbracket \,,\, \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : C \rrbracket \rangle \,;\, \beta_{B,C}$

$= \langle$ semantic weakening lemma 5.6 $\rangle$

$\langle id_\Gamma\,, f \rangle \,;\, \langle \pi_1 \,;\, \llbracket \Gamma \vdash e_1 : B \rrbracket \,,\, \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : C \rrbracket \rangle \,;\, \beta_{B,C}$

$= \langle$ universal property of products $\rangle$

$\langle \langle id_\Gamma\,, f \rangle \,;\, \pi_1 \,;\, \llbracket \Gamma \vdash e_1 : B \rrbracket \,,\, \langle id_\Gamma\,, f \rangle \,;\, \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : C \rrbracket \rangle ; \beta_{B,C}$

$= \langle$ definition of π_1 $\rangle$

$\langle \llbracket \Gamma \vdash e_1 : B \rrbracket \,,\, \langle id_\Gamma\,, f \rangle \,;\, \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : C \rrbracket \rangle ; \beta_{B,C}$

$= \langle$ observation $\rangle$

$\langle \llbracket \Gamma \vdash e_1 : B \rrbracket \,,\, \llbracket \Gamma \vdash \mathsf{let\ box}\, \boxed{x} = e\; \mathsf{in}\; C_1\langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : C \rrbracket \rangle ; \beta_{B,C}$

$= \langle$ induction hypothesis $\rangle$

$\langle \llbracket \Gamma \vdash e_1 : B \rrbracket \,,\, \llbracket \Gamma \vdash C_1\langle\!\langle e \rangle\!\rangle : C \rrbracket \rangle ; \beta_{B,C}$

$= \langle$ definition $\rangle$

$\llbracket \Gamma \vdash (e_1\,, C_1\langle\!\langle e \rangle\!\rangle) : B \times C \rrbracket$

◇ $C = (C_1\,, e_1)$

$\llbracket \Gamma \vdash \mathsf{let\ box}\, \boxed{x} = e\; \mathsf{in}\; (C_1\langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle\,, e_1) : C \times B \rrbracket$

$= \langle$ observation $\rangle$

$\langle id_\Gamma\,, f \rangle \,;\, \llbracket \Gamma, x : A^p \vdash (C_1\langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle\,, e_1) : C \times B \rrbracket$

$= \langle$ definition $\rangle$

$\langle id_\Gamma\,, f \rangle \,;\, \langle \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : C \rrbracket \,,\, \llbracket \Gamma, x : A^p \vdash e_1 : B \rrbracket \rangle \,;\, \beta_{C,B}$

$= \langle$ semantic weakening lemma 5.6 $\rangle$

$\langle id_\Gamma\,, f \rangle \,;\, \langle \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : C \rrbracket \,,\, \pi_1 \,;\, \llbracket \Gamma \vdash e_1 : B \rrbracket \rangle \,;\, \beta_{C,B}$

$= \langle$ universal property of products $\rangle$

$$\langle\langle id_\Gamma , f\rangle ; \llbracket \Gamma, x : A^p \vdash C_1\langle\langle \mathsf{box}\,\boxed{x}\rangle\rangle : C \rrbracket , \langle id_\Gamma , f\rangle ; \pi_1 ; \llbracket \Gamma \vdash e_1 : B \rrbracket\rangle ; \beta_{C,B}$$

$=\langle$ definition of π_1 $\rangle$

$$\langle\langle id_\Gamma , f\rangle ; \llbracket \Gamma, x : A^p \vdash C_1\langle\langle \mathsf{box}\,\boxed{x}\rangle\rangle : C \rrbracket , \llbracket \Gamma \vdash e_1 : B \rrbracket\rangle ; \beta_{C,B}$$

$=\langle$ observation $\rangle$

$$\langle \llbracket \Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e\ \mathsf{in}\ C_1\langle\langle \mathsf{box}\,\boxed{x}\rangle\rangle : C \rrbracket , \llbracket \Gamma \vdash e_1 : B \rrbracket\rangle ; \beta_{C,B}$$

$=\langle$ induction hypothesis $\rangle$

$$\langle \llbracket \Gamma \vdash C_1\langle\langle e\rangle\rangle : C \rrbracket , \llbracket \Gamma \vdash e_1 : B \rrbracket\rangle ; \beta_{C,B}$$

$=\langle$ definition $\rangle$

$$\llbracket \Gamma \vdash (C_1\langle\langle e\rangle\rangle , e_1) : C \times B \rrbracket$$

◇ $C = \mathsf{box}\,\boxed{C_1}$

$$\llbracket \Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e\ \mathsf{in\ box}\,\boxed{C_1\langle\langle \mathsf{box}\,\boxed{x}\rangle\rangle} : \Box B \rrbracket$$

$=\langle$ observation $\rangle$

$$\langle id_\Gamma , f\rangle ; \llbracket \Gamma, x : A^p \vdash \mathsf{box}\,\boxed{C_1\langle\langle \mathsf{box}\,\boxed{x}\rangle\rangle} : \Box B \rrbracket$$

$=\langle$ definition $\rangle$

$$\langle id_\Gamma , f\rangle ; \llbracket \Gamma, x : A^p \vdash^p C_1\langle\langle \mathsf{box}\,\boxed{x}\rangle\rangle : B \rrbracket_p ; \eta_{\Box Y}$$

$=\langle$ observation $\rangle$

$$\llbracket \Gamma \vdash^p \mathsf{let\ box}\,\boxed{x} = e\ \mathsf{in}\ C_1\langle\langle \mathsf{box}\,\boxed{x}\rangle\rangle : B \rrbracket_p ; \eta_{\Box Y}$$

$=\langle$ induction hypothesis $\rangle$

$$\llbracket \Gamma \vdash^p C_1\langle\langle e\rangle\rangle : B \rrbracket_p ; \eta_{\Box Y}$$

$=\langle$ definition $\rangle$

$$\llbracket \Gamma \vdash \mathsf{box}\,\boxed{C_1\langle\langle e\rangle\rangle} : \Box B \rrbracket$$

◇ $C = \mathsf{let\ box}\,\boxed{z} = C_1\ \mathsf{in}\ e_1$

$$\llbracket \Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e\ \mathsf{in}\ (\mathsf{let\ box}\,\boxed{z} = C_1\langle\langle \mathsf{box}\,\boxed{x}\rangle\rangle\ \mathsf{in}\ e_1) : B \rrbracket$$

$=\langle$ observation $\rangle$

$$\langle id_\Gamma , f\rangle ; \llbracket \Gamma, x : A^p \vdash \mathsf{let\ box}\,\boxed{z} = C_1\langle\langle \mathsf{box}\,\boxed{x}\rangle\rangle\ \mathsf{in}\ e_1 : B \rrbracket$$

$=\langle$ definition $\rangle$

$$\begin{array}{l} let \quad g \;=\; \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : \square C \rrbracket \\ \quad h \;=\; \llbracket \Gamma, x : A^p, z : C^p \vdash e_1 : B \rrbracket \\ in \quad \langle id_\Gamma , f \rangle ; \langle id_{\Gamma\times\square A} , g \rangle ; \tau_{\Gamma\times\square A,\square C} ; Th ; \mu_B \end{array}$$

$\mathrel{=\!\!\prec}$ ⟨ semantic substitution theorem 5.11 and semantic weakening lemma 5.6 ⟩

$$\begin{array}{l} let \quad g \;=\; \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : \square C \rrbracket \\ \quad h \;=\; \langle \pi_1 ; \pi_1 , \pi_2 \rangle ; \llbracket \Gamma, z : C^p \vdash e_1 : B \rrbracket \\ in \quad \langle id_\Gamma , f \rangle ; \langle id_{\Gamma\times\square A} , g \rangle ; \tau_{\Gamma\times\square A,\square C} ; Th ; \mu_B \end{array}$$

$\mathrel{=\!\!\prec}$ ⟨ simplification ⟩

$$\begin{array}{l} let \quad g \;=\; \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : \square C \rrbracket \\ \quad h \;=\; \llbracket \Gamma, z : C^p \vdash e_1 : B \rrbracket \\ in \quad \langle \langle id_\Gamma , f \rangle , \langle id_\Gamma , f \rangle ; g \rangle ; \tau_{\Gamma\times\square A,\square C} ; T\langle \pi_1 ; \pi_1 , \pi_2 \rangle ; Th ; \mu_B \end{array}$$

$\mathrel{=\!\!\prec}$ ⟨ simplification ⟩

$$\begin{array}{l} let \quad g \;=\; \llbracket \Gamma, x : A^p \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : \square C \rrbracket \\ \quad h \;=\; \llbracket \Gamma, z : C^p \vdash e_1 : B \rrbracket \\ in \quad \langle id_\Gamma , \langle id_\Gamma , f \rangle ; g \rangle ; \tau_{\Gamma,\square C} ; Th ; \mu_B \end{array}$$

$\mathrel{=\!\!\prec}$ ⟨ observation ⟩

$$\begin{array}{l} let \quad g \;=\; \llbracket \Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e \;\mathsf{in}\; C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : \square C \rrbracket \\ \quad h \;=\; \llbracket \Gamma, z : C^p \vdash e_1 : B \rrbracket \\ in \quad \langle id_\Gamma , g \rangle ; \tau_{\Gamma,\square C} ; Th ; \mu_B \end{array}$$

$\mathrel{=\!\!\prec}$ ⟨ induction hypothesis ⟩

$$\begin{array}{l} let \quad g \;=\; \llbracket \Gamma \vdash C_1\langle\!\langle e \rangle\!\rangle : \square C \rrbracket \\ \quad h \;=\; \llbracket \Gamma, z : C^p \vdash e_1 : B \rrbracket \\ in \quad \langle id_\Gamma , g \rangle ; \tau_{\Gamma,\square C} ; Th ; \mu_B \end{array}$$

$\mathrel{=\!\!\prec}$ ⟨ definition ⟩

$$\llbracket \Gamma \vdash \mathsf{let\ box}\,\boxed{z} = C_1\langle\!\langle e \rangle\!\rangle \;\mathsf{in}\; e_1 : B \rrbracket$$

◇ $C = \mathsf{let\ box}\,\boxed{z} = e_1 \;\mathsf{in}\; C_1$

$$\llbracket \Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e \;\mathsf{in}\; (\mathsf{let\ box}\,\boxed{z} = e_1 \;\mathsf{in}\; C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle) : B \rrbracket$$

$\mathrel{=\!\!\prec}$ ⟨ observation ⟩

$$\langle id_\Gamma , f \rangle ; \llbracket \Gamma, x : A^p \vdash \mathsf{let\ box}\,\boxed{z} = e_1 \;\mathsf{in}\; C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B \rrbracket$$

$\mathrel{=\!\!\prec}$ ⟨ definition ⟩

$$\begin{array}{l} let \quad h_1 \;=\; \llbracket \Gamma, x : A^p \vdash e_1 : \square C \rrbracket \\ \quad h_2 \;=\; \llbracket \Gamma, x : A^p, z : C^p \vdash C_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B \rrbracket \\ in \quad \langle id_\Gamma , f \rangle ; \langle id_{\Gamma\times\square A} , h_1 \rangle ; \tau_{\Gamma\times\square A,\square C} ; Th_2 ; \mu_B \end{array}$$

