

# Enriched Category Basics

Eugene W. Stark

Department of Computer Science  
Stony Brook University  
Stony Brook, New York 11794 USA

March 17, 2025

## **Abstract**

The notion of an enriched category generalizes the concept of category by replacing the hom-sets of an ordinary category by objects of an arbitrary monoidal category. In this article we give a formal definition of enriched categories and we give formal proofs of a relatively narrow selection of facts about them. One of the main results is a proof that a closed monoidal category can be regarded as a category “enriched in itself”. The other main result is a proof of a version of the Yoneda Lemma for enriched categories.

# Contents

<b>Contents</b>	<b>2</b>
<b>Introduction</b>	<b>2</b>
<b>1 Closed Monoidal Categories</b>	<b>5</b>
1.1 Definition and Basic Facts . . . . .	5
1.1.1 An ECMC is a CMC . . . . .	6
1.1.2 A CMC Extends to an ECMC . . . . .	9
1.2 Internal Hom Functors . . . . .	11
1.2.1 Exponentiation by Unity . . . . .	16
1.2.2 Internal Currying . . . . .	18
1.2.3 Yoneda Embedding . . . . .	25
1.3 Enriched Structure . . . . .	30
1.4 Cartesian Closed Monoidal Categories . . . . .	40
<b>2 Enriched Categories</b>	<b>48</b>
2.1 Basic Definitions . . . . .	49
2.1.1 Self-Enrichment . . . . .	51
2.2 Underlying Category, Functor, and Natural Transformation . . . . .	54
2.2.1 Underlying Category . . . . .	54
2.2.2 Underlying Functor . . . . .	58
2.2.3 Underlying Natural Transformation . . . . .	61
2.2.4 Self-Enriched Case . . . . .	65
2.3 Opposite of an Enriched Category . . . . .	68
2.3.1 Relation between $(-^{op})_0$ and $(-_0)^{op}$ . . . . .	76
2.4 Enriched Hom Functors . . . . .	80
2.4.1 Covariant Case . . . . .	80
2.4.2 Contravariant Case . . . . .	86
2.5 Enriched Yoneda Lemma . . . . .	93
2.5.1 Preliminaries . . . . .	93
2.5.2 Covariant Case . . . . .	102
2.5.3 Contravariant Case . . . . .	123
<b>Bibliography</b>	<b>124</b>

# Introduction

The notion of an enriched category [1] generalizes the concept of category by replacing the hom-sets of an ordinary category by objects of an arbitrary monoidal category  $\mathcal{V}$ . The choice, for each object  $a$ , of a distinguished element  $id\ a : a \rightarrow a$  as an identity, is replaced by an arrow  $Id\ a : \mathcal{I} \rightarrow Hom\ a\ a$  of  $\mathcal{V}$ . The composition operation is similarly replaced by a family of arrows  $Comp\ a\ b\ c : Hom\ B\ C \otimes Hom\ A\ B \rightarrow Hom\ A\ C$  of  $\mathcal{V}$ . The identity and composition are required to satisfy unit and associativity laws which are expressed as commutative diagrams in  $\mathcal{V}$ . Of particular interest is the case in which  $\mathcal{V}$  is symmetric monoidal and closed; in that case, as Kelly states ([1], Section 1.6): “The structure of  $\mathcal{V}$ -**CAT** then becomes rich enough to permit of Yoneda-lemma arguments formally identical with those in **CAT**.”

The goal of this article is to formalize the basic definition of enriched category and some related notions, and to prove a relatively narrow selection of facts about these definitions. For reference and inspiration, we follow the early sections of the book by Kelly [1]; however a comprehensive formalization of the material in that book is explicitly not our objective here. Rather, beyond the basic definitions we are primarily interested in the following two results: (1) that a closed monoidal category can be regarded as a category “enriched in itself”; and (2) the Yoneda Lemma for enriched categories (specifically, the weak form considered in Section 1.9 of [1]). We needed the basic definitions and result (1) for use in a separate article [4]. Although this material could have been included as part of that other article, as it is general material that does not depend on the specific application considered there, it seemed best to present it as a stand-alone development that would be more readily accessible for use by others. As far as result (2) is concerned, we originally formalized and proved it as part of our exploration leading up to [4]. Ultimately, we did not find result (2) to be necessary for the satisfactory development of that work, but as it is a result of general interest whose formalization did involve some struggle to achieve, it seems worthwhile to include it here.

This article is organized as follows: In Chapter 1 we give formal definitions for the notions “closed monoidal category” and “cartesian closed monoidal category” and prove some facts about them. This builds on the

formal development of the theory of monoidal categories in our previous article [3]. The main goals of this section are to prove some general facts about exponentials that are used in [4], and to do most of the preliminary work (the parts that do not specifically depend on the definition of enriched category) involved in showing that a closed monoidal category is “enriched in itself”. In Chapter 2 we give definitions for “enriched category” and the related notions “enriched functor,” “enriched natural transformation,” and “underlying category,” and we complete the formal statement and proof of “self-enrichment.” We then continue with the definition of the opposite of an enriched category, give definitions for the notions of covariant and contravariant enriched hom functors, and prove corresponding covariant and contravariant versions of the Yoneda Lemma.

# Chapter 1

## Closed Monoidal Categories

A *closed monoidal category* is a monoidal category such that for every object  $b$ , the functor  $- \otimes b$  is a left adjoint functor. A right adjoint to this functor takes each object  $c$  to the *exponential*  $\text{exp } b \ c$ . The adjunction yields a natural bijection between  $\text{hom } (- \otimes b) \ c$  and  $\text{hom } - \ (\text{exp } b \ c)$ . In enriched category theory, the notion of “hom-set” from classical category theory is generalized to that of “hom-object” in a monoidal category. When the monoidal category in question is closed, much of the theory of set-based categories can be reproduced in the more general enriched setting. The main purpose of this section is to prepare the way for such a development; in particular we do the main work required to show that a closed monoidal category is “enriched in itself.”

```
theory ClosedMonoidalCategory
imports MonoidalCategory.CartesianMonoidalCategory
begin
```

### 1.1 Definition and Basic Facts

As is pointed out in [2], unless symmetry is assumed as part of the definition, there are in fact two notions of closed monoidal category: *left*-closed monoidal category and *right*-closed monoidal category. Here we define versions with and without symmetry, so that we can identify the places where symmetry is actually required.

```
locale closed-monoidal-category =
  monoidal-category +
assumes left-adjoint-tensor:  $\bigwedge b. \text{ide } b \implies \text{left-adjoint-functor } C \ C \ (\lambda x. x \otimes b)$ 
```

```
locale closed-symmetric-monoidal-category =
  closed-monoidal-category +
  symmetric-monoidal-category
```

Similarly to what we have done in previous work, besides the definition of *closed-monoidal-category*, which adds an assumed property to *monoidal-category*

but not any additional structure, we find it convenient also to define *elementary-closed-monoidal-category*, which assumes particular exponential structure to have been chosen, and uses this given structure to express the properties of a closed monoidal category in a more elementary way.

```

locale elementary-closed-monoidal-category =
  monoidal-category +
fixes exp :: 'a ⇒ 'a ⇒ 'a
and eval :: 'a ⇒ 'a ⇒ 'a
and Curry :: 'a ⇒ 'a ⇒ 'a ⇒ 'a ⇒ 'a
assumes eval-in-hom-ax: [ ide b; ide c ] ⇒ «eval b c : exp b c ⊗ b → c»
and ide-exp [intro, simp]: [ ide b; ide c ] ⇒ ide (exp b c)
and Curry-in-hom-ax: [ ide a; ide b; ide c; «g : a ⊗ b → c» ]
  ⇒ «Curry a b c g : a → exp b c»
and Uncurry-Curry: [ ide a; ide b; ide c; «g : a ⊗ b → c» ]
  ⇒ eval b c · (Curry a b c g ⊗ b) = g
and Curry-Uncurry: [ ide a; ide b; ide c; «h : a → exp b c» ]
  ⇒ Curry a b c (eval b c · (h ⊗ b)) = h

```

```

locale elementary-closed-symmetric-monoidal-category =
  symmetric-monoidal-category +
  elementary-closed-monoidal-category
begin

```

```

  sublocale elementary-symmetric-monoidal-category
    C tensor I lunit runit assoc sym
  using induces-elementary-symmetric-monoidal-categoryCMC by blast

```

```

end

```

We now show that, except for the fact that a particular choice of structure has been made, closed monoidal categories and elementary closed monoidal categories amount to the same thing.

### 1.1.1 An ECMC is a CMC

```

context elementary-closed-monoidal-category
begin

```

```

  notation Curry (⟨Curry[-, -, -]⟩)

```

```

  abbreviation Uncurry (⟨Uncurry[-, -]⟩)
  where Uncurry[b, c] f ≡ eval b c · (f ⊗ b)

```

```

  lemma Curry-in-hom [intro]:
  assumes ide a and ide b and «g : a ⊗ b → c» and y = exp b c
  shows «Curry[a, b, c] g : a → y»
  using assms Curry-in-hom-ax [of a b c g] by fastforce

```

**lemma** *Curry-simps* [*simp*]:  
**assumes** *ide a* **and** *ide b* **and** «*g* :  $a \otimes b \rightarrow c$ »  
**shows** *arr* (*Curry*[*a*, *b*, *c*] *g*)  
**and** *dom* (*Curry*[*a*, *b*, *c*] *g*) = *a*  
**and** *cod* (*Curry*[*a*, *b*, *c*] *g*) = *exp b c*  
**using** *assms Curry-in-hom* **by** *blast+*

**lemma** *eval-in-hom<sub>ECMC</sub>* [*intro*]:  
**assumes** *ide b* **and** *ide c* **and**  $x = \text{exp } b \ c \ \otimes \ b$   
**shows** «*eval b c* :  $x \rightarrow c$ »  
**using** *assms eval-in-hom-ax* **by** *blast*

**lemma** *eval-simps* [*simp*]:  
**assumes** *ide b* **and** *ide c*  
**shows** *arr* (*eval b c*) **and** *dom* (*eval b c*) =  $\text{exp } b \ c \ \otimes \ b$  **and** *cod* (*eval b c*) = *c*  
**using** *assms eval-in-hom<sub>ECMC</sub>* **by** *blast+*

**lemma** *Uncurry-in-hom* [*intro*]:  
**assumes** *ide b* **and** *ide c* **and** «*f* :  $a \rightarrow \text{exp } b \ c$ » **and**  $x = a \ \otimes \ b$   
**shows** «*Uncurry*[*b*, *c*] *f* :  $x \rightarrow c$ »  
**using** *assms* **by** *auto*

**lemma** *Uncurry-simps* [*simp*]:  
**assumes** *ide b* **and** *ide c* **and** «*f* :  $a \rightarrow \text{exp } b \ c$ »  
**shows** *arr* (*Uncurry*[*b*, *c*] *f*)  
**and** *dom* (*Uncurry*[*b*, *c*] *f*) =  $a \ \otimes \ b$   
**and** *cod* (*Uncurry*[*b*, *c*] *f*) = *c*  
**using** *assms Uncurry-in-hom* **by** *blast+*

**lemma** *Uncurry-exp*:  
**assumes** *ide a* **and** *ide b*  
**shows** *Uncurry*[*a*, *b*] (*exp a b*) = *eval a b*  
**using** *assms*  
**by** (*metis comp-arr-dom eval-in-hom<sub>ECMC</sub> in-homE*)

**lemma** *comp-Curry-arr*:  
**assumes** *ide b* **and** «*f* :  $x \rightarrow a$ » **and** «*g* :  $a \ \otimes \ b \rightarrow c$ »  
**shows** *Curry*[*a*, *b*, *c*] *g* · *f* = *Curry*[*x*, *b*, *c*] (*g* · (*f* · *b*))  
**proof** –  
**have** *a*: *ide a* **and** *c*: *ide c* **and** *x*: *ide x*  
**using** *assms(2–3)* **by** *auto*  
**have** *Curry*[*a*, *b*, *c*] *g* · *f* =  
*Curry*[*x*, *b*, *c*] (*Uncurry*[*b*, *c*] (*Curry*[*a*, *b*, *c*] *g* · *f*))  
**using** *assms(1–3)* *a c x Curry-Uncurry comp-in-homI Curry-in-hom*  
**by** *presburger*  
**also have** ... = *Curry*[*x*, *b*, *c*] (*eval b c* · (*Curry*[*a*, *b*, *c*] *g* · *b*) · (*f* · *b*))  
**using** *assms a c interchange*  
**by** (*metis comp-ide-self Curry-in-hom ideD(1) seqI'*)  
**also have** ... = *Curry*[*x*, *b*, *c*] (*Uncurry*[*b*, *c*] (*Curry*[*a*, *b*, *c*] *g*) · (*f* · *b*))

**using** *comp-assoc* **by** *simp*  
**also have**  $\dots = \text{Curry}[x, b, c] (g \cdot (f \otimes b))$   
**using**  $a\ c\ \text{assms}(1,3)$  *Uncurry-Curry* **by** *simp*  
**finally show** *?thesis* **by** *blast*  
**qed**