$\preccurlyeq$ ⟨ semantic weakening lemma 5.6 ⟩

$$\begin{array}{ll} let & h_1 = [\![\Gamma \vdash e_1 : \square C]\!] \\ & h_2 = [\![\Gamma, x : A^p, z : C^p \vdash C_1\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ in & \langle id_\Gamma , f\rangle \,;\, \langle id_{\Gamma\times\square A} , \pi_1 \,;\, h_1\rangle \,;\, \tau_{\Gamma\times\square A,\square C} \,;\, Th_2 \,;\, \mu_B \end{array}$$

$\preccurlyeq$ ⟨ simplification ⟩

$$\begin{array}{ll} let & h_1 = [\![\Gamma \vdash e_1 : \square C]\!] \\ & h_2 = [\![\Gamma, x : A^p, z : C^p \vdash C_1\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ in & \langle\langle id_\Gamma , f\rangle , h_1\rangle \,;\, \tau_{\Gamma\times\square A,\square C} \,;\, Th_2 \,;\, \mu_B \end{array}$$

$\preccurlyeq$ ⟨ semantic substitution theorem 5.11 and semantic weakening lemma 5.6 ⟩

$$\begin{array}{ll} let & h_1 = [\![\Gamma \vdash e_1 : \square C]\!] \\ & h_2 = [\![\Gamma, z : C^p, x : A^p \vdash C_1\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ in & \langle\langle id_\Gamma , f\rangle , h_1\rangle \,;\, \tau_{\Gamma\times\square A,\square C} \,;\, T\langle \pi_1 \,;\, \pi_1 , \pi_2 , \pi_1 \,;\, \pi_2\rangle \,;\, Th_2 \,;\, \mu_B \end{array}$$

$\preccurlyeq$ ⟨ simplification ⟩

$$\begin{array}{ll} let & h_1 = [\![\Gamma \vdash e_1 : \square C]\!] \\ & h_2 = [\![\Gamma, z : C^p, x : A^p \vdash C_1\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ in & \langle id_\Gamma , h_1\rangle \,;\, \tau_{\Gamma,\square C} \,;\, T\langle id_{\Gamma\times\square C} , \pi_1 \,;\, f\rangle \,;\, Th_2 \,;\, \mu_B \end{array}$$

$\preccurlyeq$ ⟨ observation ⟩

$$\begin{array}{ll} let & h_1 = [\![\Gamma \vdash e_1 : \square C]\!] \\ & h_2 = [\![\Gamma, z : C^p \vdash \text{let box}\,\boxed{x} = e \text{ in } C_1\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ in & \langle id_\Gamma , h_1\rangle \,;\, \tau_{\Gamma,\square C} \,;\, Th_2 \,;\, \mu_B \end{array}$$

$\preccurlyeq$ ⟨ induction hypothesis ⟩

$$\begin{array}{ll} let & h_1 = [\![\Gamma \vdash e_1 : \square C]\!] \\ & h_2 = [\![\Gamma, z : C^p \vdash C_1\langle\!\langle e \rangle\!\rangle : B]\!] \\ in & \langle id_\Gamma , h_1\rangle \,;\, \tau_{\Gamma,\square C} \,;\, Th_2 \,;\, \mu_B \end{array}$$

$\preccurlyeq$ ⟨ definition ⟩

$$[\![\Gamma \vdash \text{let box}\,\boxed{z} = e_1 \text{ in } C_1\langle\!\langle e \rangle\!\rangle : B]\!]$$

$$\diamond\ \frac{\Gamma \vdash e : \square A \qquad \Gamma \vdash \mathcal{E}\langle\!\langle e \rangle\!\rangle : B \qquad \Gamma \vdash \text{let box}\,\boxed{x} = e \text{ in } \mathcal{E}\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B}{\Gamma \vdash \mathcal{E}\langle\!\langle e \rangle\!\rangle \approx \text{let box}\,\boxed{x} = e \text{ in } \mathcal{E}\langle\!\langle \text{box}\,\boxed{x} \rangle\!\rangle : B}\ \square\eta\text{-IMPURE}$$

We proceed by cases on $\mathcal{E}$.

$\diamond\ \mathcal{E} = [\cdot]$

$$[\![\Gamma \vdash \text{let box}\,\boxed{x} = e \text{ in box}\,\boxed{x} : \square A]\!]$$

$\preccurlyeq$ ⟨ definition ⟩

$$\langle id_\Gamma , [\![\Gamma \vdash e : \square A]\!]\rangle \,;\, \tau_{\Gamma,\square A} \,;\, T[\![\Gamma, x : A^p \vdash \text{box}\,\boxed{x} : \square A]\!] \,;\, \mu_{\square A}$$

$=\langle$ definition $\rangle$

$$\langle id_\Gamma \mathbin{,} [\![\Gamma \vdash e : \Box A]\!]\rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} T[\![\Gamma, x : A^p \vdash^p x : A]\!]_p \mathbin{;} T\eta_{\Box A} \mathbin{;} \mu_{\Box A}$$

$=\langle$ definition $\rangle$

$$\langle id_\Gamma \mathbin{,} [\![\Gamma \vdash e : \Box A]\!]\rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} T\pi_2 \mathbin{;} T\eta_{\Box A} \mathbin{;} \mu_{\Box A}$$

$=\langle$ monad laws $\rangle$

$$\langle id_\Gamma \mathbin{,} [\![\Gamma \vdash e : \Box A]\!]\rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} T\pi_2 \mathbin{;} id_{T\Box A}$$

$=\langle$ tensorial action of T $\rangle$

$$\langle id_\Gamma \mathbin{,} [\![\Gamma \vdash e : \Box A]\!]\rangle \mathbin{;} \pi_2$$

$=\langle$ applying π_2 $\rangle$

$$[\![\Gamma \vdash e : \Box A]\!]$$

$\diamond\ \mathcal{E} = e_1\ \mathcal{E}_1$

$$[\![\Gamma \vdash \mathsf{let\ box}\ \boxed{x} = e\ \mathsf{in}\ e_1\ \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x}\rangle\!\rangle : B]\!]$$

$=\langle$ definition $\rangle$

$$\begin{array}{lrcl} let & f & = & [\![\Gamma \vdash e : \Box A]\!] \\ & g & = & [\![\Gamma, x : A^p \vdash e_1\ \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x}\rangle\!\rangle : B]\!] \\ in & \multicolumn{3}{l}{\langle id_\Gamma \mathbin{,} f\rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} \mu_B} \end{array}$$

$=\langle$ definition $\rangle$

$$\begin{array}{lrcl} & f & = & [\![\Gamma \vdash e : \Box A]\!] \\ let & g_1 & = & [\![\Gamma, x : A^p \vdash e_1 : C \Rightarrow B]\!] \\ & g_2 & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x}\rangle\!\rangle : C]\!] \\ & g & = & \langle g_1 \mathbin{,} g_2\rangle \mathbin{;} \beta_{C\to TB,C} \mathbin{;} T\,\mathsf{ev}_{C,TY} \mathbin{;} \mu_B \\ in & \multicolumn{3}{l}{\langle id_\Gamma \mathbin{,} f\rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} \mu_B} \end{array}$$

$=\langle$ functoriality of T $\rangle$

$$\begin{array}{lrcl} & f & = & [\![\Gamma \vdash e : \Box A]\!] \\ let & g_1 & = & [\![\Gamma, x : A^p \vdash e_1 : C \Rightarrow B]\!] \\ & g_2 & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x}\rangle\!\rangle : C]\!] \\ in & \multicolumn{3}{l}{\langle id_\Gamma \mathbin{,} f\rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} T\langle g_1 \mathbin{,} g_2\rangle \mathbin{;} T\beta_{C\to TB,C} \mathbin{;} T^2\,\mathsf{ev}_{C,TY} \mathbin{;} T\mu_B \mathbin{;} \mu_B} \end{array}$$

$=\langle$ semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{lrcl} & f & = & [\![\Gamma \vdash e : \Box A]\!] \\ let & g_1 & = & \pi_1 \mathbin{;} [\![\Gamma \vdash e_1 : C \Rightarrow B]\!] \\ & g_2 & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x}\rangle\!\rangle : C]\!] \\ in & \multicolumn{3}{l}{\langle id_\Gamma \mathbin{,} f\rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} T\langle g_1 \mathbin{,} g_2\rangle \mathbin{;} T\beta_{C\to TB,C} \mathbin{;} T^2\,\mathsf{ev}_{C,TY} \mathbin{;} T\mu_B \mathbin{;} \mu_B} \end{array}$$

$=\langle$ simplification $\rangle$

$$\begin{array}{ll} let & \begin{array}{rcl} f & = & [\![\Gamma \vdash e : \Box A]\!] \\ g_1 & = & [\![\Gamma \vdash e_1 : C \Rightarrow B]\!] \\ g_2 & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C]\!] \end{array} \\ in & \langle id_\Gamma , f \rangle ; \tau_{\Gamma,\Box A} ; T\langle \pi_1 ; g_1 , g_2 \rangle ; T\beta_{C \to TB, C} ; T^2\, \mathsf{ev}_{C,TY} ; T\mu_B ; \mu_B \end{array}$$

$\preceq\langle$ simplification $\rangle$

$$\begin{array}{ll} let & \begin{array}{rcl} f & = & [\![\Gamma \vdash e : \Box A]\!] \\ g_1 & = & [\![\Gamma \vdash e_1 : C \Rightarrow B]\!] \\ g_2 & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C]\!] \end{array} \\ in & \langle g_1 , \langle id_\Gamma , f \rangle ; \tau_{\Gamma,\Box A} ; Tg_2 ; \mu_Z \rangle ; \beta_{C \to TB, C} ; T\, \mathsf{ev}_{C,TY} ; \mu_B \end{array}$$

$\preceq\langle$ definition $\rangle$

$$\begin{array}{ll} let & \begin{array}{rcl} g_1 & = & [\![\Gamma \vdash e_1 : C \Rightarrow B]\!] \\ g_2 & = & [\![\Gamma \vdash \mathsf{let\ box}\ \boxed{x} = e\ \mathsf{in}\ \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C]\!] \end{array} \\ in & \langle g_1 , g_2 \rangle ; \beta_{C \to TB, C} ; T\, \mathsf{ev}_{C,TY} ; \mu_B \end{array}$$

$\preceq\langle$ induction hypothesis $\rangle$

$$\begin{array}{ll} let & \begin{array}{rcl} g_1 & = & [\![\Gamma \vdash e_1 : C \Rightarrow B]\!] \\ g_2 & = & [\![\Gamma \vdash \mathcal{E}_1\langle\!\langle e \rangle\!\rangle : C]\!] \end{array} \\ in & \langle g_1 , g_2 \rangle ; \beta_{C \to TB, C} ; T\, \mathsf{ev}_{C,TY} ; \mu_B \end{array}$$

$\preceq\langle$ definition $\rangle$

$$[\![\Gamma \vdash e_1\ \mathcal{E}_1\langle\!\langle e \rangle\!\rangle : B]\!]$$

$\diamond\ \mathcal{E} = \mathcal{E}_1\ v$

$$[\![\Gamma \vdash \mathsf{let\ box}\ \boxed{x} = e\ \mathsf{in}\ \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle\ v : B]\!]$$

$\preceq\langle$ definition $\rangle$

$$\begin{array}{ll} let & \begin{array}{rcl} f & = & [\![\Gamma \vdash e : \Box A]\!] \\ g & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle\ v : B]\!] \end{array} \\ in & \langle id_\Gamma , f \rangle ; \tau_{\Gamma,\Box A} ; Tg ; \mu_B \end{array}$$

$\preceq\langle$ definition $\rangle$

$$\begin{array}{ll} let & \begin{array}{rcl} f & = & [\![\Gamma \vdash e : \Box A]\!] \\ g_1 & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C \Rightarrow B]\!] \\ g_2 & = & [\![\Gamma, x : A^p \vdash v : C]\!] \\ g & = & \langle g_1 , g_2 \rangle ; \beta_{(C \to TB), C} ; T\, \mathsf{ev}_{C,TY} ; \mu_B \end{array} \\ in & \langle id_\Gamma , f \rangle ; \tau_{\Gamma,\Box A} ; Tg ; \mu_B \end{array}$$

$\preceq\langle$ functoriality of T $\rangle$

$$\begin{array}{lrcl} & f & = & [\![\Gamma \vdash e : \square A]\!] \\ \mathit{let} & g_1 & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \Rightarrow B]\!] \\ & g_2 & = & [\![\Gamma, x : A^p \vdash v : C]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma \,, f\rangle \,;\, \tau_{\Gamma,\square A} \,;\, T\langle g_1 \,, g_2\rangle \,;\, T\beta_{C\to TB,C} \,;\, T^2\,\mathsf{ev}_{C,TY} \,;\, T\mu_B \,;\, \mu_B} \end{array}$$

$\preccurlyeq\langle$ semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{lrcl} & f & = & [\![\Gamma \vdash e : \square A]\!] \\ \mathit{let} & g_1 & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \Rightarrow B]\!] \\ & g_2 & = & \pi_1 \,;\, [\![\Gamma \vdash v : C]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma \,, f\rangle \,;\, \tau_{\Gamma,\square A} \,;\, T\langle g_1 \,, g_2\rangle \,;\, T\beta_{C\to TB,C} \,;\, T^2\,\mathsf{ev}_{C,TY} \,;\, T\mu_B \,;\, \mu_B} \end{array}$$

$\preccurlyeq\langle$ simplification $\rangle$

$$\begin{array}{lrcl} & f & = & [\![\Gamma \vdash e : \square A]\!] \\ \mathit{let} & g_1 & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \Rightarrow B]\!] \\ & g_2 & = & [\![\Gamma \vdash v : C]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma \,, f\rangle \,;\, \tau_{\Gamma,\square A} \,;\, T\langle g_1 \,, \pi_1 \,;\, g_2\rangle \,;\, T\beta_{C\to TB,C} \,;\, T^2\,\mathsf{ev}_{C,TY} \,;\, T\mu_B \,;\, \mu_B} \end{array}$$

$\preccurlyeq\langle$ simplification $\rangle$

$$\begin{array}{lrcl} & f & = & [\![\Gamma \vdash e : \square A]\!] \\ \mathit{let} & g_1 & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \Rightarrow B]\!] \\ & g_2 & = & [\![\Gamma \vdash v : C]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle\langle id_\Gamma \,, f\rangle \,;\, \tau_{\Gamma,\square A} \,;\, Tg_1 \,;\, \mu_{C\to TB} \,, g_2\rangle \,;\, \beta_{C\to TB,C} \,;\, T\,\mathsf{ev}_{C,TY} \,;\, \mu_B} \end{array}$$

$\preccurlyeq\langle$ definition $\rangle$

$$\begin{array}{lrcl} \mathit{let} & g_1 & = & [\![\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e\ \mathsf{in}\ \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \Rightarrow B]\!] \\ & g_2 & = & [\![\Gamma \vdash v : C]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle g_1 \,, g_2\rangle \,;\, \beta_{C\to TB,C} \,;\, T\,\mathsf{ev}_{C,TY} \,;\, \mu_B} \end{array}$$

$\preccurlyeq\langle$ induction hypothesis $\rangle$

$$\begin{array}{lrcl} \mathit{let} & g_1 & = & [\![\Gamma \vdash \mathcal{E}_1\langle\!\langle e \rangle\!\rangle : C \Rightarrow B]\!] \\ & g_2 & = & [\![\Gamma \vdash v : C]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle g_1 \,, g_2\rangle \,;\, \beta_{C\to TB,C} \,;\, T\,\mathsf{ev}_{C,TY} \,;\, \mu_B} \end{array}$$

$\preccurlyeq\langle$ definition $\rangle$

$$[\![\Gamma \vdash \mathcal{E}_1\langle\!\langle e \rangle\!\rangle\, v : B]\!]$$

$\diamond\ \mathcal{E} = \mathsf{fst}\ \mathcal{E}_1$

$$[\![\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e\ \mathsf{in\ fst}\ \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B]\!]$$

$\preccurlyeq\langle$ definition $\rangle$

$$\begin{array}{|l|}\hline \begin{array}{lrcl} \mathit{let} & f & = & [\![\Gamma \vdash e : \Box A]\!] \\ & g & = & [\![\Gamma, x : A^p \vdash \mathsf{fst}\ \mathcal{E}_1 \langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma , f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} \mu_B} \end{array} \\ \hline \end{array}$$

$\eqslantless$ definition $\rangle$

$$\begin{array}{|l|}\hline \begin{array}{lrcl} \mathit{let} & f & = & [\![\Gamma \vdash e : \Box A]\!] \\ & g & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1 \langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B \times C]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma , f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} T^2\pi_1 \mathbin{;} \mu_B} \end{array} \\ \hline \end{array}$$

$\eqslantless$ monad laws $\rangle$

$$\begin{array}{|l|}\hline \begin{array}{lrcl} \mathit{let} & f & = & [\![\Gamma \vdash e : \Box A]\!] \\ & g & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1 \langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B \times C]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma , f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} \mu_B \mathbin{;} T\pi_1} \end{array} \\ \hline \end{array}$$

$\eqslantless$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash \mathsf{let\ box}\ \boxed{x} = e\ \mathsf{in}\ \mathcal{E}_1 \langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B \times C]\!] \mathbin{;} T\pi_1}$$

$\eqslantless$ induction hypothesis $\rangle$

$$\boxed{[\![\Gamma \vdash \mathcal{E}_1 \langle\!\langle e \rangle\!\rangle : B \times C]\!] \mathbin{;} T\pi_1}$$

$\eqslantless$ definition $\rangle$

$$\boxed{[\![\Gamma \vdash \mathsf{fst}\ \mathcal{E}_1 \langle\!\langle e \rangle\!\rangle : B]\!]}$$

- $\mathcal{E} = \mathsf{snd}\ \mathcal{E}_1$

$$\boxed{[\![\Gamma \vdash \mathsf{let\ box}\ \boxed{x} = e\ \mathsf{in\ snd}\ \mathcal{E}_1 \langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B]\!]}$$

$\eqslantless$ definition $\rangle$

$$\begin{array}{|l|}\hline \begin{array}{lrcl} \mathit{let} & f & = & [\![\Gamma \vdash e : \Box A]\!] \\ & g & = & [\![\Gamma, x : A^p \vdash \mathsf{snd}\ \mathcal{E}_1 \langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma , f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} \mu_B} \end{array} \\ \hline \end{array}$$

$\eqslantless$ definition $\rangle$

$$\begin{array}{|l|}\hline \begin{array}{lrcl} \mathit{let} & f & = & [\![\Gamma \vdash e : \Box A]\!] \\ & g & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1 \langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C \times B]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma , f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} T^2\pi_2 \mathbin{;} \mu_B} \end{array} \\ \hline \end{array}$$

$\eqslantless$ monad laws $\rangle$

$$\begin{array}{|l|}\hline \begin{array}{lrcl} \mathit{let} & f & = & [\![\Gamma \vdash e : \Box A]\!] \\ & g & = & [\![\Gamma, x : A^p \vdash \mathcal{E}_1 \langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C \times B]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma , f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} \mu_B \mathbin{;} T\pi_2} \end{array} \\ \hline \end{array}$$

$\eqslantless$ definition $\rangle$

$$\llbracket \Gamma \vdash \mathsf{let\ box}\ \boxed{x} = e\ \mathsf{in}\ \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C \times B \rrbracket \mathbin{;} T\pi_2$$

$\eqsim\langle$ induction hypothesis $\rangle$

$$\llbracket \Gamma \vdash \mathcal{E}_1\langle\!\langle e \rangle\!\rangle : C \times B \rrbracket \mathbin{;} T\pi_2$$

$\eqsim\langle$ definition $\rangle$

$$\llbracket \Gamma \vdash \mathsf{snd}\ \mathcal{E}_1\langle\!\langle e \rangle\!\rangle : B \rrbracket$$

$\diamond\ \mathcal{E} = (e_1 , \mathcal{E}_1)$

$$\llbracket \Gamma \vdash \mathsf{let\ box}\ \boxed{x} = e\ \mathsf{in}\ (e_1 , \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle) : B \times C \rrbracket$$

$\eqsim\langle$ definition $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & \llbracket \Gamma \vdash e : \Box A \rrbracket \\ g & = & \llbracket \Gamma, x : A^p \vdash (e_1 , \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle) : B \times C \rrbracket \end{array} \\ \mathit{in} & \langle id_\Gamma , f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} \mu_{B\times C} \end{array}$$

$\eqsim\langle$ definition $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & \llbracket \Gamma \vdash e : \Box A \rrbracket \\ g_1 & = & \llbracket \Gamma, x : A^p \vdash e_1 : B \rrbracket \\ g_2 & = & \llbracket \Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C \rrbracket \\ g & = & \langle g_1 , g_2 \rangle \mathbin{;} \beta_{B,C} \end{array} \\ \mathit{in} & \langle id_\Gamma , f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} \mu_{B\times C} \end{array}$$

$\eqsim\langle$ semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & \llbracket \Gamma \vdash e : \Box A \rrbracket \\ g_1 & = & \pi_1 \mathbin{;} \llbracket \Gamma \vdash e_1 : B \rrbracket \\ g_2 & = & \llbracket \Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C \rrbracket \\ g & = & \langle g_1 , g_2 \rangle \mathbin{;} \beta_{B,C} \end{array} \\ \mathit{in} & \langle id_\Gamma , f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} \mu_{B\times C} \end{array}$$

$\eqsim\langle$ simplification $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & \llbracket \Gamma \vdash e : \Box A \rrbracket \\ g_1 & = & \llbracket \Gamma \vdash e_1 : B \rrbracket \\ g_2 & = & \llbracket \Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C \rrbracket \end{array} \\ \mathit{in} & \langle id_\Gamma , f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} T\langle \pi_1 \mathbin{;} g_1 , g_2 \rangle \mathbin{;} T\beta_{B,C} \mathbin{;} \mu_{B\times C} \end{array}$$