**lemma** *terminal-arrow-from-functor-eval*:

**assumes** *ide b* **and** *ide c*

**shows** *terminal-arrow-from-functor*  $C\ C\ (\lambda x. T(x, b))\ (\text{exp } b\ c)\ c\ (\text{eval } b\ c)$

**proof** –

**interpret** *functor*  $C\ C\ \langle \lambda x. T(x, b) \rangle$

**using** *assms(1)* *interchange* *T.extensionality*

**by** *unfold-locales auto*

**interpret** *arrow-from-functor*  $C\ C\ \langle \lambda x. T(x, b) \rangle\ \langle \text{exp } b\ c \rangle\ c\ \langle \text{eval } b\ c \rangle$

**using** *assms eval-in-hom<sub>ECMC</sub>*

**by** *unfold-locales auto*

**show** *?thesis*

**proof**

**show**  $\bigwedge a\ f.\ \text{arrow-from-functor } C\ C\ (\lambda x. T(x, b))\ a\ c\ f \implies$

$\exists! g.\ \text{arrow-from-functor.is-coext } C\ C$

$(\lambda x. T(x, b))\ (\text{exp } b\ c)\ (\text{eval } b\ c)\ a\ f\ g$

**proof** –

**fix**  $a\ f$

**assume**  $f:\ \text{arrow-from-functor } C\ C\ (\lambda x. T(x, b))\ a\ c\ f$

**interpret**  $f:\ \text{arrow-from-functor } C\ C\ \langle \lambda x. T(x, b) \rangle\ a\ c\ f$

**using**  $f$  **by** *simp*

**show**  $\exists! g.\ \text{is-coext } a\ f\ g$

**proof**

**have**  $a:\ \text{ide } a$

**using**  $f.\text{arrow}$  **by** *simp*

**show** *is-coext*  $a\ f\ (\text{Curry}[a, b, c] f)$

**unfolding** *is-coext-def*

**using** *assms a Curry-in-hom Uncurry-Curry f.arrow* **by** *force*

**show**  $\bigwedge g.\ \text{is-coext } a\ f\ g \implies g = \text{Curry}[a, b, c] f$

**unfolding** *is-coext-def*

**using** *assms a Curry-Uncurry f.arrow arrI* **by** *force*

**qed**

**qed**

**qed**

**qed**

**lemma** *is-closed-monoidal-category*:

**shows** *closed-monoidal-category*  $C\ T\ \alpha\ \iota$

**using** *T.extensionality* *interchange terminal-arrow-from-functor-eval*

**apply** *unfold-locales*

**apply** *auto[5]*

**by** *metis*

**lemma** *retraction-eval-ide-self*:



**assumes** *ide a*  
**shows** *retraction (eval a a)*  
 by (*metis Uncurry-Curry assms comp-lunit-lunit'(1) ide-unity comp-assoc lunit-in-hom retractionI*)

**end**

**context** *elementary-closed-symmetric-monoidal-category*  
**begin**

**lemma** *is-closed-symmetric-monoidal-category:*  
**shows** *closed-symmetric-monoidal-category C T  $\alpha$   $\iota$   $\sigma$*   
 by (*simp add: closed-symmetric-monoidal-category.intro is-closed-monoidal-category symmetric-monoidal-category-axioms*)

**end**

### 1.1.2 A CMC Extends to an ECMC

**context** *closed-monoidal-category*  
**begin**

**lemma** *has-exponentials:*  
**assumes** *ide b and ide c*  
**shows**  $\exists x e. \text{ide } x \wedge \langle e : x \otimes b \rightarrow c \rangle \wedge$   
 $(\forall a g. \text{ide } a \wedge$   
 $\langle g : a \otimes b \rightarrow c \rangle \longrightarrow (\exists !f. \langle f : a \rightarrow x \rangle \wedge g = e \cdot (f \otimes b)))$

**proof** –

**interpret** *F: left-adjoint-functor C C  $\langle \lambda x. x \otimes b \rangle$*   
**using** *assms(1) left-adjoint-tensor by simp*  
**obtain** *x e where e: terminal-arrow-from-functor C C  $(\lambda x. x \otimes b)$  x c e*  
**using** *assms F.ex-terminal-arrow [of c] by auto*  
**interpret** *e: terminal-arrow-from-functor C C  $\langle \lambda x. x \otimes b \rangle$  x c e*  
**using** *e by simp*  
**have**  $\bigwedge a g. \llbracket \text{ide } a; \langle g : a \otimes b \rightarrow c \rangle \rrbracket$   
 $\implies \exists !f. \langle f : a \rightarrow x \rangle \wedge g = e \cdot (f \otimes b)$   
**using** *e.is-terminal category-axioms F.functor-axioms*  
**unfolding** *e.is-coext-def arrow-from-functor-def*  
*arrow-from-functor-axioms-def*  
**by** *simp*  
**thus** *?thesis*  
**using** *e.arrow by metis*

**qed**

**definition** *some-exp  $(\langle \text{exp}^? \rangle)$*   
**where** *exp<sup>?</sup> b c  $\equiv$  SOME x. ide x  $\wedge$*   
 $(\exists e. \langle e : x \otimes b \rightarrow c \rangle \wedge$

$$\begin{aligned}
& (\forall a g. \text{ide } a \wedge \langle\langle g : a \otimes b \rightarrow c \rangle\rangle \\
& \quad \rightarrow (\exists ! f. \langle\langle f : a \rightarrow x \rangle\rangle \wedge g = e \cdot (f \otimes b)))
\end{aligned}$$

**definition** *some-eval* ( $\langle\text{eval}^?\rangle$ )

**where**  $\text{eval}^? b c \equiv \text{SOME } e. \langle\langle e : \text{exp}^? b c \otimes b \rightarrow c \rangle\rangle \wedge$   
 $(\forall a g. \text{ide } a \wedge \langle\langle g : a \otimes b \rightarrow c \rangle\rangle$   
 $\quad \rightarrow (\exists ! f. \langle\langle f : a \rightarrow \text{exp}^? b c \rangle\rangle \wedge g = e \cdot (f \otimes b)))$

**definition** *some-Curry* ( $\langle\text{Curry}^?[-, -, -]\rangle$ )

**where**  $\text{Curry}^?[a, b, c] g \equiv$   
 $\text{THE } f. \langle\langle f : a \rightarrow \text{exp}^? b c \rangle\rangle \wedge g = \text{eval}^? b c \cdot (f \otimes b)$

**abbreviation** *some-Uncurry* ( $\langle\text{Uncurry}^?[-, -]\rangle$ )

**where**  $\text{Uncurry}^?[b, c] f \equiv \text{eval}^? b c \cdot (f \otimes b)$

**lemma** *Curry-uniqueness*:

**assumes** *ide b and ide c*

**shows** *ide (exp<sup>?</sup> b c) and*  $\langle\langle \text{eval}^? b c : \text{exp}^? b c \otimes b \rightarrow c \rangle\rangle$

**and**  $\llbracket \text{ide } a; \langle\langle g : a \otimes b \rightarrow c \rangle\rangle \rrbracket$

$\implies \exists ! f. \langle\langle f : a \rightarrow \text{exp}^? b c \rangle\rangle \wedge g = \text{Uncurry}^?[b, c] f$

**using** *assms some-exp-def some-eval-def has-exponentials*

*someI-ex [of  $\lambda x. \text{ide } x \wedge (\exists e. \langle\langle e : x \otimes b \rightarrow c \rangle\rangle \wedge$*

$(\forall a g. \text{ide } a \wedge \langle\langle g : a \otimes b \rightarrow c \rangle\rangle$

$\quad \rightarrow (\exists ! f. \langle\langle f : a \rightarrow x \rangle\rangle \wedge g = e \cdot (f \otimes b)))]$

*someI-ex [of  $\lambda e. \langle\langle e : \text{exp}^? b c \otimes b \rightarrow c \rangle\rangle \wedge$*

$(\forall a g. \text{ide } a \wedge \langle\langle g : a \otimes b \rightarrow c \rangle\rangle$

$\quad \rightarrow (\exists ! f. \langle\langle f : a \rightarrow \text{exp}^? b c \rangle\rangle \wedge g = e \cdot (f \otimes b)))]$

**by** *auto*

**lemma** *some-eval-in-hom [intro]*:

**assumes** *ide b and ide c and*  $x = \text{exp}^? b c \otimes b$

**shows**  $\langle\langle \text{eval}^? b c : x \rightarrow c \rangle\rangle$

**using** *assms Curry-uniqueness by simp*

**lemma** *some-Uncurry-some-Curry*:

**assumes** *ide a and ide b and*  $\langle\langle g : a \otimes b \rightarrow c \rangle\rangle$

**shows**  $\langle\langle \text{Curry}^?[a, b, c] g : a \rightarrow \text{exp}^? b c \rangle\rangle$

**and**  $\text{Uncurry}^?[b, c] (\text{Curry}^?[a, b, c] g) = g$

**proof** –

**have** *ide c*

**using** *assms(3) by auto*

**hence** *1*:  $\langle\langle \text{Curry}^?[a, b, c] g : a \rightarrow \text{exp}^? b c \rangle\rangle \wedge$

$g = \text{Uncurry}^?[b, c] (\text{Curry}^?[a, b, c] g)$

**using** *assms some-Curry-def Curry-uniqueness*

*theI' [of  $\lambda f. \langle\langle f : a \rightarrow \text{exp}^? b c \rangle\rangle \wedge g = \text{Uncurry}^?[b, c] f$*

**by** *simp*

**show**  $\langle\langle \text{Curry}^?[a, b, c] g : a \rightarrow \text{exp}^? b c \rangle\rangle$

**using** *1 by simp*

**show**  $\text{Uncurry}^?[b, c] (\text{Curry}^?[a, b, c] g) = g$

using 1 by simp  
qed

**lemma** *some-Curry-some-Uncurry:*

**assumes** *ide b and ide c and*  $\langle h : a \rightarrow \text{exp}^? b c \rangle$

**shows**  $\text{Curry}^?[a, b, c] (\text{Uncurry}^?[b, c] h) = h$

**proof** –

**have**  $\exists! f. \langle f : a \rightarrow \text{exp}^? b c \rangle \wedge \text{Uncurry}^?[b, c] h = \text{Uncurry}^?[b, c] f$

**using** *assms ide-dom ide-in-hom*

*Curry-uniqueness*( $\exists$ ) [of b c a  $\text{Uncurry}^?[b, c] h$ ]

**by** *auto*

**moreover have**  $\langle h : a \rightarrow \text{exp}^? b c \rangle \wedge \text{Uncurry}^?[b, c] h = \text{Uncurry}^?[b, c] h$

**using** *assms by simp*

**ultimately show** *?thesis*

**using** *assms some-Curry-def Curry-uniqueness some-Uncurry-some-Curry*

*the1-equality* [of  $\lambda f. \langle f : a \rightarrow \text{some-exp } b c \rangle \wedge$

$\text{Uncurry}^?[b, c] h = \text{Uncurry}^?[b, c] f$ ]

**by** *simp*

qed

**lemma** *extends-to-elementary-closed-monoidal-category<sub>CMC</sub>:*

**shows** *elementary-closed-monoidal-category*

*C T  $\alpha$   $\iota$  some-exp some-eval some-Curry*

**using** *Curry-uniqueness some-Uncurry-some-Curry*

*some-Curry-some-Uncurry*

**by** *unfold-locales auto*

end

**context** *closed-symmetric-monoidal-category*

**begin**

**lemma** *extends-to-elementary-closed-symmetric-monoidal-category<sub>CMC</sub>:*

**shows** *elementary-closed-symmetric-monoidal-category*

*C T  $\alpha$   $\iota$   $\sigma$  some-exp some-eval some-Curry*

**by** (*simp add: elementary-closed-symmetric-monoidal-category-def*

*extends-to-elementary-closed-monoidal-category<sub>CMC</sub>*

*symmetric-monoidal-category-axioms*)

end

## 1.2 Internal Hom Functors

For each object  $x$  of a closed monoidal category  $C$ , we can define a covariant endofunctor  $\text{Exp}^{\rightarrow} x -$  of  $C$ , which takes each arrow  $g$  to an arrow  $\langle \text{Exp}^{\rightarrow} x g : \text{exp } x (\text{dom } g) \rightarrow \text{exp } x (\text{cod } g) \rangle$ . Similarly, for each object  $y$ , we can define a contravariant endofunctor  $\text{Exp}^{\leftarrow} - y$  of  $C$ , which takes each arrow  $f$  of  $C^{\text{op}}$  to an arrow  $\langle \text{Exp}^{\leftarrow} f y : \text{exp } (\text{cod } f) y \rightarrow \text{exp } (\text{dom } f) y \rangle$  of  $C$ .

These two endofunctors commute with each other and compose to form a single binary “internal hom” functor  $Exp$  from  $C^{op} \times C$  to  $C$ .