$\eqsim\langle$ simplification $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & \llbracket \Gamma \vdash e : \Box A \rrbracket \\ g_1 & = & \llbracket \Gamma \vdash e_1 : B \rrbracket \\ g_2 & = & \llbracket \Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : C \rrbracket \end{array} \\ \mathit{in} & \langle g_1 , \langle id_\Gamma , f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg_2 \mathbin{;} \mu_C \rangle \mathbin{;} \beta_{B,C} \end{array}$$

$=\langle$ definition $\rangle$

$$\begin{array}{ll} \textit{let} & \begin{array}{lcl} g_1 & = & \llbracket \Gamma \vdash e_1 : B \rrbracket \\ g_2 & = & \llbracket \Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e\ \mathsf{in}\ \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \rrbracket \end{array} \\ \textit{in} & \langle g_1 , g_2 \rangle \,;\, \beta_{B,C} \end{array}$$

$=\langle$ induction hypothesis $\rangle$

$$\begin{array}{ll} \textit{let} & \begin{array}{lcl} g_1 & = & \llbracket \Gamma \vdash e_1 : B \rrbracket \\ g_2 & = & \llbracket \Gamma \vdash \mathcal{E}_1\langle\!\langle e \rangle\!\rangle : C \rrbracket \end{array} \\ \textit{in} & \langle g_1 , g_2 \rangle \,;\, \beta_{B,C} \end{array}$$

$=\langle$ definition $\rangle$

$$\llbracket \Gamma \vdash (e_1 , \mathcal{E}_1\langle\!\langle e \rangle\!\rangle) : B \times C \rrbracket$$

$\diamond\ \mathcal{E} = (\mathcal{E}_1 , v)$

$$\llbracket \Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e\ \mathsf{in}\ (\mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle , v) : C \times B \rrbracket$$

$=\langle$ definition $\rangle$

$$\begin{array}{ll} \textit{let} & \begin{array}{lcl} f & = & \llbracket \Gamma \vdash e : \square A \rrbracket \\ g & = & \llbracket \Gamma, x : A^p \vdash (\mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle , v) : C \times B \rrbracket \end{array} \\ \textit{in} & \langle id_\Gamma , f \rangle \,;\, \tau_{\Gamma,\square A} \,;\, Tg \,;\, \mu_{C \times B} \end{array}$$

$=\langle$ definition $\rangle$

$$\begin{array}{ll} \textit{let} & \begin{array}{lcl} f & = & \llbracket \Gamma \vdash e : \square A \rrbracket \\ g_1 & = & \llbracket \Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \rrbracket \\ g_2 & = & \llbracket \Gamma, x : A^p \vdash v : B \rrbracket \\ g & = & \langle g_1 , g_2 \rangle \,;\, \beta_{C,B} \end{array} \\ \textit{in} & \langle id_\Gamma , f \rangle \,;\, \tau_{\Gamma,\square A} \,;\, Tg \,;\, \mu_{C \times B} \end{array}$$

$=\langle$ semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{ll} \textit{let} & \begin{array}{lcl} f & = & \llbracket \Gamma \vdash e : \square A \rrbracket \\ g_1 & = & \llbracket \Gamma \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \rrbracket \\ g_2 & = & \pi_1 \,;\, \llbracket \Gamma, x : A^p \vdash v : B \rrbracket \\ g & = & \langle g_1 , g_2 \rangle \,;\, \beta_{C,B} \end{array} \\ \textit{in} & \langle id_\Gamma , f \rangle \,;\, \tau_{\Gamma,\square A} \,;\, Tg \,;\, \mu_{C \times B} \end{array}$$

$=\langle$ simplification $\rangle$

$$\begin{array}{ll} \textit{let} & \begin{array}{lcl} f & = & \llbracket \Gamma \vdash e : \square A \rrbracket \\ g_1 & = & \llbracket \Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C \rrbracket \\ g_2 & = & \llbracket \Gamma \vdash v : B \rrbracket \end{array} \\ \textit{in} & \langle id_\Gamma , f \rangle \,;\, \tau_{\Gamma,\square A} \,;\, T\langle g_1 , \pi_1 \,;\, g_2 \rangle \,;\, T\beta_{C,B} \,;\, \mu_{C \times B} \end{array}$$

$\eqslantless$ simplification $\rangle$

$$\begin{array}{ll} & f \;=\; [\![\Gamma \vdash e : \square A]\!] \\ let & g_1 \;=\; [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C]\!] \\ & g_2 \;=\; [\![\Gamma \vdash v : B]\!] \\ in & \langle\langle id_\Gamma , f\rangle ; \tau_{\Gamma,\square A} ; T g_1 ; \mu_C , g_2\rangle ; \beta_{C,B} \end{array}$$

$\eqslantless$ definition $\rangle$

$$\begin{array}{ll} let & g_1 \;=\; [\![\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e \,\mathsf{in}\, \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : C]\!] \\ & g_2 \;=\; [\![\Gamma \vdash v : B]\!] \\ in & \langle g_1 , g_2\rangle ; \beta_{C,B} \end{array}$$

$\eqslantless$ induction hypothesis $\rangle$

$$\begin{array}{ll} let & g_1 \;=\; [\![\Gamma \vdash \mathcal{E}_1\langle\!\langle e \rangle\!\rangle : C]\!] \\ & g_2 \;=\; [\![\Gamma \vdash v : B]\!] \\ in & \langle g_1 , g_2\rangle ; \beta_{C,B} \end{array}$$

$\eqslantless$ definition $\rangle$

$$[\![\Gamma \vdash (\mathcal{E}_1\langle\!\langle e \rangle\!\rangle , v) : C \times B]\!]$$

$\diamond$ $\mathcal{E} = \mathsf{let\ box}\,\boxed{z} = \mathcal{E}_1 \,\mathsf{in}\, e_1$

$$[\![\Gamma \vdash \mathsf{let\ box}\,\boxed{x} = e \,\mathsf{in}\, (\mathsf{let\ box}\,\boxed{z} = \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle \,\mathsf{in}\, e_1) : B]\!]$$

$\eqslantless$ definition $\rangle$

$$\begin{array}{ll} & f \;=\; [\![\Gamma \vdash e : \square A]\!] \\ let & g \;=\; [\![\Gamma, x : A^p \vdash \mathsf{let\ box}\,\boxed{z} = \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle \,\mathsf{in}\, e_1 : B]\!] \\ in & \langle id_\Gamma , f\rangle ; \tau_{\Gamma,\square A} ; T g ; \mu_B \end{array}$$

$\eqslantless$ definition $\rangle$

$$\begin{array}{ll} & f \;=\; [\![\Gamma \vdash e : \square A]\!] \\ & g_1 \;=\; [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : \square C]\!] \\ let & g_2 \;=\; [\![\Gamma, x : A^p, z : C^p \vdash e_1 : B]\!] \\ & g \;=\; \langle id_{\Gamma\times\square A} , g_1\rangle ; \tau_{\Gamma\times\square A,\square C} ; T g_2 ; \mu_B \\ in & \langle id_\Gamma , f\rangle ; \tau_{\Gamma,\square A} ; T g ; \mu_B \end{array}$$

$\eqslantless$ functoriality of T $\rangle$

$$\begin{array}{ll} & f \;=\; [\![\Gamma \vdash e : \square A]\!] \\ let & g_1 \;=\; [\![\Gamma, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : \square C]\!] \\ & g_2 \;=\; [\![\Gamma, x : A^p, z : C^p \vdash e_1 : B]\!] \\ in & \langle id_\Gamma , f\rangle ; \tau_{\Gamma,\square A} ; T\langle id_{\Gamma\times\square A} , g_1\rangle ; T\tau_{\Gamma\times\square A,\square C} ; T^2 g_2 ; T\mu_B ; \mu_B \end{array}$$

$\eqslantless$ semantic substitution theorem 5.11 and semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{lrcl} & f & = & \llbracket \Gamma \vdash e : \Box A \rrbracket \\ \mathit{let} & g_1 & = & \llbracket \Gamma, x : A^p \vdash \mathcal{E}_1 \langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : \Box C \rrbracket \\ & g_2 & = & \langle \pi_1 \mathbin{;} \pi_1 \,, \pi_2 \rangle \mathbin{;} \llbracket \Gamma, z : C^p \vdash e_1 : B \rrbracket \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma \,, f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} T\langle id_{\Gamma\times\Box A} \,, g_1 \rangle \mathbin{;} T\tau_{\Gamma\times\Box A,\Box C} \mathbin{;} T^2 g_2 \mathbin{;} T\mu_B \mathbin{;} \mu_B} \end{array}$$

$\preceq\langle$ simplification $\rangle$

$$\begin{array}{lrcl} & f & = & \llbracket \Gamma \vdash e : \Box A \rrbracket \\ \mathit{let} & g_1 & = & \llbracket \Gamma, x : A^p \vdash \mathcal{E}_1 \langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : \Box C \rrbracket \\ & g_2 & = & \llbracket \Gamma, z : C^p \vdash e_1 : B \rrbracket \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma \,, f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} T\langle id_{\Gamma\times\Box A} \,, g_1 \rangle \mathbin{;} T\tau_{\Gamma\times\Box A,\Box C}} \\ & \multicolumn{3}{l}{\mathbin{;} T^2\langle \pi_1 \mathbin{;} \pi_1 \,, \pi_2 \rangle \mathbin{;} T^2 g_2 \mathbin{;} T\mu_B \mathbin{;} \mu_B} \end{array}$$

$\preceq\langle$ simplification $\rangle$

$$\begin{array}{lrcl} & f & = & \llbracket \Gamma \vdash e : \Box A \rrbracket \\ \mathit{let} & g_1 & = & \llbracket \Gamma, x : A^p \vdash \mathcal{E}_1 \langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : \Box C \rrbracket \\ & g_2 & = & \llbracket \Gamma, z : C^p \vdash e_1 : B \rrbracket \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma \,, \langle id_\Gamma \,, f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg_1 \mathbin{;} \mu_{\Box C} \rangle \mathbin{;} \tau_{\Gamma,\Box C} \mathbin{;} Tg_2 \mathbin{;} \mu_B} \end{array}$$

$\preceq\langle$ definition $\rangle$

$$\begin{array}{lrcl} \mathit{let} & g_1 & = & \llbracket \Gamma \vdash \mathsf{let\ box}\, \boxed{x} = e \mathsf{\ in\ } \mathcal{E}_1 \langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : \Box C \rrbracket \\ & g_2 & = & \llbracket \Gamma, z : C^p \vdash e_1 : B \rrbracket \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma \,, g_1 \rangle \mathbin{;} \tau_{\Gamma,\Box C} \mathbin{;} Tg_2 \mathbin{;} \mu_B} \end{array}$$