**context** *elementary-closed-monoidal-category*  
**begin**

**abbreviation** *cov-Exp* ( $\langle Exp^{\rightarrow} \rangle$ )

**where**  $Exp^{\rightarrow} x g \equiv$  if *arr*  $g$   
     then  $Curry[exp\ x\ (dom\ g),\ x,\ cod\ g]\ (g \cdot eval\ x\ (dom\ g))$   
     else *null*

**abbreviation** *cnt-Exp* ( $\langle Exp^{\leftarrow} \rangle$ )

**where**  $Exp^{\leftarrow} f y \equiv$  if *arr*  $f$   
     then  $Curry[exp\ (cod\ f)\ y,\ dom\ f,\ y]$   
     ( $eval\ (cod\ f)\ y \cdot (exp\ (cod\ f)\ y \otimes f)$ )  
     else *null*

**lemma** *cov-Exp-in-hom*:

**assumes** *ide*  $x$  **and** *arr*  $g$

**shows**  $\langle Exp^{\rightarrow} x g : exp\ x\ (dom\ g) \rightarrow exp\ x\ (cod\ g) \rangle$

**using** *assms* **by** *auto*

**lemma** *cnt-Exp-in-hom*:

**assumes** *arr*  $f$  **and** *ide*  $y$

**shows**  $\langle Exp^{\leftarrow} f y : exp\ (cod\ f)\ y \rightarrow exp\ (dom\ f)\ y \rangle$

**using** *assms* **by** *force*

**lemma** *cov-Exp-ide*:

**assumes** *ide*  $a$  **and** *ide*  $b$

**shows**  $Exp^{\rightarrow} a\ b = exp\ a\ b$

**using** *assms*

**by** (*metis comp-ide-arr Curry-Uncurry eval-in-hom<sub>ECMC</sub> ideD(2-3) ide-exp ide-in-hom seqI' Uncurry-exp*)

**lemma** *cnt-Exp-ide*:

**assumes** *ide*  $a$  **and** *ide*  $b$

**shows**  $Exp^{\leftarrow} a\ b = exp\ a\ b$

**using** *assms* *Curry-Uncurry ide-exp ide-in-hom* **by** *force*

**lemma** *cov-Exp-comp*:

**assumes** *ide*  $x$  **and** *seq*  $g\ f$

**shows**  $Exp^{\rightarrow} x\ (g \cdot f) = Exp^{\rightarrow} x\ g \cdot Exp^{\rightarrow} x\ f$

**proof** –

**have**  $Exp^{\rightarrow} x\ g \cdot Exp^{\rightarrow} x\ f =$

$Curry[exp\ x\ (cod\ f),\ x,\ cod\ g]\ (g \cdot eval\ x\ (cod\ f)) \cdot$

$Curry[exp\ x\ (dom\ f),\ x,\ cod\ f]\ (f \cdot eval\ x\ (dom\ f))$

**using** *assms* **by** *auto*

**also have**  $\dots = Curry[exp\ x\ (dom\ f),\ x,\ cod\ g]$

$((g \cdot eval\ x\ (dom\ g)) \cdot$

$(Curry[exp\ x\ (dom\ f),\ x,\ cod\ f]\ (f \cdot eval\ x\ (dom\ f)) \otimes x))$

**using** *assms cov-Exp-in-hom comp-Curry-arr* **by** *auto*  
**also have**  $\dots = \text{Exp}^{\rightarrow} x (g \cdot f)$   
**using** *assms Uncurry-Curry comp-assoc* **by** *fastforce*  
**finally show** *?thesis* **by** *simp*  
**qed**

**lemma** *cnt-Exp-comp*:

**assumes** *seq g f* **and** *ide y*

**shows**  $\text{Exp}^{\leftarrow} (g \cdot f) y = \text{Exp}^{\leftarrow} f y \cdot \text{Exp}^{\leftarrow} g y$

**proof** –

**have**  $\text{Exp}^{\leftarrow} f y \cdot \text{Exp}^{\leftarrow} g y =$   
 $\text{Curry}[\text{exp} (\text{cod } g) y, \text{dom } f, y]$   
 $((\text{eval} (\text{cod } f) y \cdot (\text{exp} (\text{cod } f) y \otimes f)) \cdot$   
 $(\text{Curry}[\text{exp} (\text{cod } g) y, \text{cod } f, y]$   
 $(\text{eval} (\text{cod } g) y \cdot (\text{exp} (\text{cod } g) y \otimes g)) \otimes \text{dom } f))$

**using** *assms*

*comp-Curry-arr*

$[\text{of } \text{dom } f \text{ Curry}[\text{exp} (\text{cod } g) y, \text{cod } f, y]$   
 $(\text{eval} (\text{cod } g) y \cdot (\text{exp} (\text{cod } g) y \otimes g))]$

**by** *fastforce*

**also have**  $\dots = \text{Curry}[\text{exp} (\text{cod } g) y, \text{dom } f, y]$   
 $((\text{Uncurry}[\text{cod } f, y]$   
 $(\text{Curry}[\text{exp} (\text{cod } g) y, \text{cod } f, y]$   
 $(\text{eval} (\text{cod } g) y \cdot (\text{exp} (\text{cod } g) y \otimes g)))) \cdot$   
 $(\text{exp} (\text{cod } g) y \otimes f))$

**using** *assms interchange comp-arr-dom comp-cod-arr comp-assoc* **by** *auto*

**also have**  $\dots = \text{Curry}[\text{exp} (\text{cod } g) y, \text{dom } f, y]$   
 $((\text{eval} (\text{cod } g) y \cdot (\text{exp} (\text{cod } g) y \otimes g)) \cdot (\text{exp} (\text{cod } g) y \otimes f))$

**using** *assms Uncurry-Curry* **by** *auto*

**also have**  $\dots = \text{Exp}^{\leftarrow} (g \cdot f) y$

**using** *assms interchange comp-assoc* **by** *auto*

**finally show** *?thesis* **by** *simp*

**qed**

**lemma** *functor-cov-Exp*:

**assumes** *ide x*

**shows** *functor C C* ( $\text{Exp}^{\rightarrow} x$ )

**using** *assms cov-Exp-ide cov-Exp-in-hom cov-Exp-comp*

**by** *unfold-locales auto*

**interpretation** *Cop*: *dual-category C ..*

**lemma** *functor-cnt-Exp*:

**assumes** *ide x*

**shows** *functor Cop.comp C* ( $\lambda f. \text{Exp}^{\leftarrow} f x$ )

**using** *assms cnt-Exp-ide cnt-Exp-in-hom cnt-Exp-comp*

**by** *unfold-locales auto*

**lemma** *cov-cnt-Exp-commute*:

**assumes**  $arr\ f$  **and**  $arr\ g$   
**shows**  $Exp^{\rightarrow} (dom\ f)\ g \cdot Exp^{\leftarrow} f (dom\ g) =$   
 $Exp^{\leftarrow} f (cod\ g) \cdot Exp^{\rightarrow} (cod\ f)\ g$   
**proof** –  
**have**  $Exp^{\rightarrow} (dom\ f)\ g \cdot Exp^{\leftarrow} f (dom\ g) =$   
 $Curry[exp\ (cod\ f)\ (dom\ g),\ dom\ f,\ cod\ g]$   
 $((g \cdot eval\ (dom\ f)\ (dom\ g)) \cdot$   
 $(Curry[exp\ (cod\ f)\ (dom\ g),\ dom\ f,\ dom\ g]$   
 $(eval\ (cod\ f)\ (dom\ g) \cdot (exp\ (cod\ f)\ (dom\ g) \otimes f)) \otimes dom\ f))$   
**using**  $assms\ cnt-Exp-in-hom\ comp-Curry-arr$  **by**  $force$   
**also have**  $\dots = Curry[exp\ (cod\ f)\ (dom\ g),\ dom\ f,\ cod\ g]$   
 $(Uncurry[cod\ f,\ cod\ g]\ (Exp^{\rightarrow} (cod\ f)\ g) \cdot$   
 $(exp\ (cod\ f)\ (dom\ g) \otimes f))$   
**using**  $assms\ comp-assoc\ Uncurry-Curry$  **by**  $auto$   
**also have**  $\dots = Curry[exp\ (cod\ f)\ (dom\ g),\ dom\ f,\ cod\ g]$   
 $(eval\ (cod\ f)\ (cod\ g) \cdot (Exp^{\rightarrow} (cod\ f)\ g \otimes cod\ f) \cdot$   
 $(exp\ (cod\ f)\ (dom\ g) \otimes f))$   
**using**  $comp-assoc$  **by**  $auto$   
**also have**  $\dots = Curry[exp\ (cod\ f)\ (dom\ g),\ dom\ f,\ cod\ g]$   
 $(eval\ (cod\ f)\ (cod\ g) \cdot (Exp^{\rightarrow} (cod\ f)\ g \otimes f))$   
**using**  $assms\ interchange\ comp-arr-dom\ comp-cod-arr$   
**by**  $(metis\ cov-Exp-in-hom\ ide-cod\ in-homE)$   
**also have**  $\dots = Curry[exp\ (cod\ f)\ (dom\ g),\ dom\ f,\ cod\ g]$   
 $(eval\ (cod\ f)\ (cod\ g) \cdot$   
 $(exp\ (cod\ f)\ (cod\ g) \otimes f) \cdot (Exp^{\rightarrow} (cod\ f)\ g \otimes dom\ f))$   
**using**  $assms\ interchange\ comp-arr-dom\ comp-cod-arr\ cov-Exp-in-hom$   
**by**  $auto$   
**also have**  $\dots = Exp^{\leftarrow} f (cod\ g) \cdot Exp^{\rightarrow} (cod\ f)\ g$   
**using**  $assms\ cov-Exp-in-hom\ comp-assoc$   
 $comp-Curry-arr$   
 $[of\ dom\ f\ Exp^{\rightarrow} (cod\ f)\ g\ exp\ (cod\ f)\ (dom\ g) -$   
 $eval\ (cod\ f)\ (cod\ g) \cdot (exp\ (cod\ f)\ (cod\ g) \otimes f)\ cod\ g]$   
**by**  $simp$   
**finally show**  $?thesis$  **by**  $simp$   
**qed**

**definition**  $Exp$

**where**  $Exp\ f\ g \equiv Exp^{\rightarrow} (dom\ f)\ g \cdot Exp^{\leftarrow} f (dom\ g)$

**lemma**  $Exp-in-hom$ :

**assumes**  $arr\ f$  **and**  $arr\ g$

**shows**  $\ll Exp\ f\ g : Exp\ (cod\ f)\ (dom\ g) \rightarrow Exp\ (dom\ f)\ (cod\ g) \gg$

**using**  $Exp-def\ assms(1-2)\ cnt-Exp-ide\ cov-Exp-ide$  **by**  $auto$

**lemma**  $Exp-ide$ :

**assumes**  $ide\ a$  **and**  $ide\ b$

**shows**  $Exp\ a\ b = exp\ a\ b$

**unfolding**  $Exp-def$

**using**  $assms\ cov-Exp-ide\ cnt-Exp-ide$  **by**  $simp$

**lemma** *Exp-comp*:  
**assumes** *seq g f and seq k h*  
**shows**  $Exp (g \cdot f) (k \cdot h) = Exp f k \cdot Exp g h$   
**proof** –  
**have**  $Exp (g \cdot f) (k \cdot h) = Exp^{\rightarrow} (dom f) (k \cdot h) \cdot Exp^{\leftarrow} (g \cdot f) (dom h)$   
**unfolding** *Exp-def*  
**using** *assms by auto*  
**also have**  $\dots = (Exp^{\rightarrow} (dom f) k \cdot Exp^{\rightarrow} (dom f) h) \cdot$   
 $(Exp^{\leftarrow} f (dom h) \cdot Exp^{\leftarrow} g (dom h))$   
**using** *assms cov-Exp-comp cnt-Exp-comp by auto*  
**also have**  $\dots = (Exp^{\rightarrow} (dom f) k \cdot Exp^{\leftarrow} f (dom k)) \cdot$   
 $(Exp^{\rightarrow} (dom g) h \cdot Exp^{\leftarrow} g (dom h))$   
**using** *assms comp-assoc cov-cnt-Exp-commute*  
**by** (*metis (no-types, lifting) seqE*)  
**also have**  $\dots = Exp f k \cdot Exp g h$   
**unfolding** *Exp-def by blast*  
**finally show** *?thesis by blast*  
**qed**

**interpretation** *CopxC*: *product-category Cop.comp C ..*

**lemma** *functor-Exp*:  
**shows** *binary-functor Cop.comp C C* ( $\lambda fg. Exp (fst fg) (snd fg)$ )  
**using** *Exp-in-hom*  
**apply** *unfold-locales*  
**apply** *auto[4]*  
**using** *Exp-def*  
**apply** *auto[2]*  
**using** *Exp-comp*  
**by** *fastforce*

**lemma** *Exp-x-ide*:  
**assumes** *ide y*  
**shows**  $(\lambda x. Exp x y) = (\lambda x. Exp^{\leftarrow} x y)$   
**using** *assms Exp-ide Exp-def comp-cod-arr cov-Exp-ide by auto*

**lemma** *Exp-ide-y*:  
**assumes** *ide x*  
**shows**  $(\lambda y. Exp x y) = (\lambda y. Exp^{\rightarrow} x y)$   
**using** *assms Exp-ide Exp-def comp-arr-dom cnt-Exp-ide by auto*

**lemma** *Uncurry-Exp-dom*:  
**assumes** *arr f*  
**shows**  $Uncurry (dom f) (cod f) (Exp (dom f) f) = f \cdot eval (dom f) (dom f)$   
**proof** –  
**have**  $Uncurry[dom f, cod f] (Exp (dom f) f) =$   
 $Uncurry[dom f, cod f]$   
 $(Curry[exp (dom f) (dom f), dom f, cod f] (f \cdot eval (dom f) (dom f))) \cdot$

$\text{Curry}[\text{exp } (\text{dom } f) (\text{dom } f), \text{dom } f, \text{dom } f] (\text{eval } (\text{dom } f) (\text{dom } f))$   
**unfolding** *Exp-def*  
**using** *assms Curry-Uncurry comp-arr-dom by simp*  
**also have**  $\dots = \text{Uncurry}[\text{dom } f, \text{cod } f]$   
 $(\text{Curry}[\text{exp } (\text{dom } f) (\text{dom } f), \text{dom } f, \text{cod } f]$   
 $((f \cdot \text{eval } (\text{dom } f) (\text{dom } f)) \cdot$   
 $(\text{Curry}[\text{exp } (\text{dom } f) (\text{dom } f), \text{dom } f, \text{dom } f]$   
 $(\text{eval } (\text{dom } f) (\text{dom } f)) \otimes \text{dom } f))$   
**using** *assms comp-Curry-arr*  
**by** (*metis comp-in-homI' Curry-in-hom eval-in-hom<sub>ECMC</sub> ide-dom*  
*ide-exp in-homE*)  
**also have**  $\dots = f \cdot \text{eval } (\text{dom } f) (\text{dom } f)$   
**using** *assms Uncurry-Curry eval-in-hom<sub>ECMC</sub> comp-assoc by simp*  
**finally show** *?thesis by simp*  
**qed**

### 1.2.1 Exponentiation by Unity

In this section we define and develop the properties of inverse arrows  $Up\ a : a \rightarrow \text{exp } \mathcal{I}\ a$  and  $Dn\ a : \text{exp } \mathcal{I}\ a \rightarrow a$ , which exist in any closed monoidal category.

**interpretation** *elementary-monoidal-category C tensor unity lunit runit assoc*  
**using** *induces-elementary-monoidal-category by blast*

**abbreviation**  $Up$   
**where**  $Up\ a \equiv \text{Curry}[a, \mathcal{I}, a]\ \text{r}[a]$

**abbreviation**  $Dn$   
**where**  $Dn\ a \equiv \text{eval } \mathcal{I}\ a \cdot \text{r}^{-1}[\text{exp } \mathcal{I}\ a]$

**lemma** *isomorphic-exp-unity:*  
**assumes** *ide a*  
**shows**  $\langle\langle Up\ a : a \rightarrow \text{exp } \mathcal{I}\ a \rangle\rangle$   
**and**  $\langle\langle Dn\ a : \text{exp } \mathcal{I}\ a \rightarrow a \rangle\rangle$   
**and** *inverse-arrows (Up a) (Dn a)*  
**and** *isomorphic (exp I a) a*

**proof** –

**show** 1:  $\langle\langle Up\ a : a \rightarrow \text{exp } \mathcal{I}\ a \rangle\rangle$   
**using** *assms ide-unity Curry-in-hom by blast*  
**show** 2:  $\langle\langle Dn\ a : \text{exp } \mathcal{I}\ a \rightarrow a \rangle\rangle$   
**using** *assms eval-in-hom<sub>ECMC</sub> [of I a] runit-in-hom ide-unity by blast*  
**show** *inverse-arrows (Up a) (Dn a)*

**proof**

**show** *ide ((Dn a) · Up a)*  
**by** (*metis (no-types, lifting) <<Up a : a → exp I a>>*  
*assms comp-runit-runit'(1) ide-unity in-homE comp-assoc*  
*runit'-naturality runit-in-hom Uncurry-Curry*)

**show** *ide (Up a · Dn a)*

**proof** –



**have**  $Up\ a \cdot Dn\ a = (Curry[a, \mathcal{I}, a]\ r[a] \cdot eval\ \mathcal{I}\ a) \cdot r^{-1}[exp\ \mathcal{I}\ a]$   
**using** *comp-assoc by simp*  
**also have** ... =  
 $Curry[exp\ \mathcal{I}\ a \otimes \mathcal{I}, \mathcal{I}, a]\ (r[a] \cdot (eval\ \mathcal{I}\ a \otimes \mathcal{I})) \cdot r^{-1}[exp\ \mathcal{I}\ a]$   
**using** *assms comp-Curry-arr*  
**by** (*metis eval-in-hom-ax ide-unity runit-in-hom*)  
**also have** ... =  
 $Curry[exp\ \mathcal{I}\ a \otimes \mathcal{I}, \mathcal{I}, a]\ (eval\ \mathcal{I}\ a \cdot r[exp\ \mathcal{I}\ a \otimes \mathcal{I}]) \cdot r^{-1}[exp\ \mathcal{I}\ a]$   
**using** *assms runit-naturality*  
**by** (*metis (no-types, lifting) eval-in-hom<sub>ECMC</sub> ide-unity in-homE*)  
**also have** ... =  $(Curry[exp\ \mathcal{I}\ a, \mathcal{I}, a]\ (eval\ \mathcal{I}\ a) \cdot r[exp\ \mathcal{I}\ a]) \cdot r^{-1}[exp\ \mathcal{I}\ a]$   
**by** (*metis assms comp-Curry-arr eval-in-hom<sub>ECMC</sub> ide-exp ide-unity runit-commutes-with-R runit-in-hom*)  
**also have** ... =  $Curry[exp\ \mathcal{I}\ a, \mathcal{I}, a]\ (eval\ \mathcal{I}\ a) \cdot r[exp\ \mathcal{I}\ a] \cdot r^{-1}[exp\ \mathcal{I}\ a]$   
**using** *comp-assoc by simp*  
**also have** ... =  $Curry[exp\ \mathcal{I}\ a, \mathcal{I}, a]\ (eval\ \mathcal{I}\ a)$   
**by** (*metis assms 1 2 calculation comp-arr-ide comp-runit-runit'(1) ide-exp ide-unity seqI'*)  
**also have** ... =  $exp\ \mathcal{I}\ a$   
**using** *assms Curry-Uncurry*  
**by** (*metis ide-exp ide-in-hom ide-unity Uncurry-exp*)  
**finally show** *?thesis*  
**using** *assms ide-exp ide-unity by presburger*  
**qed**  
**qed**  
**thus** *isomorphic (exp I a) a*  
**by** (*metis «Up a : a → exp I a» in-homE isoI isomorphicI isomorphic-symmetric*)  
**qed**

The maps  $Up$  and  $Dn$  are natural in  $a$ .

**lemma** *Up-Dn-naturality:*

**assumes** *arr f*

**shows**  $Exp^{\rightarrow}\ \mathcal{I}\ f \cdot Up\ (dom\ f) = Up\ (cod\ f) \cdot f$

**and**  $Dn\ (cod\ f) \cdot Exp^{\rightarrow}\ \mathcal{I}\ f = f \cdot Dn\ (dom\ f)$

**proof** –

**show** *1:  $Exp^{\rightarrow}\ \mathcal{I}\ f \cdot Up\ (dom\ f) = Up\ (cod\ f) \cdot f$*

**proof** –

**have**  $Exp^{\rightarrow}\ \mathcal{I}\ f \cdot Up\ (dom\ f) =$

$Curry[dom\ f, \mathcal{I}, cod\ f]$

$((f \cdot eval\ \mathcal{I}\ (dom\ f)) \cdot (Curry[dom\ f, \mathcal{I}, dom\ f]\ r[dom\ f] \otimes \mathcal{I}))$

**using** *assms comp-Curry-arr isomorphic-exp-unity(1) by auto*

**also have** ... =  $Curry[dom\ f, \mathcal{I}, cod\ f]\ (r[cod\ f] \cdot (f \otimes \mathcal{I}))$

**using** *assms comp-assoc Uncurry-Curry runit-naturality by simp*

**also have** ... =  $Up\ (cod\ f) \cdot f$

**by** (*metis assms comp-Curry-arr ide-cod ide-unity in-homI runit-in-hom*)

**finally show** *?thesis by blast*

**qed**

**have**  $Exp^{\rightarrow}\ \mathcal{I}\ f \cdot inv\ (Dn\ (dom\ f)) = inv\ (Dn\ (cod\ f)) \cdot f$

**using** *assms 1 isomorphic-exp-unity isomorphic-exp-unity*  
**by** (*metis ide-cod ide-dom inverse-arrows-sym inverse-unique*)  
**moreover have** 2: *iso (Dn (cod f))*  
**using** *assms isomorphic-exp-unity [of cod f]* **by** *auto*  
**moreover have** 3: *iso (Dn (dom f))*  
**using** *assms isomorphic-exp-unity [of dom f]* **by** *auto*  
**moreover have** *seq (inv (Dn (cod f))) f*  
**using** *assms 2* **by** *auto*  
**ultimately show** *Dn (cod f) · Exp<sup>→</sup> I f = f · Dn (dom f)*  
**using** *assms 2 3 inv-inv iso-inv-iso comp-assoc isomorphic-exp-unity*  
*invert-opposite-sides-of-square*  
*[of inv (eval I (cod f) · r<sup>-1</sup>[exp I (cod f)]) f Exp<sup>→</sup> I f*  
*inv (eval I (dom f) · r<sup>-1</sup>[exp I (dom f)])]*  
**by** *metis*  
**qed**

### 1.2.2 Internal Currying

Currying internalizes to an isomorphism between  $\text{exp } (x \otimes a) b$  and  $\text{exp } x (a b)$ .

**abbreviation** *curry*  
**where** *curry x b c*  $\equiv$   
 $\text{Curry}[\text{exp } (x \otimes b) c, x, \text{exp } b c]$   
 $(\text{Curry}[\text{exp } (x \otimes b) c \otimes x, b, c])$   
 $(\text{eval } (x \otimes b) c \cdot a[\text{exp } (x \otimes b) c, x, b])$

**abbreviation** *uncurry*  
**where** *uncurry x b c*  $\equiv$   
 $\text{Curry}[\text{exp } x (a b c), x \otimes b, c]$   
 $(\text{eval } b c \cdot (\text{eval } x (a b c) \otimes b) \cdot a^{-1}[\text{exp } x (a b c), x, b])$

**lemma** *internal-curry:*

**assumes** *ide x and ide a and ide b*

**shows**  $\langle\langle \text{curry } x a b : \text{exp } (x \otimes a) b \rightarrow \text{exp } x (a b) \rangle\rangle$

**and**  $\langle\langle \text{uncurry } x a b : \text{exp } x (a b) \rightarrow \text{exp } (x \otimes a) b \rangle\rangle$

**and** *inverse-arrows (curry x a b) (uncurry x a b)*

**proof** –

**show** 1:  $\langle\langle \text{curry } x a b : \text{exp } (x \otimes a) b \rightarrow \text{exp } x (a b) \rangle\rangle$

**using** *assms*

**by** (*meson assoc-in-hom comp-in-homI Curry-in-hom eval-in-hom<sub>ECMC</sub>*  
*ide-exp tensor-preserves-ide*)

**show** 2:  $\langle\langle \text{uncurry } x a b : \text{exp } x (a b) \rightarrow \text{exp } (x \otimes a) b \rangle\rangle$