$\preceq\langle$ induction hypothesis $\rangle$

$$\begin{array}{lrcl} \mathit{let} & g_1 & = & \llbracket \Gamma \vdash \mathcal{E}_1 \langle\!\langle e \rangle\!\rangle : \Box C \rrbracket \\ & g_2 & = & \llbracket \Gamma, z : C^p \vdash e_1 : B \rrbracket \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma \,, g_1 \rangle \mathbin{;} \tau_{\Gamma,\Box C} \mathbin{;} Tg_2 \mathbin{;} \mu_B} \end{array}$$

$\preceq\langle$ definition $\rangle$

$$\llbracket \Gamma \vdash \mathsf{let\ box}\, \boxed{z} = \mathcal{E}_1 \langle\!\langle e \rangle\!\rangle \mathsf{\ in\ } e_1 : B \rrbracket$$

$\diamond$ $\mathcal{E} = \mathsf{let\ box}\, \boxed{z} = v \mathsf{\ in\ } \mathcal{E}_1$

$$\llbracket \Gamma \vdash \mathsf{let\ box}\, \boxed{x} = e \mathsf{\ in\ } (\mathsf{let\ box}\, \boxed{z} = v \mathsf{\ in\ } \mathcal{E}_1 \langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle) : B \rrbracket$$

$$\begin{array}{lrcl} \mathit{let} & f & = & \llbracket \Gamma \vdash e : \Box A \rrbracket \\ & g & = & \llbracket \Gamma, x : A^p \vdash \mathsf{let\ box}\, \boxed{z} = v \mathsf{\ in\ } \mathcal{E}_1 \langle\!\langle \mathsf{box}\, \boxed{x} \rangle\!\rangle : B \rrbracket \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma \,, f \rangle \mathbin{;} \tau_{\Gamma,\Box A} \mathbin{;} Tg \mathbin{;} \mu_B} \end{array}$$

$\preceq\langle$ definition $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & [\![\Gamma \vdash e : \square A]\!] \\ g_1 & = & [\![\Gamma, x : A^p \vdash v : \square C]\!] \\ g_2 & = & [\![\Gamma, x : A^p, z : C^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B]\!] \\ g & = & \langle id_{\Gamma\times\square A}, g_1\rangle ; \tau_{\Gamma\times\square A,\square C} ; Tg_2 ; \mu_B \end{array} \\ \mathit{in} & \langle id_\Gamma, f\rangle ; \tau_{\Gamma,\square A} ; Tg ; \mu_B \end{array}$$

$=\langle$ functoriality of T $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & [\![\Gamma \vdash e : \square A]\!] \\ g_1 & = & [\![\Gamma, x : A^p \vdash v : \square C]\!] \\ g_2 & = & [\![\Gamma, x : A^p, z : C^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B]\!] \end{array} \\ \mathit{in} & \langle id_\Gamma, f\rangle ; \tau_{\Gamma,\square A} ; T\langle id_{\Gamma\times\square A}, g_1\rangle ; T\tau_{\Gamma\times\square A,\square C} ; T^2 g_2 ; T\mu_B ; \mu_B \end{array}$$

$=\langle$ semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & [\![\Gamma \vdash e : \square A]\!] \\ g_1 & = & \pi_1 ; [\![\Gamma \vdash v : \square C]\!] \\ g_2 & = & [\![\Gamma, x : A^p, z : C^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B]\!] \end{array} \\ \mathit{in} & \langle id_\Gamma, f\rangle ; \tau_{\Gamma,\square A} ; T\langle id_{\Gamma\times\square A}, g_1\rangle ; T\tau_{\Gamma\times\square A,\square C} ; T^2 g_2 ; T\mu_B ; \mu_B \end{array}$$

$=\langle$ simplification $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & [\![\Gamma \vdash e : \square A]\!] \\ g_1 & = & [\![\Gamma \vdash v : \square C]\!] \\ g_2 & = & [\![\Gamma, x : A^p, z : C^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B]\!] \end{array} \\ \mathit{in} & \langle id_\Gamma, f\rangle ; \tau_{\Gamma,\square A} ; T\langle id_{\Gamma\times\square A}, \pi_1 ; g_1\rangle ; T\tau_{\Gamma\times\square A,\square C} ; T^2 g_2 ; T\mu_B ; \mu_B \end{array}$$

$=\langle$ semantic substitution theorem 5.11 and semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & [\![\Gamma \vdash e : \square A]\!] \\ g_1 & = & [\![\Gamma \vdash v : \square C]\!] \\ g_2 & = & \langle \pi_1 ; \pi_1, \pi_2, \pi_1 ; \pi_2\rangle ; [\![\Gamma, z : C^p, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B]\!] \end{array} \\ \mathit{in} & \langle id_\Gamma, f\rangle ; \tau_{\Gamma,\square A} ; T\langle id_{\Gamma\times\square A}, \pi_1 ; g_1\rangle ; T\tau_{\Gamma\times\square A,\square C} ; T^2 g_2 ; T\mu_B ; \mu_B \end{array}$$

$=\langle$ functoriality of T $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & [\![\Gamma \vdash e : \square A]\!] \\ g_1 & = & [\![\Gamma \vdash v : \square C]\!] \\ g_2 & = & [\![\Gamma, z : C^p, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B]\!] \end{array} \\ \mathit{in} & \begin{array}{l} \langle id_\Gamma, f\rangle ; \tau_{\Gamma,\square A} ; T\langle id_{\Gamma\times\square A}, \pi_1 ; g_1\rangle ; T\tau_{\Gamma\times\square A,\square C} \\ ; T^2\langle \pi_1 ; \pi_1, \pi_2, \pi_1 ; \pi_2\rangle ; T^2 g_2 ; T\mu_B ; \mu_B \end{array} \end{array}$$

$=\langle$ semantic weakening lemma 5.6 $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & [\![\Gamma, z : C^p \vdash e : \square A]\!] \\ g_1 & = & [\![\Gamma \vdash v : \square C]\!] \\ g_2 & = & [\![\Gamma, z : C^p, x : A^p \vdash \mathcal{E}_1\langle\!\langle \mathsf{box}\ \boxed{x} \rangle\!\rangle : B]\!] \end{array} \\ \mathit{in} & \begin{array}{l} \langle id_\Gamma, \pi_1 ; f\rangle ; \tau_{\Gamma,\square A} ; T\langle id_{\Gamma\times\square A}, \pi_1 ; g_1\rangle ; T\tau_{\Gamma\times\square A,\square C} \\ ; T^2\langle \pi_1 ; \pi_1, \pi_2, \pi_1 ; \pi_2\rangle ; T^2 g_2 ; T\mu_B ; \mu_B \end{array} \end{array}$$

$= \langle$ simplification $\rangle$

$$\begin{array}{lrcl} \mathit{let} & f & = & [\![\Gamma, z : C^p \vdash e : \square A]\!] \\ & g_1 & = & [\![\Gamma \vdash v : \square C]\!] \\ & g_2 & = & [\![\Gamma, z : C^p, x : A^p \vdash \mathcal{E}_1 \langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma , g_1 \rangle \mathbin{;} \tau_{\Gamma,\square C} \mathbin{;} T\langle id_{\Gamma\times\square C} , f \rangle \mathbin{;} T\tau_{\Gamma\times\square C,\square A} \mathbin{;} T^2 g_2 \mathbin{;} T\mu_B \mathbin{;} \mu_B} \end{array}$$

$= \langle$ definition $\rangle$

$$\begin{array}{lrcl} \mathit{let} & g_1 & = & [\![\Gamma \vdash v : \square C]\!] \\ & g_2 & = & [\![\Gamma, z : C^p \vdash \mathsf{let\ box}\,\boxed{x} = e\ \mathsf{in}\ \mathcal{E}_1 \langle\!\langle \mathsf{box}\,\boxed{x} \rangle\!\rangle : B]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma , g_1 \rangle \mathbin{;} \tau_{\Gamma,\square C} \mathbin{;} T g_2 \mathbin{;} \mu_B} \end{array}$$

$= \langle$ induction hypothesis $\rangle$

$$\begin{array}{lrcl} \mathit{let} & g_1 & = & [\![\Gamma \vdash v : \square C]\!] \\ & g_2 & = & [\![\Gamma, z : C^p \vdash \mathcal{E}_1 \langle\!\langle e \rangle\!\rangle : B]\!] \\ \mathit{in} & \multicolumn{3}{l}{\langle id_\Gamma , g_1 \rangle \mathbin{;} \tau_{\Gamma,\square C} \mathbin{;} T g_2 \mathbin{;} \mu_B} \end{array}$$

$= \langle$ definition $\rangle$

$$[\![\Gamma \vdash \mathsf{let\ box}\,\boxed{z} = v\ \mathsf{in}\ \mathcal{E}_1 \langle\!\langle e \rangle\!\rangle : B]\!]$$

□

E PROOFS FOR SECTION 7 (EMBEDDING)

Lemma E.1. *For any context Γ, we have $\Gamma^p = \Gamma$.*

Proof. We do induction on the context Γ.

(1) $\boxed{\Gamma}$

(2) $\boxed{\Gamma = \cdot}$

(3) $\cdot^p = \cdot^p = \cdot = \cdot$ — by definition

(4) $\boxed{\Gamma = \Delta, x : A}$

(5) $\Delta, x : A^p = (\Delta, x : A^p)^p$ — by definition

(6) $(\Delta, x : A^p)^p = \Delta^p, x : A^p$ — by definition

(7) $\Delta^p, x : A^p = \Delta, x : A^p$ — induction hypothesis

(8) $\Delta, x : A^p = \Delta, x : A^p$

(9) $\Gamma^p = \Gamma$

□

Lemma E.2. $[e'/x]\,e = [e'/x]\,e.$

Proof. We proceed by cases on e.

(1) $[e'/x]\,e$

(2) $e = ()$

(3) $[e'/x]\ ()$ by definition

(4) $()$ by definition

(5) $[e'/x]\ ()$ by definition

(6) $[e'/x]\ ()$ by definition

(7) $e = x$

(8) $[e'/x]\,x$ by definition

(9) e' by definition

(10) $[e'/x]\,x$ by definition

(11) $[e'/x]\ x$ by definition

(12) $e = y, (y \neq x)$

(13) $[e'/y]\,x$ by definition

(14) x by definition

(15) x by definition

(16) $[e'/y]\,x$ by definition

(17) $[e'/y]\ x$ by definition

(18) $e = \lambda y.\, e_1, (y \neq x)$

(19) $[e'/x]\,\lambda y.\, e_1$ by definition

(20) $\lambda y.\ [e'/x]\,e_1$ by definition

(21) $\lambda z.$ let box $\boxed{y}$ $= z$ in $[e'/x]\,e_1$ by definition

(22) $\lambda z.$ let box $\boxed{y}$ $= [e'/x]\,z$ in $[e'/x]\ e_1$ by definition

(23) $[e'/x]\,\lambda z.$ let box $\boxed{x}$ $= z$ in e_1 by definition

(24) $[e'/x]\ \lambda x.\ e_1$ by definition

(25) $e = e_1\,e_2$

(26) $[e'/x]\, e_1\, e_2$ by definition

(27) $[e'/x]\, e_1\, [e'/x]\, e_2$ by definition

(28) $[e'/x]\, e_1\, (\text{box}\ \boxed{[e'/x]\, e_2})$ by definition

(29) $[e'/x]\, e_1\, ([e'/x]\ \text{box}\ \boxed{e_2})$ by definition

(30) $[e'/x]\ e_1\ (\text{box}\ \boxed{e_2})$ by definition

(31) $[e'/x]\ e_1\, e_2$ by definition

(32) $[e'/x]\ e$

$\square$

Lemma E.3. *If* $x : A \in \Gamma$, *then* $x : A^p \in \Gamma$.