**using** *assms ide-exp* **by** *auto*

**show** *inverse-arrows (curry x a b) (uncurry x a b)*

(**is** *inverse-arrows*

$(\text{Curry } (\text{exp } (x \otimes a) b) x (\text{exp } a b)$

$(\text{Curry } (\text{exp } (x \otimes a) b \otimes x) a b ?F))$

$(\text{Curry } (\text{exp } x (a b)) (x \otimes a) b ?G))$

**proof**

```

have F: «?F : (exp (x ⊗ a) b ⊗ x) ⊗ a → b»
  using assms ide-exp by simp
have G: «?G : exp x (exp a b) ⊗ x ⊗ a → b»
  using assms ide-exp by auto
show ide (uncurry x a b · curry x a b)
proof -
  have uncurry x a b · curry x a b =
    Curry[exp (x ⊗ a) b, x ⊗ a, b] (?G · (curry x a b ⊗ x ⊗ a))
  using assms F 1 ide-exp comp-Curry-arr comp-assoc by auto
  also have ... = Curry[exp (x ⊗ a) b, x ⊗ a, b]
    (eval a b · (eval x (exp a b) ⊗ a) · a-1[exp x (exp a b), x, a] ·
      (curry x a b ⊗ x ⊗ a))
  using comp-assoc by simp
  also have ... = Curry[exp (x ⊗ a) b, x ⊗ a, b]
    (eval a b · (eval x (exp a b) ⊗ a) ·
      ((curry x a b ⊗ x) ⊗ a) · a-1[exp (x ⊗ a) b, x, a])
  using assms 1 comp-assoc assoc'-naturality [of curry x a b x a]
    ide-char in-homE
  by metis
  also have ... = Curry[exp (x ⊗ a) b, x ⊗ a, b]
    (eval a b · ((eval x (exp a b) ⊗ a) · ((curry x a b ⊗ x) ⊗ a)) ·
      a-1[exp (x ⊗ a) b, x, a])
  using comp-assoc by simp
  also have ... = Curry[exp (x ⊗ a) b, x ⊗ a, b]
    (eval a b · (Uncurry[x, exp a b] (curry x a b) ⊗ a) ·
      a-1[exp (x ⊗ a) b, x, a])
  using assms comp-ide-self
    interchange [of eval x (exp a b)
      Curry[exp (x ⊗ a) b, x, exp a b]
      (Curry[exp (x ⊗ a) b ⊗ x, a, b] ?F) ⊗ x
      a a]
  by fastforce
  also have ... = Curry[exp (x ⊗ a) b, x ⊗ a, b]
    (eval a b ·
      (Curry[exp (x ⊗ a) b ⊗ x, a, b] ?F ⊗ a) ·
      a-1[exp (x ⊗ a) b, x, a])
  using assms F ide-exp comp-assoc comp-ide-self
    Uncurry-Curry
    [of exp (x ⊗ a) b x exp a b Curry[exp (x ⊗ a) b ⊗ x, a, b] ?F]
  by fastforce
  also have ... = Curry[exp (x ⊗ a) b, x ⊗ a, b]
    (eval (x ⊗ a) b · a[exp (x ⊗ a) b, x, a] ·
      a-1[exp (x ⊗ a) b, x, a])
  using assms Uncurry-Curry
  by (metis F ide-exp comp-assoc tensor-preserves-ide)
  also have ... = Curry[exp (x ⊗ a) b, x ⊗ a, b] (eval (x ⊗ a) b)
  using assms Uncurry-exp by simp
  also have ... = exp (x ⊗ a) b
  using assms Curry-Uncurry

```

by (*metis Curry-Uncurry ide-exp ide-in-hom tensor-preserves-ide*  
*Uncurry-exp*)  
 finally have  $\text{uncurry } x \ a \ b \cdot \text{curry } x \ a \ b = \text{exp } (x \otimes a) \ b$   
 by *blast*  
 thus *?thesis*  
 using *assms by simp*  
**qed**  
 show *ide* ( $\text{curry } x \ a \ b \cdot \text{uncurry } x \ a \ b$ )  
**proof** –  
 have  $\text{curry } x \ a \ b \cdot \text{uncurry } x \ a \ b =$   
 $\text{Curry}[\text{exp } x \ (\text{exp } a \ b), x, \text{exp } a \ b]$   
 $(\text{Curry}[\text{exp } (x \otimes a) \ b \otimes x, a, b] \ ?F \cdot (\text{uncurry } x \ a \ b \otimes x))$   
 using *assms 2 F Curry-in-hom comp-Curry-arr by simp*  
 also have  $\dots = \text{Curry}[\text{exp } x \ (\text{exp } a \ b), x, \text{exp } a \ b]$   
 $(\text{Curry}[\text{exp } x \ (\text{exp } a \ b) \otimes x, a, b]$   
 $(\text{eval } (x \otimes a) \ b \cdot \text{a}[\text{exp } (x \otimes a) \ b, x, a] \cdot$   
 $((\text{uncurry } x \ a \ b \otimes x) \otimes a)))$   
**proof** –  
 have  $\text{Curry}[\text{exp } (x \otimes a) \ b \otimes x, a, b] \ ?F \cdot (\text{uncurry } x \ a \ b \otimes x) =$   
 $\text{Curry}[\text{exp } x \ (\text{exp } a \ b) \otimes x, a, b] \ (?F \cdot ((\text{uncurry } x \ a \ b \otimes x) \otimes a))$   
 using *assms(1-2) 2 F comp-Curry-arr ide-in-hom by auto*  
 thus *?thesis*  
 using *comp-assoc by simp*  
**qed**  
 also have  $\dots = \text{Curry}[\text{exp } x \ (\text{exp } a \ b), x, \text{exp } a \ b]$   
 $(\text{Curry}[\text{exp } x \ (\text{exp } a \ b) \otimes x, a, b]$   
 $(\text{eval } (x \otimes a) \ b \cdot$   
 $(\text{uncurry } x \ a \ b \otimes x \otimes a) \cdot \text{a}[\text{exp } x \ (\text{exp } a \ b), x, a]))$   
 using *assms 2*  
*assoc-naturality* [*of Curry (exp x (exp a b)) (x ⊗ a) b ?G x a*]  
 by *auto*  
 also have  $\dots = \text{Curry}[\text{exp } x \ (\text{exp } a \ b), x, \text{exp } a \ b]$   
 $(\text{Curry}[\text{exp } x \ (\text{exp } a \ b) \otimes x, a, b]$   
 $(\text{eval } a \ b \cdot (\text{eval } x \ (\text{exp } a \ b) \otimes a) \cdot$   
 $\text{a}^{-1}[\text{exp } x \ (\text{exp } a \ b), x, a] \cdot \text{a}[\text{exp } x \ (\text{exp } a \ b), x, a]))$   
 using *assms Uncurry-Curry*  
 by (*metis G ide-exp comp-assoc tensor-preserves-ide*)  
 also have  $\dots = \text{Curry}[\text{exp } x \ (\text{exp } a \ b), x, \text{exp } a \ b]$   
 $(\text{Curry}[\text{exp } x \ (\text{exp } a \ b) \otimes x, a, b]$   
 $(\text{Uncurry}[a, b] (\text{eval } x \ (\text{exp } a \ b))))$   
 using *assms*  
 by (*metis G arrI cod-assoc' comp-arr-dom comp-assoc-assoc'(2)*  
*ide-exp seqE*)  
 also have  $\dots = \text{Curry}[\text{exp } x \ (\text{exp } a \ b), x, \text{exp } a \ b] (\text{eval } x \ (\text{exp } a \ b))$   
 by (*simp add: assms(1-3) Curry-Uncurry eval-in-hom<sub>EMC</sub>*)  
 also have  $\dots = \text{exp } x \ (\text{exp } a \ b)$   
 using *assms Curry-Uncurry Uncurry-exp*  
 by (*metis ide-exp ide-in-hom*)  
 finally have  $\text{curry } x \ a \ b \cdot \text{uncurry } x \ a \ b = \text{exp } x \ (\text{exp } a \ b)$

```

    by blast
  thus ?thesis
    using assms by fastforce
qed
qed
qed

```

Internal currying and uncurrying are the components of natural isomorphisms between the contravariant functors  $Exp^{\leftarrow} (- \otimes b) c$  and  $Exp^{\leftarrow} - (exp b c)$ .

```

lemma uncurry-naturality:
assumes ide b and ide c and Cop.arr f
shows uncurry (Cop.cod f) b c · Exp← f (exp b c) =
  Curry[exp (Cop.dom f) (exp b c), Cop.cod f ⊗ b, c]
  (eval (Cop.dom f ⊗ b) c · (uncurry (Cop.dom f) b c ⊗ f ⊗ b))
and Exp← (f ⊗ b) c · uncurry (Cop.dom f) b c =
  Curry[exp (Cop.dom f) (exp b c), Cop.cod f ⊗ b, c]
  (eval (Cop.dom f ⊗ b) c · (uncurry (Cop.dom f) b c ⊗ f ⊗ b))
and uncurry (Cop.cod f) b c · Exp← f (exp b c) =
  Exp← (f ⊗ b) c · uncurry (Cop.dom f) b c
proof –
interpret xb: functor Cop.comp Cop.comp ⟨λx. x ⊗ b⟩
using assms(1) T.fixing-ide-gives-functor-2 [of b]
by (simp add: category-axioms dual-category.intro dual-functor.intro
  dual-functor.is-functor)
interpret F: functor Cop.comp C ⟨λx. Exp← x (exp b c)⟩
using assms functor-cnt-Exp by blast
have *: ∧x. Cop.ide x ⇒
  Uncurry (x ⊗ b) c (uncurry x b c) =
  eval b c · (eval x (exp b c) ⊗ b) · a-1[exp x (exp b c), x, b]
using assms Uncurry-Curry Cop.ide-char by auto
show 1: uncurry (Cop.cod f) b c · cnt-Exp f (exp b c) =
  Curry[exp (Cop.dom f) (exp b c), Cop.cod f ⊗ b, c]
  (eval (Cop.dom f ⊗ b) c · (uncurry (Cop.dom f) b c ⊗ f ⊗ b))
proof –
have uncurry (Cop.cod f) b c · cnt-Exp f (exp b c) =
  Curry[exp (Cop.dom f) (exp b c), Cop.cod f ⊗ b, c]
  ((eval b c ·
    (eval (Cop.cod f) (exp b c) ⊗ b) ·
    a-1[exp (Cop.cod f) (exp b c), (Cop.cod f), b]) ·
    (cnt-Exp f (exp b c) ⊗ Cop.cod f ⊗ b))
using assms ide-exp cnt-Exp-in-hom comp-Curry-arr by auto
also have ... = Curry[exp (Cop.dom f) (exp b c), Cop.cod f ⊗ b, c]
  ((eval b c ·
    (eval (Cop.cod f) (exp b c) ⊗ b) ·
    ((cnt-Exp f (exp b c) ⊗ Cop.cod f) ⊗ b)) ·
    a-1[exp (Cop.dom f) (exp b c), Cop.cod f, b])
using assms comp-assoc
  assoc'-naturality [of cnt-Exp f (exp b c) Cop.cod f b]

```

**by auto**  
**also have** ... =  $\text{Curry}[\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c), \text{Cop.cod } f \otimes b, c]$   
 $(\text{Uncurry}[b, c]$   
 $(\text{Uncurry}[\text{Cop.cod } f, \text{exp } b \ c]$   
 $(\text{Curry}[\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c), \text{Cop.cod } f, \text{exp } b \ c]$   
 $(\text{eval } (\text{Cop.dom } f) (\text{exp } b \ c) \cdot$   
 $(\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c) \otimes f)))) \cdot$   
 $a^{-1}[\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c), \text{Cop.cod } f, b])$   
**using** *assms interchange* **by simp**  
**also have** ... =  $\text{Curry}[\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c), \text{Cop.cod } f \otimes b, c]$   
 $(\text{eval } b \ c \cdot$   
 $((\text{eval } (\text{Cop.dom } f) (\text{exp } b \ c) \cdot$   
 $(\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c) \otimes f)) \otimes b) \cdot$   
 $a^{-1}[\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c), \text{Cop.cod } f, b])$   
**using** *assms Uncurry-Curry comp-assoc* **by force**  
**also have** ... =  $\text{Curry}[\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c), \text{Cop.cod } f \otimes b, c]$   
 $(\text{eval } b \ c \cdot$   
 $((\text{eval } (\text{Cop.dom } f) (\text{exp } b \ c) \otimes b) \cdot$   
 $((\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c) \otimes f) \otimes b)) \cdot$   
 $a^{-1}[\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c), \text{Cop.cod } f, b])$   
**using** *assms interchange* **by simp**  
**also have** ... =  $\text{Curry}[\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c), \text{Cop.cod } f \otimes b, c]$   
 $((\text{eval } b \ c \cdot (\text{eval } (\text{Cop.dom } f) (\text{exp } b \ c) \otimes b) \cdot$   
 $a^{-1}[\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c), \text{cod } f, b]) \cdot$   
 $(\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c) \otimes f \otimes b))$   
**using** *assms assoc'-naturality [of exp (Cop.dom f) (exp b c) f b] comp-assoc*  
**by simp**  
**also have** ... =  $\text{Curry}[\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c), \text{Cop.cod } f \otimes b, c]$   
 $(\text{Uncurry}[\text{Cop.dom } f \otimes b, c]$   
 $(\text{uncurry } (\text{Cop.dom } f) b \ c) \cdot$   
 $(\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c) \otimes f \otimes b))$   
**using** *assms \** **by simp**  
**also have** ... =  
 $\text{Curry } (\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c)) (\text{Cop.cod } f \otimes b) \ c$   
 $(\text{eval } (\text{Cop.dom } f \otimes b) \ c \cdot$   
 $(\text{uncurry } (\text{Cop.dom } f) b \ c \otimes (\text{Cop.dom } f \otimes b) \cdot (f \otimes b)))$   
**using** *assms ide-exp internal-curry(2) interchange comp-assoc*  
*comp-arr-dom [of uncurry (Cop.dom f) b c]*  
**by auto**  
**also have** ... =  $\text{Curry}[\text{exp } (\text{Cop.dom } f) (\text{exp } b \ c), \text{Cop.cod } f \otimes b, c]$   
 $(\text{eval } (\text{Cop.dom } f \otimes b) \ c \cdot$   
 $(\text{uncurry } (\text{Cop.dom } f) b \ c \otimes f \otimes b))$   
**using** *assms(1,3) comp-cod-arr interchange* **by fastforce**  
**finally show** *?thesis* **by blast**  
**qed**  
**show** 2:  $\text{Exp}^{\leftarrow} (f \otimes b) \ c \cdot \text{uncurry } (\text{Cop.dom } f) \ b \ c = \dots$   
**proof** –  
**have**  $\text{Exp}^{\leftarrow} (f \otimes b) \ c \cdot \text{uncurry } (\text{Cop.dom } f) \ b \ c =$   
 $\text{Curry}[\text{exp } (\text{Cop.dom } f \otimes b) \ c, \text{Cop.cod } f \otimes b, c]$

$(\text{eval } (\text{Cop.dom } f \otimes b) \ c \cdot (\text{exp } (\text{Cop.dom } f \otimes b) \ c \otimes f \otimes b)) \cdot$   
 $\text{uncurry } (\text{Cop.dom } f) \ b \ c$   
**using** *assms comp-arr-dom* **by** *simp*  
**also have**  $\dots = \text{Curry}[\text{exp } (\text{Cop.dom } f) \ (\text{exp } \ b \ c), \ \text{Cop.cod } f \otimes b, \ c]$   
 $((\text{eval } (\text{Cop.dom } f \otimes b) \ c \cdot$   
 $(\text{exp } (\text{Cop.dom } f \otimes b) \ c \otimes f \otimes b)) \cdot$   
 $(\text{uncurry } (\text{Cop.dom } f) \ b \ c \otimes \text{Cop.cod } f \otimes b))$   
**using** *assms Curry-in-hom comp-Curry-arr* **by** *force*  
**also have**  $\dots = \text{Curry}[\text{exp } (\text{Cop.dom } f) \ (\text{exp } \ b \ c), \ \text{Cop.cod } f \otimes b, \ c]$   
 $(\text{eval } (\text{Cop.dom } f \otimes b) \ c \cdot$   
 $(\text{exp } (\text{Cop.dom } f \otimes b) \ c \cdot \text{uncurry } (\text{Cop.dom } f) \ b \ c$   
 $\otimes (f \otimes b) \cdot (\text{Cop.cod } f \otimes b)))$   
**proof** –  
**have**  $\text{seq } (\text{exp } (\text{Cop.dom } f \otimes b) \ c) \ (\text{uncurry } (\text{Cop.dom } f) \ b \ c)$   
**using** *assms* **by** *fastforce*  
**thus** *?thesis*  
**using** *assms internal-curry comp-assoc interchange* **by** *simp*  
**qed**  
**also have**  $\dots = \text{Curry}[\text{exp } (\text{Cop.dom } f) \ (\text{exp } \ b \ c), \ \text{Cop.cod } f \otimes b, \ c]$   
 $(\text{eval } (\text{Cop.dom } f \otimes b) \ c \cdot$   
 $(\text{uncurry } (\text{Cop.dom } f) \ b \ c \otimes f \otimes b))$   
**proof** –  
**have**  $(f \otimes b) \cdot (\text{Cop.cod } f \otimes b) = f \otimes b$   
**using** *assms interchange comp-arr-dom comp-cod-arr* **by** *simp*  
**thus** *?thesis*  
**using** *assms internal-curry comp-cod-arr* [of *uncurry (Cop.dom f) b c*]  
**by** *simp*  
**qed**  
**finally show** *?thesis* **by** *simp*  
**qed**  
**show**  $\text{uncurry } (\text{Cop.cod } f) \ b \ c \cdot \text{Exp}^{\leftarrow} f \ (\text{exp } \ b \ c) =$   
 $\text{Exp}^{\leftarrow} (f \otimes b) \ c \cdot \text{uncurry } (\text{Cop.dom } f) \ b \ c$   
**using** 1 2 **by** *simp*  
**qed**

**lemma** *natural-isomorphism-uncurry*:  
**assumes** *ide b* **and** *ide c*  
**shows** *natural-isomorphism Cop.comp C*  
 $(\lambda x. \text{Exp}^{\leftarrow} x \ (\text{exp } \ b \ c)) \ (\lambda x. \text{Exp}^{\leftarrow} (x \otimes b) \ c)$   
 $(\lambda f. \text{uncurry } (\text{Cop.cod } f) \ b \ c \cdot \text{Exp}^{\leftarrow} f \ (\text{exp } \ b \ c))$   
**proof** –  
**interpret** *xb: functor Cop.comp Cop.comp*  $\langle \lambda x. x \otimes b \rangle$   
**using** *assms(1) T.fixing-ide-gives-functor-2*  
**by** (*simp add: category-axioms dual-category.intro dual-functor.intro*  
*dual-functor.is-functor*)  
**interpret** *Exp-c: functor Cop.comp C*  $\langle \lambda x. \text{Exp}^{\leftarrow} x \ c \rangle$   
**using** *assms functor-cnt-Exp* **by** *blast*  
**interpret** *F: functor Cop.comp C*  $\langle \lambda x. \text{Exp}^{\leftarrow} x \ (\text{exp } \ b \ c) \rangle$   
**using** *assms functor-cnt-Exp* **by** *blast*

```

interpret G: functor Cop.comp C ⟨λx. Exp← (x ⊗ b) c⟩
proof –
  interpret G: composite-functor Cop.comp Cop.comp C
    ⟨λx. x ⊗ b⟩ ⟨λy. Exp← y c⟩
  ..
  have G.map = (λx. Exp← (x ⊗ b) c)
  by auto
  thus functor Cop.comp C (λx. Exp← (x ⊗ b) c)
  using G.functor-axioms by metis
qed
interpret φ: transformation-by-components Cop.comp C
  ⟨λx. Exp← x (exp b c)⟩ ⟨λx. Exp← (x ⊗ b) c⟩
  ⟨λx. uncurry x b c⟩
proof
  show ∧a. Cop.ide a ⇒
    «uncurry a b c : Exp← a (exp b c) → Exp← (a ⊗ b) c»
  using assms internal-curry(2) Cop.ide-char cnt-Exp-ide by auto
  show ∧f. Cop.arr f ⇒
    uncurry (Cop.cod f) b c · Exp← f (exp b c) =
    Exp← (f ⊗ b) c · uncurry (Cop.dom f) b c
  using assms uncurry-naturality by simp
qed
have natural-isomorphism Cop.comp C
  (λx. Exp← x (exp b c)) (λx. Exp← (x ⊗ b) c) φ.map
proof
  fix a
  assume a: Cop.ide a
  show iso (φ.map a)
  using a assms internal-curry [of a b c] φ.map-simp-ide
    inverse-arrows-sym
  by auto
qed
moreover have φ.map = (λf. uncurry (Cop.cod f) b c · Exp← f (exp b c))
  using assms φ.map-def by auto
ultimately show ?thesis
  unfolding φ.map-def by simp
qed

lemma natural-isomorphism-curry:
assumes ide b and ide c
shows natural-isomorphism Cop.comp C
  (λx. Exp← (x ⊗ b) c) (λx. Exp← x (exp b c))
  (λf. curry (Cop.cod f) b c · Exp← (f ⊗ b) c)
proof –
  interpret φ: natural-isomorphism Cop.comp C
  ⟨λx. Exp← x (exp b c)⟩ ⟨λx. Exp← (x ⊗ b) c⟩
  ⟨λf. uncurry (Cop.cod f) b c · Exp← f (exp b c)⟩
  using assms natural-isomorphism-uncurry by blast
  interpret ψ: inverse-transformation Cop.comp C

```



```

    ⟨λx. Exp← x (exp b c)⟩ ⟨λx. Exp← (x ⊗ b) c⟩
    ⟨λf. uncurry (Cop.cod f) b c · Exp← f (exp b c)⟩
  ..
have 1: ∧a. Cop.ide a ⇒ ψ.map a = curry a b c
proof –
  fix a
  assume a: Cop.ide a
  have inverse-arrows
    (uncurry (Cop.cod a) b c · Exp← a (exp b c)) (ψ.map a)
  using assms a ψ.inverts-components by blast
  moreover
  have inverse-arrows
    (uncurry (Cop.cod a) b c · Exp← a (exp b c)) (curry a b c)
  by (metis assms a Cop.ideD(1,3) Cop.ide-char φ.F.preserves-ide
    φ.preserves-reflects-arr comp-arr-ide internal-curry(3)
    inverse-arrows-sym)
  ultimately show ψ.map a = curry a b c
  using internal-curry inverse-arrow-unique by simp
qed
have ψ.map = (λf. curry (Cop.cod f) b c · Exp← (f ⊗ b) c)
proof
  fix f
  show ψ.map f = curry (Cop.cod f) b c · Exp← (f ⊗ b) c
  using assms 1 ψ.inverts-components internal-curry(3) ψ.naturality2
    Cop.ide-char ψ.extensionality
  by auto
qed
thus ?thesis
  using ψ.natural-isomorphism-axioms by simp
qed

```

### 1.2.3 Yoneda Embedding

The internal hom provides a closed monoidal category  $C$  with a "Yoneda embedding", which is a mapping that takes each arrow  $g$  of  $C$  to a natural transformation from the contravariant functor  $Exp^{\leftarrow} - (dom\ g)$  to the contravariant functor  $Exp^{\leftarrow} - (cod\ g)$ . Note that here the target category is  $C$  itself, not a category of sets and functions as in the classical case. Note also that we are talking here about ordinary functors and natural transformations. We can easily prove from general considerations that the Yoneda embedding (so-defined) is faithful. However, to obtain a fullness result requires the development of a certain amount of enriched category theory, which we do elsewhere.

```

lemma yoneda-embedding:
assumes «g : a → b»
shows natural-transformation Cop.comp C
  (λx. Exp← x a) (λx. Exp← x b) (λx. Exp x g)
and Uncurry[a, b] (Exp a g · Curry[ $\mathcal{I}$ , a, a] l[a]) · l-1[a] = g

```

**proof** –  
**interpret** *Exp*: *binary-functor Cop.comp C C*  $\langle \lambda fg. \text{Exp } (fst\ fg) (snd\ fg) \rangle$   
**using** *functor-Exp* **by** *blast*  
**interpret** *Exp-g*: *natural-transformation Cop.comp C*  
 $\langle \lambda x. \text{Exp } x (dom\ g) \rangle \langle \lambda x. \text{Exp } x (cod\ g) \rangle \langle \lambda x. \text{Exp } x\ g \rangle$   
**using** *assms Exp.fixing-arr-gives-natural-transformation-2* [of *g*] **by** *auto*  
**show** *natural-transformation Cop.comp C*  $(\lambda x. \text{Exp}^{\leftarrow} x\ a) (\lambda x. \text{Exp}^{\leftarrow} x\ b)$   
 $(\lambda x. \text{Exp } x\ g)$   
**using** *assms Exp-x-ide Exp-x-ide Exp-g.natural-transformation-axioms*  
**by** *auto*  
**show** *Uncurry*[*a*, *b*]  $(\text{Exp } a\ g \cdot \text{Curry}[\mathcal{I}, a, a] \ 1[a]) \cdot 1^{-1}[a] = g$   
**proof** –  
**have** *Uncurry*[*a*, *b*]  $(\text{Exp } a\ g \cdot \text{Curry}[\mathcal{I}, a, a] \ 1[a]) \cdot 1^{-1}[a] =$   
 $(eval\ a\ b \cdot (\text{Exp } a\ g \otimes a) \cdot (\text{Curry}[\mathcal{I}, a, a] \ 1[a] \otimes a)) \cdot 1^{-1}[a]$   
**using** *assms Exp-ide lunit-in-hom*  
*interchange* [of *Exp a g Curry*[\mathcal{I}, a, a] 1[a] a a]  
**by** *auto*  
**also have** ... =  $g \cdot (eval\ a\ a \cdot (\text{Curry}[\mathcal{I}, a, a] \ 1[a] \otimes a)) \cdot 1^{-1}[a]$   
**using** *assms Uncurry-Exp-dom comp-assoc* **by** (*metis in-homE*)  
**also have** ... =  $g \cdot 1[a] \cdot 1^{-1}[a]$   
**using** *assms Uncurry-Curry ide-dom ide-unity lunit-in-hom* **by** *auto*  
**also have** ... =  $g$   
**using** *assms comp-arr-dom* **by** *force*  
**finally show** *?thesis*  
**by** *blast*  
**qed**  
**qed**

**lemma** *yoneda-embedding-is-faithful*:  
**assumes** *par g g'* **and**  $(\lambda x. \text{Exp } x\ g) = (\lambda x. \text{Exp } x\ g')$   
**shows**  $g = g'$   
**proof** –  
**have**  $g \cdot eval\ (dom\ g)\ (dom\ g) = g' \cdot eval\ (dom\ g)\ (dom\ g)$   
**using** *assms Uncurry-Exp-dom* **by** *metis*  
**thus**  $g = g'$   
**using** *assms retraction-eval-ide-self retraction-is-epi*  
**by** (*metis epi-cancel eval-simps(1,3) ide-dom seqI*)  
**qed**

The following is a version of the key fact underlying the classical Yoneda Lemma: for any natural transformation  $\tau$  from  $\text{Exp}^{\leftarrow} - a$  to  $\text{Exp}^{\leftarrow} - b$ , there is a fixed arrow  $g : a \rightarrow b$ , depending only on the single component  $\tau a$ , such that the compositions  $\tau x \cdot e$  of an arbitrary component  $\tau x$  with arbitrary global elements  $e : \mathcal{I} \rightarrow \text{exp } x\ a$  depend on  $\tau$  only via  $g$ , and hence only via  $\tau a$ .