Proof. We do induction on $x : A \in \Gamma$.

(1) $\boxed{x : A \in \Gamma}$

(2) $\dfrac{}{x : A \in (\Gamma, x : A)}$ $\in$-id

(3) $x : A^p \in \Gamma, x : A^p$ $\in$-id

(4) $x : A^p \in \Gamma, x : A$ by definition

(5) $\dfrac{x : A \in \Gamma \qquad (x \neq y)}{x : A \in (\Gamma, y : B)}$ $\in$-ex

(6) $x : A \in \Gamma$ inversion

(7) $x : A^p \in \Gamma$ induction hypothesis

(8) $x : A^p \in \Gamma, y : B^p$ $\in$-ex

(9) $x : A^p \in \Gamma, y : B$ by definition

(10) $x : A^p \in \Gamma$

$\square$

Theorem 7.1 Type preservation. *If* $\Gamma \vdash_\lambda e : A$, *then* $\Gamma \vdash e : A$.

Proof. We do induction on $\Gamma \vdash_\lambda e : A$.

(1) $\boxed{\Gamma \vdash_\lambda e : A}$

(2)	$\dfrac{}{\Gamma \vdash_\lambda () : \mathsf{unit}}$	unitI
(3)	$\Gamma \vdash () : \mathsf{unit}$	unitI
(4)	$\Gamma \vdash () : \mathsf{unit}$	by definition
(5)	$\dfrac{x : A \in \Gamma}{\Gamma \vdash_\lambda x : A}$	Var
(6)	$x : A \in \Gamma$	inversion
(7)	$x : A^p \in \Gamma$	lemma E.3
(8)	$\Gamma \vdash x : A$	Var
(9)	$\Gamma \vdash x : A$	by definition
(10)	$\dfrac{\Gamma, x : A \vdash_\lambda e : B}{\Gamma \vdash_\lambda \lambda x : A.\, e : A \Rightarrow B}$	$\Rightarrow$I
(11)	$\Gamma, x : A \vdash_\lambda e : B$	inversion
(12)	$\Gamma, x : A \vdash e : B$	induction hypothesis
(13)	$\Gamma, x : A^p \vdash e : B$	by definition
(14)	$\Gamma, z : \square A \vdash z : \square A$	Var
(15)	$(\Gamma, z : \square A) \supseteq \Gamma$	$\supseteq$-wk
(16)	$(\Gamma, z : \square A, x : A^p) \supseteq (\Gamma, x : A^p)$	$\supseteq$-cong
(17)	$\Gamma, z : \square A, x : A^p \vdash e : B$	lemma 3.1 (16) (13)
(18)	$\Gamma, z : \square A \vdash \mathsf{let\ box}\ \boxed{x} = z\ \mathsf{in}\ e : B$	$\square$E (14) (17)
(19)	$\Gamma \vdash \lambda z : \square A.\ \mathsf{let\ box}\ \boxed{x} = z\ \mathsf{in}\ e : \square A \Rightarrow B$	$\Rightarrow$I
(20)	$\Gamma \vdash \lambda x : A.\, e : A \Rightarrow B$	by definition
(21)	$\dfrac{\Gamma \vdash_\lambda e_1 : A \Rightarrow B \qquad \Gamma \vdash_\lambda e_2 : A}{\Gamma \vdash_\lambda e_1\, e_2 : B}$	$\Rightarrow$E
(22)	$\Gamma \vdash_\lambda e_1 : A \Rightarrow B$	inversion
(23)	$\Gamma \vdash_\lambda e_2 : A$	inversion
(24)	$\Gamma \vdash e_1 : A \Rightarrow B$	induction hypothesis
(25)	$\Gamma \vdash e_1 : \square A \Rightarrow B$	by definition

(26) $\Gamma \vdash e_2 : A$ — induction hypothesis

(27) $\Gamma^{p} \vdash e_2 : A$ — lemma E.1

(28) $\Gamma \vdash^{p} e_2 : A$ — CTX-PURE

(29) $\Gamma \vdash \text{box}\,\boxed{e_2} : \square A$ — $\square$I

(30) $\Gamma \vdash e_1\,(\text{box}\,\boxed{e_2}) : B$ — $\Rightarrow$E (25) (29)

(31) $\Gamma \vdash e_1\, e_2 : B$ — by definition

(32) $\Gamma \vdash e : A$

$\square$

Theorem 7.2 Equality preservation. *If $\Gamma \vdash_\lambda e_1 \approx e_2 : A$, then $\Gamma \vdash e_1 \approx e_2 : A$.*

Proof. We do induction on $\Gamma \vdash_\lambda e_1 \approx e_2 : A$.

(1) $\Gamma \vdash_\lambda e_1 \approx e_2 : A$

(2) $\dfrac{\Gamma, x : A \vdash_\lambda e_1 : B \qquad \Gamma \vdash_\lambda e_2 : A}{\Gamma \vdash_\lambda (\lambda x : A.\, e_1)\, e_2 \approx [e_2/x]e_1 : B}$ — $\Rightarrow\beta$

(3) $\Gamma, x : A \vdash_\lambda e_1 : B$ — inversion

(4) $\Gamma, x : A \vdash e_1 : B$ — theorem 7.1

(5) $\Gamma, x : A^{p} \vdash e_1 : B$ — by definition

(6) $\Gamma \vdash_\lambda e_2 : A$ — inversion

(7) $\Gamma \vdash e_2 : A$ — theorem 7.1

(8) $\Gamma^{p} \vdash e_2 : A$ — lemma E.1

(9) $\Gamma \vdash \text{let box}\,\boxed{x} = \text{box}\,\boxed{e_2}\ \text{in}\ e \approx [e_2/x]e : B$ — $\square\beta$

(10) $\Gamma \vdash \begin{array}{c} (\lambda z : \square A.\ \text{let box}\,\boxed{x} = z\ \text{in}\ e_1)\,(\text{box}\,\boxed{e_2}) \\ \approx \\ \text{let box}\,\boxed{x} = \text{box}\,\boxed{e_2}\ \text{in}\ e_1 \end{array} : B$ — $\Rightarrow\beta$

(11) $\Gamma \vdash \begin{array}{c} (\lambda z : \square A.\ \text{let box}\,\boxed{x} = z\ \text{in}\ e_1)\,(\text{box}\,\boxed{e_2}) \\ \approx \\ [e_1/x]e_2 \end{array} : B$ — TRANS

(12) $\Gamma \vdash (\lambda x : A.\, e_1)\, e_2 \approx [e_2/x]e_1 : B$ by definition

(13) $\dfrac{\Gamma \vdash_\lambda e : A \Rightarrow B}{\Gamma \vdash_\lambda e \approx \lambda x : A.\, e\, x : A \Rightarrow B}$ $\Rightarrow\eta$

(14) $\Gamma \vdash_\lambda e : A \Rightarrow B$ inversion

(15) $\Gamma \vdash e : A \Rightarrow B$ theorem 7.1

(16) $\Gamma \vdash e : \square A \Rightarrow B$ by definition

(17) $\Gamma^{p} \vdash e : \square A \Rightarrow B$ lemma E.1

(18) $\Gamma \vdash^{p} e : \square A \Rightarrow B$ CTX-PURE

(19) $\Gamma \vdash e \approx \lambda z.\, e\, z : \square A \Rightarrow B$ $\Rightarrow\eta$-PURE

(20) $\Gamma, z : \square A \vdash z : \square A$ VAR

(21) $\Gamma, z : \square A \vdash e : \square A \Rightarrow B$ lemma 3.1 (16)

(22) $\Gamma, z : \square A \vdash e\, z : B$ $\Rightarrow$E

(23) $\Gamma, z : \square A, x : A^{p} \vdash x : A$ VAR

(24) $\Gamma, z : \square A, x : A^{p} \vdash \mathsf{box}\, \boxed{x} : \square A$ $\square$I

(25) $\Gamma, z : \square A, x : A^{p} \vdash e\, (\mathsf{box}\, \boxed{x}) : B$ $\Rightarrow$E

(26) $\Gamma, z : \square A \vdash \mathsf{let\ box}\, \boxed{x} = z\ \mathsf{in}\ e\, (\mathsf{box}\, \boxed{x}) : B$ $\square$E

(27) $\Gamma, z : \square A \vdash \boxed{e\, z \approx \mathsf{let\ box}\, \boxed{x} = z\ \mathsf{in}\ e\, (\mathsf{box}\, \boxed{x})} : B$ $\square\eta$-IMPURE on e $\mathcal{E}$

(28) $\Gamma \vdash \boxed{\lambda z.\, e\, z \approx \lambda z.\, \mathsf{let\ box}\, \boxed{x} = z\ \mathsf{in}\ e\, (\mathsf{box}\, \boxed{x})} : \square A \Rightarrow B$ λ-CONG

(29) $\Gamma \vdash \boxed{e \approx \lambda z.\, \mathsf{let\ box}\, \boxed{x} = z\ \mathsf{in}\ e\, (\mathsf{box}\, \boxed{x})} : \square A \Rightarrow B$ TRANS

(30) $\Gamma \vdash \boxed{e \approx \lambda z.\, \mathsf{let\ box}\, \boxed{x} = z\ \mathsf{in}\ e\, (\mathsf{box}\, \boxed{x})} : \square A \Rightarrow B$ by definition

(31) $\Gamma \vdash e \approx \lambda x.\, e\, x : A \Rightarrow B$ by definition

(32) $\dfrac{\Gamma \vdash_\lambda e : A}{\Gamma \vdash_\lambda e \approx e : A}$ REFL

(33)	$\Gamma \vdash_\lambda e : A$	inversion
(34)	$\Gamma \vdash e : A$	theorem 7.1
(35)	$\Gamma \vdash e \approx e : A$	REFL
(36)	$\dfrac{\Gamma \vdash_\lambda e_1 \approx e_2 : A}{\Gamma \vdash_\lambda e_2 \approx e_1 : A}$	SYM
(37)	$\Gamma \vdash_\lambda e_1 \approx e_2 : A$	inversion
(38)	$\Gamma \vdash e_1 \approx e_2 : A$	induction hypothesis
(39)	$\Gamma \vdash e_2 \approx e_1 : A$	SYM
(40)	$\dfrac{\Gamma \vdash_\lambda e_1 \approx e_2 : A \qquad \Gamma \vdash_\lambda e_2 \approx e_3 : A}{\Gamma \vdash_\lambda e_1 \approx e_3 : A}$	TRANS
(41)	$\Gamma \vdash_\lambda e_1 \approx e_2 : A$	inversion
(42)	$\Gamma \vdash_\lambda e_2 \approx e_3 : A$	inversion
(43)	$\Gamma \vdash e_1 \approx e_2 : A$	induction hypothesis
(44)	$\Gamma \vdash e_2 \approx e_3 : A$	induction hypothesis
(45)	$\Gamma \vdash e_1 \approx e_3 : A$	TRANS
(46)	$\dfrac{\Gamma, x : A \vdash_\lambda e_1 \approx e_2 : B}{\Gamma \vdash_\lambda \lambda x : A.\, e_1 \approx \lambda x : A.\, e_2 : A \Rightarrow B}$	λ-CONG
(47)	$\Gamma, x : A \vdash_\lambda e_1 \approx e_2 : B$	inversion
(48)	$\Gamma, x : A \vdash e_1 \approx e_2 : B$	induction hypothesis
(49)	$\Gamma, x : A^{p} \vdash e_1 \approx e_2 : B$	by definition
(50)	$\Gamma, z : \Box A, x : A^{p} \vdash e_1 \approx e_2 : B$	lemma 3.1
(51)	$\Gamma, z : \Box A \vdash z : \Box A$	VAR
(52)	$\Gamma, z : \Box A \vdash z \approx z : \Box A$	REFL
(53)	$\Gamma, z : \Box A \vdash \begin{array}{c}(\text{let box } \boxed{x} = z \text{ in } e_1) \\ \approx \\ (\text{let box } \boxed{x} = z \text{ in } e_2)\end{array} : B$	let box-CONG
(54)	$\Gamma \vdash \begin{array}{c}(\lambda z.\ \text{let box } \boxed{x} = z \text{ in } e_1) \\ \approx \\ (\lambda z.\ \text{let box } \boxed{x} = z \text{ in } e_2)\end{array} : \Box A \Rightarrow B$	λ-CONG

(55)	$\underparen{\Gamma} \vdash \underparen{\lambda x.\, e_1} \approx \underparen{\lambda x.\, e_2} : \underparen{A \Rightarrow B}$	by definition
(56)	$\dfrac{\Gamma \vdash_\lambda e_1 \approx e_2 : A \Rightarrow B \qquad \Gamma \vdash_\lambda e_3 \approx e_4 : A}{\Gamma \vdash_\lambda e_1\, e_3 \approx e_2\, e_4 : B}$	APP-CONG
(57)	$\Gamma \vdash_\lambda e_1 \approx e_2 : A \Rightarrow B$	inversion
(58)	$\underparen{\Gamma} \vdash \underparen{e_1} \approx \underparen{e_2} : \underparen{A \Rightarrow B}$	induction hypothesis
(59)	$\underparen{\Gamma} \vdash \underparen{e_1} \approx \underparen{e_2} : \square \underparen{A} \Rightarrow \underparen{B}$	by definition
(60)	$\underparen{\Gamma} \vdash \underparen{e_3} \approx \underparen{e_4} : \underparen{A}$	induction hypothesis
(61)	$\underparen{\Gamma}^{p} \vdash \underparen{e_3} \approx \underparen{e_4} : \underparen{A}$	lemma E.1
(62)	$\underparen{\Gamma} \vdash \text{box}\, \boxed{\underparen{e_3}} \approx \text{box}\, \boxed{\underparen{e_4}} : \square \underparen{A}$	BOX-CONG
(63)	$\underparen{\Gamma} \vdash \underparen{e_1}\, (\text{box}\, \boxed{\underparen{e_2}}) \approx \underparen{e_3}\, (\text{box}\, \boxed{\underparen{e_4}}) : \underparen{B}$	APP-CONG
(64)	$\underparen{\Gamma} \vdash \underparen{e_1\, e_2} \approx \underparen{e_3\, e_4} : \underparen{B}$	by definition
(65)	$\underparen{\Gamma} \vdash \underparen{e_1} \approx \underparen{e_2} : \underparen{A}$	

□

We can define a reverse translation which forgets the purity annotations, in figure 17.

$$
\begin{array}{lrcl}
\textsc{Types} & \overparen{b} & := & \text{unit} \\
 & \overparen{A \Rightarrow B} & := & \overparen{A} \Rightarrow \overparen{B} \\
 & \overparen{\square A} & := & \overparen{A} \\
 & & & \\
\textsc{Contexts} & \overparen{\cdot} & := & \cdot \\
 & \overparen{\Gamma, x : A^{q}} & := & \overparen{\Gamma}, x : \overparen{A} \\
 & & & \\
\textsc{Terms} & \overparen{()} & := & () \\
 & \overparen{s} & := & () \\
 & \overparen{x} & := & x \\
 & \overparen{\lambda x : A.\, e} & := & \lambda x : \overparen{A}.\, \overparen{e} \\
 & \overparen{e_1\, e_2} & := & \overparen{e_1}\, \overparen{e_2} \\
 & \overparen{\text{box}\, \boxed{e}} & := & \overparen{e} \\
 & \overparen{\text{let box}\, \boxed{x} = e_1 \,\text{in}\, e_2} & := & (\lambda x.\, \overparen{e_2})\, \overparen{e_1} \\
 & \overparen{e_1 \cdot \text{print}(e_2)} & := & ()
\end{array}
$$

Fig. 17. Reverse Translation to STLC

We use the notation $\wideparen{X}$ to denote the *unembedding* of a syntactic object X from our calculus to STLC. We use b to mean base types, i.e., unit, str and cap.

We prove some properties of the unembedding of an embedded term.

Lemma E.4. *For any STLC type A, $\wideparen{\underparen{A}} = A$.*

Proof. We do induction on A.

(1) $\boxed{A}$

(2) $\boxed{\wideparen{\underparen{b}}}$

(3) $\wideparen{b}$ — definition

(4) b — definition

(5) $\boxed{\wideparen{\underparen{A \Rightarrow B}}}$

(6) $\wideparen{\Box\, \underparen{A} \Rightarrow \underparen{B}}$ — definition

(7) $\wideparen{\Box\, \underparen{A}} \Rightarrow \wideparen{\underparen{B}}$ — definition

(8) $\wideparen{\underparen{A}} \Rightarrow \wideparen{\underparen{B}}$ — definition

(9) $A \Rightarrow B$ — induction hypothesis

(10) $\wideparen{\underparen{A}} = A$

□

Lemma E.5. *For any STLC context Γ, $\wideparen{\underparen{\Gamma}} = \Gamma$.*

Proof. We do induction on Γ.

(1) $\boxed{\Gamma}$

(2) $\boxed{\wideparen{\underparen{\cdot}}}$

(3) $\wideparen{\cdot}$ — definition

(4) $\cdot$ — definition

(5) $\boxed{\wideparen{\underparen{\Gamma, x : A}}}$

(6) $\wideparen{\underparen{\Gamma}}, x : \underparen{A}^{p}$ — definition

(7) $\wideparen{\underparen{\Gamma}}, x : \wideparen{\underparen{A}}$ — definition

(8) $\wideparen{\underparen{\Gamma}}, x : A$ — lemma E.4

(9) $\Gamma, x : A$ — induction hypothesis

(10) $\wideparen{\underparen{\Gamma}} = \Gamma$

□

Lemma E.6. *If* $\Gamma \vdash_\lambda e : A$*, then* $\Gamma \vdash_\lambda \underset{\smile}{\overset{\frown}{e}} : A$*.*

Proof. We do induction on $\Gamma \vdash_\lambda e : A$.

(1) $\boxed{\Gamma \vdash_\lambda e : A}$

(2) $\boxed{\dfrac{}{\Gamma \vdash_\lambda () : \mathsf{unit}}}$ unitI

(3) $\Gamma \vdash_\lambda () : \mathsf{unit}$ unitI

(4) $\boxed{\dfrac{x : A \in \Gamma}{\Gamma \vdash_\lambda x : A}}$ Var

(5) $x : A \in \Gamma$ inversion

(6) $\Gamma \vdash_\lambda x : A$ Var

(7) $\boxed{\dfrac{\Gamma, x : A \vdash_\lambda e : B}{\Gamma \vdash_\lambda \lambda x : A.\, e : A \Rightarrow B}}$ $\Rightarrow$I

(8) $\Gamma, x : A \vdash_\lambda e : B$ inversion

(9) $\Gamma \vdash_\lambda \underset{\smile}{\overset{\frown}{\lambda x : A.\, e}} : A \Rightarrow B$

(10) $\boxed{\dfrac{\Gamma \vdash_\lambda e_1 : A \Rightarrow B \qquad \Gamma \vdash_\lambda e_2 : A}{\Gamma \vdash_\lambda e_1\, e_2 : B}}$ $\Rightarrow$E

(11) $\Gamma \vdash_\lambda e_1 : A \Rightarrow B$ inversion

(12) $\Gamma \vdash_\lambda e_2 : A$ inversion

(13) $\Gamma \vdash_\lambda \underset{\smile}{\overset{\frown}{e_1\, e_2}} : B$

(14) $\Gamma \vdash_\lambda \underset{\smile}{\overset{\frown}{e}} : A$

□

We observe that an embedding followed by an unembedding gives a $\beta\eta$-equal term.

Lemma E.7. *If* $\Gamma \vdash_\lambda e : A$*, then* $\Gamma \vdash_\lambda e \approx \underset{\smile}{\overset{\frown}{e}} : A$*.*

Proof. We do induction on $\Gamma \vdash_\lambda e : A$.

(1) $\boxed{\Gamma \vdash_\lambda e : A}$

(2) $\boxed{\dfrac{x : A \in \Gamma}{\Gamma \vdash_\lambda x : A}}$ Var

(3)	$\widehat{x} = \widehat{x} = x$	definition
(4)	$\Gamma \vdash_\lambda \widehat{x} : A$	
(5)	$\Gamma \vdash_\lambda x \approx \widehat{x} : A$	REFL
(6)	$\dfrac{\Gamma, x : A \vdash_\lambda e : B}{\Gamma \vdash_\lambda \lambda x : A.\, e : A \Rightarrow B}$	$\Rightarrow$I
(7)	$\Gamma \vdash_\lambda \lambda x : A.\, e \approx \lambda z : A.\, (\lambda x : A.\, e)\, z : B$	$\Rightarrow\eta$
(8)	$\lambda z : A.\, (\lambda x : A.\, e)\, z = \lambda z : \widehat{A}.\, (\lambda x : A.\, e)\, z$	lemma E.4
(9)	$\lambda z : \widehat{A}.\, (\lambda x : A.\, e)\, z = \lambda z : \square\widehat{A}.\, \widehat{\text{let box } \boxed{x} = z \text{ in } e}$	definition
(10)	$\lambda z : \widehat{\square A}.\, \widehat{\text{let box } \boxed{x} = z \text{ in } e} = \widehat{\lambda z : \square A.\, \text{let box } \boxed{x} = z \text{ in } e}$	definition
(11)	$\widehat{\lambda z : \square A.\, \text{let box } \boxed{x} = z \text{ in } e} = \widehat{\lambda x : A.\, e}$	definition
(12)	$\Gamma \vdash_\lambda \lambda x : A.\, e \approx \widehat{\lambda x : A.\, e} : A \Rightarrow B$	
(13)	$\dfrac{\Gamma \vdash_\lambda e_1 : A \Rightarrow B \qquad \Gamma \vdash_\lambda e_2 : A}{\Gamma \vdash_\lambda e_1\, e_2 : B}$	$\Rightarrow$E
(14)	$\Gamma \vdash_\lambda e_1 : A \Rightarrow B$	inversion
(15)	$\Gamma \vdash_\lambda e_2 : A$	inversion
(16)	$\Gamma \vdash_\lambda e_1 \approx \widehat{e_1} : A \Rightarrow B$	induction hypothesis
(17)	$\Gamma \vdash_\lambda e_2 \approx \widehat{e_2} : A$	induction hypothesis
(18)	$\Gamma \vdash_\lambda e_1\, e_2 \approx \widehat{e_1}\ \widehat{e_2} : B$	APP-CONG
(19)	$\widehat{e_1}\ \widehat{e_2} = \widehat{e_1}\ \widehat{\text{box } \boxed{e_2}} = \widehat{e_1 \text{ box } \boxed{e_2}} = \widehat{e_1\, e_2}$	definition
(20)	$\Gamma \vdash_\lambda e_1\, e_2 \approx \widehat{e_1\, e_2} : B$	
(21)	$\Gamma \vdash_\lambda e \approx \widehat{e} : A$	

□

At this point, we could setup a syntactic logical relation to show a conservative extension result. Instead, we will use an abstract trick.