**lemma** *hom-transformation-expansion*:  
**assumes** *natural-transformation*  
 $Cop.comp\ C\ (\lambda x. \text{Exp}^{\leftarrow} x\ a)\ (\lambda x. \text{Exp}^{\leftarrow} x\ b)\ \tau$   
**and** *ide a* **and** *ide b*

**shows**  $\llbracket \text{Uncurry}[a, b] (\tau a \cdot \text{Curry}[\mathcal{I}, a, a] \text{ l}[a]) \cdot \text{l}^{-1}[a] : a \rightarrow b \rrbracket$   
**and**  $\bigwedge x e. \llbracket \text{ide } x; \llbracket e : \mathcal{I} \rightarrow \text{exp } x a \rrbracket \rrbracket \implies$   
 $\tau x \cdot e = \text{Exp } x (\text{Uncurry}[a, b] (\tau a \cdot \text{Curry}[\mathcal{I}, a, a] \text{ l}[a]) \cdot \text{l}^{-1}[a]) \cdot e$

**proof** –

**interpret**  $\tau$ : *natural-transformation Cop.comp C*  
 $\langle \lambda x. \text{Exp}^{\leftarrow} x a \rangle \langle \lambda x. \text{Exp}^{\leftarrow} x b \rangle \tau$

**using** *assms by blast*

**let**  $?Id\text{-}a = \text{Curry}[\mathcal{I}, a, a] \text{ l}[a]$

**have**  $Id\text{-}a$ :  $\llbracket ?Id\text{-}a : \mathcal{I} \rightarrow \text{exp } a a \rrbracket$

**using** *assms ide-unity by blast*

**let**  $?g = \text{Uncurry}[a, b] (\tau a \cdot ?Id\text{-}a) \cdot \text{l}^{-1}[a]$

**show**  $g$ :  $\llbracket ?g : a \rightarrow b \rrbracket$

**using** *assms(2-3) Id-a cnt-Exp-ide by auto*

**have**  $*$ :  $\bigwedge x e. \llbracket \text{ide } x; \llbracket e : \mathcal{I} \rightarrow \text{exp } x a \rrbracket \rrbracket$   
 $\implies \tau x \cdot e = \text{Curry}[\text{exp } x a, x, b] (?g \cdot \text{eval } x a) \cdot e$

**proof** –

**fix**  $x e$

**assume**  $x$ : *ide x*

**assume**  $e$ :  $\llbracket e : \mathcal{I} \rightarrow \text{exp } x a \rrbracket$

**let**  $?e' = \text{Uncurry } x a e \cdot \text{l}^{-1}[x]$

**have**  $e'$ :  $\llbracket ?e' : x \rightarrow a \rrbracket$

**using** *assms(2) x e by blast*

**have**  $1$ :  $e = \text{Exp}^{\leftarrow} ?e' a \cdot ?Id\text{-}a$

**proof** –

**have**  $\text{Exp}^{\leftarrow} ?e' a \cdot ?Id\text{-}a =$   
 $\text{Curry}[\text{exp } a a, x, a] (\text{eval } a a \cdot (\text{exp } a a \otimes ?e')) \cdot ?Id\text{-}a$

**using** *assms(2) e' by auto*

**also have**  $\dots =$   
 $\text{Curry}[\mathcal{I}, x, a] (\text{eval } a a \cdot (\text{exp } a a \otimes ?e') \cdot (?Id\text{-}a \otimes x))$

**using** *assms(2) Id-a e' x comp-Curry-arr comp-assoc by auto*

**also have**  $\dots = \text{Curry}[\mathcal{I}, x, a] (\text{eval } a a \cdot (?Id\text{-}a \otimes ?e'))$

**using** *assms(2) e' Id-a interchange comp-arr-dom comp-cod-arr in-homE*

**by** (*metis (no-types, lifting)*)

**also have**  $\dots = \text{Curry } \mathcal{I} x a (\text{eval } a a \cdot (?Id\text{-}a \otimes a) \cdot (\mathcal{I} \otimes ?e'))$

**using** *assms(2) interchange*

**by** (*metis (no-types, lifting) e' Id-a comp-arr-ide comp-cod-arr ide-char ide-unity in-homE seqI*)

**also have**  $\dots =$   
 $\text{Curry}[\mathcal{I}, x, a] (\text{Uncurry } a a (\text{Curry}[\mathcal{I}, a, a] \text{ l}[a]) \cdot (\mathcal{I} \otimes ?e'))$

**using** *comp-assoc by simp*

**also have**  $\dots = \text{Curry}[\mathcal{I}, x, a] (\text{l}[a] \cdot (\mathcal{I} \otimes ?e'))$

**using** *assms(2) Uncurry-Curry comp-assoc ide-unity lunit-in-hom*

**by** *presburger*

**also have**  $\dots = \text{Curry}[\mathcal{I}, x, a] (?e' \cdot \text{l}[x])$

**using** *assms(2) e' in-homE lunit-naturality*

**by** (*metis (no-types, lifting)*)

**also have**  $\dots = \text{Curry}[\mathcal{I}, x, a] (\text{Uncurry}[x, a] e \cdot \text{l}^{-1}[x] \cdot \text{l}[x])$

**using** *comp-assoc by simp*

**also have**  $\dots = \text{Curry}[\mathcal{I}, x, a] (\text{Uncurry}[x, a] e)$

using *assms(2) x e comp-arr-dom Uncurry-simps(2)* by *force*  
 also have ... = *e*  
 using *assms(2) x e Curry-Uncurry ide-unity* by *blast*  
 finally show *?thesis* by *simp*  
 qed  
 have  $\tau x \cdot e = \tau x \cdot \text{Exp}^{\leftarrow} ?e' a \cdot ?Id-a$   
 using *1* by *simp*  
 also have ... =  $(\tau x \cdot \text{Exp}^{\leftarrow} ?e' a) \cdot ?Id-a$   
 using *comp-assoc* by *simp*  
 also have ... =  $(\text{Exp}^{\leftarrow} ?e' b \cdot \tau a) \cdot ?Id-a$   
 using *e'  $\tau$ .naturality [of ?e']* by *auto*  
 also have ... =  $\text{Curry}[exp a b, x, b] (eval a b \cdot (exp a b \otimes ?e')) \cdot \tau a \cdot ?Id-a$   
 using *assms(2) e' comp-assoc* by *auto*  
 also have ... =  
 $\text{Curry}[\mathcal{I}, x, b] ((eval a b \cdot (exp a b \otimes ?e')) \cdot (\tau a \cdot ?Id-a \otimes x))$   
 proof –  
 have  $\langle \tau a \cdot ?Id-a : \mathcal{I} \rightarrow exp a b \rangle$   
 using *Id-a assms(2-3) in-homI cnt-Exp-ide*  
 by *(intro comp-in-homI) auto*  
 moreover have  $\langle eval a b \cdot (exp a b \otimes ?e') : exp a b \otimes x \rightarrow b \rangle$   
 using *assms(2-3) e' ide-in-hom* by *blast*  
 ultimately show *?thesis*  
 using *x comp-Curry-arr* by *blast*  
 qed  
 also have ... =  $\text{Curry}[\mathcal{I}, x, b] (eval a b \cdot (exp a b \otimes ?e') \cdot (\tau a \cdot ?Id-a \otimes x))$   
 using *comp-assoc* by *simp*  
 also have ... =  $\text{Curry}[\mathcal{I}, x, b] (eval a b \cdot (exp a b \cdot \tau a \cdot ?Id-a \otimes ?e' \cdot x))$   
 proof –  
 have  $seq (exp a b) (\tau a \cdot \text{Curry}[\mathcal{I}, a, a] 1[a])$   
 using *assms ide-exp  $\tau$ .natural-transformation-axioms Id-a Curry-Uncurry ide-exp ide-in-hom*  
 by *auto*  
 moreover have  $seq (\text{Uncurry}[x, a] e \cdot 1^{-1}[x]) x$   
 using *x e' by auto*  
 ultimately show *?thesis*  
 using *assms interchange* by *simp*  
 qed  
 also have ... =  $\text{Curry}[\mathcal{I}, x, b] (eval a b \cdot (\tau a \cdot ?Id-a \otimes ?e'))$   
 proof –  
 have  $exp a b \cdot \tau a \cdot ?Id-a = \tau a \cdot ?Id-a$   
 using *assms(2-3) e' ide-exp comp-ide-arr  $\tau$ .preserves-hom cnt-Exp-ide Id-a*  
 by *auto*  
 moreover have  $?e' \cdot x = ?e'$   
 using *e' comp-arr-dom* by *blast*  
 ultimately show *?thesis*  
 using *interchange* by *simp*  
 qed  
 also have ... =  $\text{Curry}[\mathcal{I}, x, b] (eval a b \cdot (\tau a \cdot ?Id-a \otimes a) \cdot (\mathcal{I} \otimes ?e'))$

**proof** –  
**have**  $(\tau a \cdot ?Id-a) \cdot \mathcal{I} = \tau a \cdot ?Id-a$   
**using** *assms(2) comp-arr-ide*  
**by** (*metis Id-a comp-arr-dom in-homE comp-assoc*)  
**moreover have**  $a \cdot ?e' = ?e'$   
**using** *e' comp-cod-arr by blast*  
**moreover have**  $seq (\tau a \cdot Curry[\mathcal{I}, a, a] \ 1[a]) \ \mathcal{I}$   
**using** *assms(2) cnt-Exp-ide Id-a by auto*  
**moreover have**  $seq a \ (Uncurry[x, a] \ e \cdot 1^{-1}[x])$   
**using** *calculation(2) e' by auto*  
**ultimately show** *?thesis*  
**using** *interchange [of  $\tau a \cdot ?Id-a \ \mathcal{I} \ a \ ?e'$ ] by simp*  
**qed**  
**also have**  $\dots = Curry[\mathcal{I}, x, b] \ (eval \ a \ b \cdot (\tau a \cdot ?Id-a \otimes a) \cdot (1^{-1}[a] \cdot 1[a]) \cdot (\mathcal{I} \otimes eval \ x \ a \cdot (e \otimes x) \cdot 1^{-1}[x]))$   
**proof** –  
**have**  $(\mathcal{I} \otimes eval \ x \ a) \cdot (\mathcal{I} \otimes (e \otimes x) \cdot 1^{-1}[x]) =$   
 $(\mathcal{I} \otimes a) \cdot (\mathcal{I} \otimes eval \ x \ a) \cdot (\mathcal{I} \otimes (e \otimes x) \cdot 1^{-1}[x])$   
**using** *assms e' L.as-nat-trans.naturality2 comp-lunit-lunit'(2) comp-assoc*  
**by** (*metis (no-types, lifting) L.as-nat-trans.preserves-comp-2 in-homE*)  
**thus** *?thesis*  
**using** *assms e' comp-assoc*  
**by** (*elim in-homE*) *auto*  
**qed**  
**also have**  $\dots = Curry[\mathcal{I}, x, b] \ (?g \cdot 1[a] \cdot (\mathcal{I} \otimes eval \ x \ a \cdot (e \otimes x) \cdot 1^{-1}[x]))$   
**using** *comp-assoc by simp*  
**also have**  $\dots = Curry[\mathcal{I}, x, b] \ (?g \cdot (eval \ x \ a \cdot (e \otimes x) \cdot 1^{-1}[x]) \cdot 1[x])$   
**using** *lunit-naturality*  
**by** (*metis (no-types, lifting) e' in-homE comp-assoc*)  
**also have**  $\dots = Curry[\mathcal{I}, x, b] \ (?g \cdot eval \ x \ a \cdot (e \otimes x) \cdot 1^{-1}[x] \cdot 1[x])$   
**using** *comp-assoc by simp*  
**also have**  $\dots = Curry[\mathcal{I}, x, b] \ (?g \cdot eval \ x \ a \cdot (e \otimes x))$   
**using** *x comp-arr-dom e interchange by fastforce*  
**also have**  $\dots = Curry[\mathcal{I}, x, b] \ ((?g \cdot eval \ x \ a) \cdot (e \otimes x))$   
**using** *comp-assoc by simp*  
**also have**  $\dots = Curry[exp \ x \ a, x, b] \ (?g \cdot eval \ x \ a) \cdot e$   
**using** *assms(2) x e g comp-Curry-arr by auto*  
**finally show**  $\tau \ x \cdot e = Curry[exp \ x \ a, x, b] \ (?g \cdot eval \ x \ a) \cdot e$   
**by** *blast*  
**qed**  
**show**  $\bigwedge x \ e. \llbracket ide \ x; \langle e : \mathcal{I} \rightarrow exp \ x \ a \rangle \rrbracket \implies \tau \ x \cdot e = Exp \ x \ ?g \cdot e$   
**proof** –  
**fix**  $x \ e$   
**assume**  $x: ide \ x$   
**assume**  $e: \langle e : \mathcal{I} \rightarrow exp \ x \ a \rangle$   
**have**  $\tau \ x \cdot e = Curry[exp \ x \ a, x, b] \ (?g \cdot eval \ x \ a) \cdot e$   
**using** *x e \*  $\tau$ .natural-transformation-axioms by blast*  
**also have**  $\dots = (Curry[exp \ x \ a, x, cod \ ?g] \ (?g \cdot eval \ x \ a) \cdot Curry[exp \ x \ a, x, a] \ (Uncurry[x, a] \ (exp \ x \ a))) \cdot e$

```

proof –
  have  $\text{Curry}[\text{exp } x \ a, x, a] (\text{Uncurry}[x, a] (\text{exp } x \ a)) = \text{exp } x \ a$ 
    using assms(2) x Curry-Uncurry ide-exp ide-in-hom by force
  thus ?thesis
    using g e comp-cod-arr comp-assoc by fastforce
qed
also have  $\dots = \text{Exp } x \ ?g \cdot e$ 
  using x Exp-def cod-comp g by auto
finally show  $\tau \ x \cdot e = \text{Exp } x \ ?g \cdot e$  by blast
qed
qed