Note that there is a forgetful functor from $\mathcal{C}$ to Set, which forgets the weight assignments. It is easy to see from our definition of $\mathcal{C}$ in section 4 that this functor preserves the cartesian closed structure, and is hence a cartesian closed functor. Forgetting the extra structure of Set, we could instead choose CCC[1], the free cartesian closed category on one generator 1. We consider the forgetful functor $\mathcal{F}$ from $\mathcal{C}$ to CCC[1], which forgets the capability annotations.

$$\begin{array}{rcl} \mathcal{F}(\mathsf{unit}) & := & 1 \\ \mathcal{F}(\Sigma^*) & := & 1 \\ \mathcal{F}(A \times B) & := & \mathcal{F}(A) \times \mathcal{F}(B) \\ \mathcal{F}(A \Rightarrow B) & := & \mathcal{F}(A) \Rightarrow \mathcal{F}(B) \end{array}$$

We note that it maps the monad and comonad to identity.

$$\begin{array}{rcl} \mathcal{F}(\Box A) & = & \mathcal{F}(A) \\ \mathcal{F}(TA) & = & \mathcal{F}(A) \end{array}$$

We observe that the action of this functor $\mathcal{F}$ on embedded terms gives back the original term.

Lemma E.8. *If* $\Gamma \vdash_\lambda e : A$, *then* $\mathcal{F}(\llbracket \Gamma \vdash e : A \rrbracket) = \llbracket \Gamma \vdash_\lambda e : A \rrbracket$.

Proof. We proceed by induction on $\Gamma \vdash_\lambda e : A$.

◇ $\dfrac{x : A \in \Gamma}{\Gamma \vdash_\lambda x : A}$ Var

$$\boxed{\mathcal{F}(\llbracket \Gamma \vdash x : A \rrbracket)}$$

$=\langle$ definition $\rangle$

$$\boxed{\mathcal{F}(\llbracket x : A \in \Gamma \rrbracket \,;\, \eta_A)}$$

$=\langle$ functoriality of $\mathcal{F}$ $\rangle$

$$\boxed{\mathcal{F}(\llbracket x : A \in \Gamma \rrbracket) \,;\, \mathcal{F}(\eta_A)}$$

$=\langle$ definition $\rangle$

$$\boxed{\llbracket x : A \in \Gamma \rrbracket}$$

$=\langle$ definition $\rangle$

$$\boxed{\llbracket \Gamma \vdash_\lambda e : A \rrbracket}$$

◇ $\dfrac{\Gamma, x : A \vdash_\lambda e : B}{\Gamma \vdash_\lambda \lambda x : A.\, e : A \Rightarrow B}$ $\Rightarrow$I

$$\boxed{\mathcal{F}(\llbracket \Gamma \vdash \lambda x : A.\, e : A \Rightarrow B \rrbracket)}$$

$=\langle$ definition $\rangle$

$$\boxed{\mathcal{F}(\llbracket \Gamma \vdash \lambda z : \Box A.\, \mathsf{let\ box}\, \boxed{x} = z \,\mathsf{in}\, e : \Box A \Rightarrow B \rrbracket)}$$

$=\langle$ definition $\rangle$

$$\boxed{\mathcal{F}(\mathsf{curry}\,(\llbracket \Gamma, z : \Box A^{i} \vdash \mathsf{let\ box}\, \boxed{x} = z \,\mathsf{in}\, e : B \rrbracket) \,;\, \eta_{A \to TB})}$$

$=\langle$ functoriality of $\mathcal{F}$ $\rangle$

$$\boxed{\mathcal{F}(\mathsf{curry}\,(\llbracket \Gamma, z : \Box A^{i} \vdash \mathsf{let\ box}\, \boxed{x} = z \,\mathsf{in}\, e : B \rrbracket)) \,;\, \mathcal{F}(\eta_{A \to TB})}$$

$=\langle$ definition $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & [\![\Gamma, z : \square A^{i} \vdash z : \square A]\!] \\ g & = & [\![\Gamma, z : \square A^{i}, x : A^{p} \vdash e : B]\!] \end{array} \\ \mathit{in} & \mathcal{F}(\mathsf{curry}\,(\langle id_{\Gamma\times\square A}\,, f\rangle \,;\, \tau_{\Gamma\times\square A,\square A} \,;\, Tg \,;\, \mu_B)) \end{array}$$

$=\langle$ simplification $\rangle$

$$\begin{array}{ll} \mathit{let} & g \;=\; [\![\Gamma, z : \square A^{i}, x : A^{p} \vdash e : B]\!] \\ \mathit{in} & \mathcal{F}(\mathsf{curry}\,(\langle id_{\Gamma\times\square A}\,, \pi_2 \,;\, \eta_{\square A}\rangle \,;\, \tau_{\Gamma\times\square A,\square A} \,;\, Tg \,;\, \mu_B)) \end{array}$$

$=\langle$ strength law and monad laws $\rangle$

$$\begin{array}{ll} \mathit{let} & g \;=\; [\![\Gamma, z : \square A^{i}, x : A^{p} \vdash e : B]\!] \\ \mathit{in} & \mathcal{F}(\mathsf{curry}\,(\langle id_{\Gamma\times\square A}\,, \pi_2\rangle \,;\, g)) \end{array}$$

$=\langle$ $\mathcal{F}$ preserves exponentials $\rangle$

$$\mathsf{curry}\,(\mathcal{F}([\![\Gamma, x : A^{p} \vdash e : B]\!]))$$

$=\langle$ definition $\rangle$

$$\mathsf{curry}\,(\mathcal{F}([\![\Gamma, x : A \vdash e : B]\!]))$$

$=\langle$ induction hypothesis $\rangle$

$$\mathsf{curry}\,([\![\Gamma, x : A \vdash_{\lambda} e : B]\!])$$

$=\langle$ definition $\rangle$

$$[\![\Gamma \vdash_{\lambda} \lambda x : A.\, e : A \Rightarrow B]\!]$$

$$\diamond \quad \frac{\Gamma \vdash_{\lambda} e_1 : A \Rightarrow B \qquad \Gamma \vdash_{\lambda} e_2 : A}{\Gamma \vdash_{\lambda} e_1\, e_2 : B} \Rightarrow\!\mathrm{E}$$

$$\mathcal{F}([\![\Gamma \vdash e_1\, e_2 : B]\!])$$

$=\langle$ definition $\rangle$

$$\mathcal{F}([\![\Gamma \vdash e_1\; \mathsf{box}\,\boxed{e_2} : B]\!])$$

$=\langle$ definition $\rangle$

$$\begin{array}{ll} \mathit{let} & \begin{array}{lcl} f & = & [\![\Gamma \vdash e_1 : \square A \Rightarrow B]\!] \\ g & = & [\![\Gamma \vdash \mathsf{box}\,\boxed{e_2} : \square A]\!] \end{array} \\ \mathit{in} & \mathcal{F}(\langle f\,, g\rangle \,;\, \beta_{\square A\to TB,\square A} \,;\, T\,\mathsf{ev}_{\square A,TB} \,;\, \mu_B) \end{array}$$

$=\langle$ functoriality of $\mathcal{F}$ $\rangle$

$$\text{let } \begin{array}{lcl} f & = & [\![\Gamma \vdash e_1 : \Box A \Rightarrow B]\!] \\ g & = & [\![\Gamma \vdash \mathsf{box}\ \boxed{e_2} : \Box A]\!] \end{array}$$
$$\text{in } \mathcal{F}(\langle f , g \rangle) \mathbin{;} \mathcal{F}(\beta_{\Box A \to TB, \Box A}) \mathbin{;} \mathcal{F}(T\,\mathsf{ev}_{\Box A, TB}) \mathbin{;} \mathcal{F}(\mu_B)$$

$\doteqdot\langle$ action of $\mathcal{F}$ $\rangle$

$$\text{let } \begin{array}{lcl} f & = & \mathcal{F}([\![\Gamma \vdash e_1 : A \Rightarrow B]\!]) \\ g & = & \mathcal{F}([\![\Gamma \vdash e_2 : A]\!]) \end{array}$$
$$\text{in } \langle f , g \rangle \mathbin{;} \mathsf{ev}_{A,B}$$

$\doteqdot\langle$ induction hypothesis $\rangle$

$$\text{let } \begin{array}{lcl} f & = & [\![\Gamma \vdash_\lambda e_1 : A \Rightarrow B]\!] \\ g & = & [\![\Gamma \vdash_\lambda e_2 : A]\!] \end{array}$$
$$\text{in } \langle f , g \rangle \mathbin{;} \mathsf{ev}_{A,B}$$

$\doteqdot\langle$ definition $\rangle$

$$[\![\Gamma \vdash_\lambda e_1\, e_2 : B]\!]$$

□

Theorem 7.3 Conservative Extension. *If* $\Gamma \vdash_\lambda e_1 : A$, $\Gamma \vdash_\lambda e_2 : A$, *and* $\Gamma \vdash e_1 \approx e_2 : A$, *then* $\Gamma \vdash_\lambda e_1 \approx e_2 : A$.

Proof.

(1)	$\Gamma \vdash_\lambda e_1 : A, \Gamma \vdash_\lambda e_2 : A$	
(2)	$\Gamma \vdash e_1 \approx e_2 : A$	
(3)	$[\![\Gamma \vdash e_1 : A]\!] = [\![\Gamma \vdash e_2 : A]\!]$	soundness of $\approx$ theorem 6.1
(4)	$\mathcal{F}([\![\Gamma \vdash e_1 : A]\!]) = \mathcal{F}([\![\Gamma \vdash e_2 : A]\!])$	congruence
(5)	$[\![\Gamma \vdash_\lambda e_1 : A]\!] = [\![\Gamma \vdash_\lambda e_2 : A]\!]$	lemma E.8
(6)	$\Gamma \vdash_\lambda e_1 \approx e_2 : A$	completeness of STLC

□