```

### 1.3 Enriched Structure

In this section we do the main work involved in showing that a closed monoidal category is “enriched in itself”. For this, we need to define, for each object  $a$ , an arrow  $\text{Id } a : \mathcal{I} \rightarrow \text{exp } a \ a$  to serve as the “identity at  $a$ ”, and for every three objects  $a$ ,  $b$ , and  $c$ , a “compositor”  $\text{Comp } a \ b \ c : \text{exp } b \ c \otimes \text{exp } a \ b \rightarrow \text{exp } a \ c$ . We also need to prove that these satisfy the appropriate unit and associativity laws. Although essentially all the work is done here, the statement and proof of the the final result is deferred to a separate theory *EnrichedCategory* so that a mutual dependence between that theory and the present one is not introduced.

```

interpretation elementary-monoidal-category C tensor unity lunit runit assoc
  using induces-elementary-monoidal-category by blast

```

```

definition Id
where  $\text{Id } a \equiv \text{Curry}[\mathcal{I}, a, a] \ 1[a]$ 

```

```

lemma Id-in-hom [intro]:
assumes ide a
shows  $\langle \text{Id } a : \mathcal{I} \rightarrow \text{exp } a \ a \rangle$ 
  unfolding Id-def
  using assms Curry-in-hom lunit-in-hom by simp

```

```

lemma Id-simps [simp]:
assumes ide a
shows arr (Id a)
and  $\text{dom } (\text{Id } a) = \mathcal{I}$ 
and  $\text{cod } (\text{Id } a) = \text{exp } a \ a$ 
  using assms Id-in-hom by blast+

```

The next definition follows Kelly [1], section 1.6.

```

definition Comp
where  $\text{Comp } a \ b \ c \equiv$ 
   $\text{Curry}[\text{exp } b \ c \otimes \text{exp } a \ b, a, c]$ 
   $(\text{eval } b \ c \cdot (\text{exp } b \ c \otimes \text{eval } a \ b) \cdot a[\text{exp } b \ c, \text{exp } a \ b, a])$ 

```

**lemma** *Comp-in-hom* [*intro*]:  
**assumes** *ide a and ide b and ide c*  
**shows**  $\langle\langle \text{Comp } a \ b \ c : \text{exp } b \ c \otimes \text{exp } a \ b \rightarrow \text{exp } a \ c \rangle\rangle$   
**using** *assms ide-exp ide-in-hom Comp-def Curry-in-hom tensor-preserves-ide*  
**by** *auto*

**lemma** *Comp-simps* [*simp*]:  
**assumes** *ide a and ide b and ide c*  
**shows** *arr (Comp a b c)*  
**and** *dom (Comp a b c) = exp b c  $\otimes$  exp a b*  
**and** *cod (Comp a b c) = exp a c*  
**using** *assms Comp-in-hom in-homE* **by** *blast+*

**lemma** *Comp-unit-right*:  
**assumes** *ide a and ide b and ide c*  
**shows**  $\langle\langle \text{Comp } a \ a \ b \cdot (\text{exp } a \ b \otimes \text{Id } a) : \text{exp } a \ b \otimes \mathcal{I} \rightarrow \text{exp } a \ b \rangle\rangle$   
**and** *Comp a a b  $\cdot$  (exp a b  $\otimes$  Id a) = r[exp a b]*  
**proof** –  
**show** *0*:  $\langle\langle \text{Comp } a \ a \ b \cdot (\text{exp } a \ b \otimes \text{Id } a) : \text{exp } a \ b \otimes \mathcal{I} \rightarrow \text{exp } a \ b \rangle\rangle$   
**using** *assms Id-in-hom tensor-in-hom ide-in-hom ide-exp* **by** *force*  
**show** *Comp a a b  $\cdot$  (exp a b  $\otimes$  Id a) = r[exp a b]*  
**proof** (*intro runit-eqI*)  
**show** *1*:  $\langle\langle \text{Comp } a \ a \ b \cdot (\text{exp } a \ b \otimes \text{Id } a) : \text{exp } a \ b \otimes \mathcal{I} \rightarrow \text{exp } a \ b \rangle\rangle$   
**by** *fact*  
**show** *Comp a a b  $\cdot$  (exp a b  $\otimes$  Id a)  $\otimes$   $\mathcal{I}$  = (exp a b  $\otimes$   $\iota$ )  $\cdot$  a[exp a b,  $\mathcal{I}$ ,  $\mathcal{I}$ ]*  
**proof** –  
**have** *r[exp a b]  $\cdot$  (Comp a a b  $\cdot$  (exp a b  $\otimes$  Id a)  $\otimes$   $\mathcal{I}$ )  $\cdot$*   
*inv a[exp a b,  $\mathcal{I}$ ,  $\mathcal{I}$ ] =*  
*r[exp a b]  $\cdot$  ((Comp a a b  $\otimes$   $\mathcal{I}$ )  $\cdot$  ((exp a b  $\otimes$  Id a)  $\otimes$   $\mathcal{I}$ ))  $\cdot$*   
*inv a[exp a b,  $\mathcal{I}$ ,  $\mathcal{I}$ ]*  
**using**  $\langle\langle \text{Comp } a \ a \ b \cdot (\text{exp } a \ b \otimes \text{Id } a) : \text{exp } a \ b \otimes \mathcal{I} \rightarrow \text{exp } a \ b \rangle\rangle$  *arrI*  
**by** *force*  
**also have** *... = (r[exp a b]  $\cdot$  (Comp a a b  $\otimes$   $\mathcal{I}$ ))  $\cdot$*   
*((exp a b  $\otimes$  Id a)  $\otimes$   $\mathcal{I}$ )  $\cdot$  inv a[exp a b,  $\mathcal{I}$ ,  $\mathcal{I}$ ]*  
**using** *comp-assoc* **by** *simp*  
**also have** *... = (Comp a a b  $\cdot$  r[exp a b  $\otimes$  exp a a])  $\cdot$*   
*((exp a b  $\otimes$  Id a)  $\otimes$   $\mathcal{I}$ )  $\cdot$  inv a[exp a b,  $\mathcal{I}$ ,  $\mathcal{I}$ ]*  
**using** *assms runit-naturality*  
**by** (*metis Comp-simps(1–2) 1 cod-comp in-homE*)  
**also have** *... = Comp a a b  $\cdot$*   
*(r[exp a b  $\otimes$  exp a a]  $\cdot$  ((exp a b  $\otimes$  Id a)  $\otimes$   $\mathcal{I}$ ))  $\cdot$*   
*inv a[exp a b,  $\mathcal{I}$ ,  $\mathcal{I}$ ]*  
**using** *comp-assoc* **by** *simp*  
**also have** *... = Comp a a b  $\cdot$  ((exp a b  $\otimes$  Id a)  $\cdot$  r[exp a b  $\otimes$   $\mathcal{I}$ ])  $\cdot$*   
*inv a[exp a b,  $\mathcal{I}$ ,  $\mathcal{I}$ ]*  
**using** *assms 1 runit-naturality*  
**by** (*metis calculation in-homE comp-assoc*)  
**also have** *... = Comp a a b  $\cdot$  (exp a b  $\otimes$  Id a)  $\cdot$  r[exp a b  $\otimes$   $\mathcal{I}$ ]  $\cdot$*

$inv\ a[exp\ a\ b,\ \mathcal{I},\ \mathcal{I}]$

**using** *comp-assoc* **by** *simp*

**also have**  $\dots = Comp\ a\ a\ b \cdot (exp\ a\ b \otimes Id\ a) \cdot (exp\ a\ b \otimes \iota)$

**using** *assms ide-unity runit-tensor' ide-exp runit-eqI unit-in-hom-ax unit-triangle(1)*

**by** *presburger*

**also have**  $\dots = (Curry[exp\ a\ b \otimes exp\ a\ a,\ a,\ b]$   
 $(eval\ a\ b \cdot (exp\ a\ b \otimes eval\ a\ a) \cdot a[exp\ a\ b,\ exp\ a\ a,\ a]) \cdot$   
 $(exp\ a\ b \otimes Id\ a)) \cdot (exp\ a\ b \otimes \iota)$

**using** *Comp-def comp-assoc* **by** *simp*

**also have**  $\dots = Curry[exp\ a\ b \otimes \mathcal{I},\ a,\ b]$   
 $((eval\ a\ b \cdot (exp\ a\ b \otimes eval\ a\ a) \cdot a[exp\ a\ b,\ exp\ a\ a,\ a]) \cdot$   
 $((exp\ a\ b \otimes Id\ a) \otimes a)) \cdot$   
 $(exp\ a\ b \otimes \iota)$

**proof** –

**have**  $\langle\langle exp\ a\ b \otimes Id\ a : exp\ a\ b \otimes \mathcal{I} \rightarrow exp\ a\ b \otimes exp\ a\ a \rangle\rangle$

**using** *assms by auto*

**moreover have**  $\langle\langle eval\ a\ b \cdot (exp\ a\ b \otimes eval\ a\ a) \cdot a[exp\ a\ b,\ exp\ a\ a,\ a] :$   
 $(exp\ a\ b \otimes exp\ a\ a) \otimes a \rightarrow b \rangle\rangle$

**using** *assms tensor-in-hom ide-in-hom ide-exp eval-in-hom<sub>ECMC</sub>*

**by** *force*

**ultimately show** *?thesis*

**using** *assms comp-Curry-arr* **by** *simp*

**qed**

**also have**  $\dots = \Gamma[exp\ a\ b] \cdot (exp\ a\ b \otimes \iota)$

**proof** –

**have** *1*:  $Uncurry[a,\ b]$   
 $(Curry[exp\ a\ b \otimes \mathcal{I},\ a,\ b]$   
 $((eval\ a\ b \cdot (exp\ a\ b \otimes eval\ a\ a) \cdot a[exp\ a\ b,\ exp\ a\ a,\ a]) \cdot$   
 $((exp\ a\ b \otimes Id\ a) \otimes a))) =$   
 $(eval\ a\ b \cdot (exp\ a\ b \otimes eval\ a\ a) \cdot a[exp\ a\ b,\ exp\ a\ a,\ a]) \cdot$   
 $((exp\ a\ b \otimes Id\ a) \otimes a)$

**proof** –

**have**  $\langle\langle (eval\ a\ b \cdot (exp\ a\ b \otimes eval\ a\ a) \cdot a[exp\ a\ b,\ exp\ a\ a,\ a]) \cdot$   
 $((exp\ a\ b \otimes Id\ a) \otimes a) : (exp\ a\ b \otimes \mathcal{I}) \otimes a \rightarrow b \rangle\rangle$

**using** *assms tensor-in-hom ide-in-hom eval-in-hom<sub>ECMC</sub> ide-exp*

**by** *force*

**thus** *?thesis*

**using** *assms Uncurry-Curry* **by** *auto*

**qed**

**also have**  $\dots = (eval\ a\ b \cdot (exp\ a\ b \otimes eval\ a\ a) \cdot (exp\ a\ b \otimes Id\ a \otimes a)) \cdot$   
 $a[exp\ a\ b,\ \mathcal{I},\ a]$

**using** *assms ide-exp comp-assoc assoc-naturality [of exp a b Id a a]*

**by** *auto*

**also have**  $\dots = (eval\ a\ b \cdot (exp\ a\ b \otimes Uncurry[a,\ a] (Id\ a))) \cdot$   
 $a[exp\ a\ b,\ \mathcal{I},\ a]$

**using** *assms interchange*

**by** (*metis (no-types, lifting) ide-exp lunit-in-hom Uncurry-Curry*  
*ide-unity comp-ide-self ideD(1) in-homE Id-def*)



**also have** ... = (eval a b · (exp a b ⊗ l[a])) · a[exp a b,  $\mathcal{I}$ , a]  
**by** (metis (no-types, lifting) assms(1) lunit-in-hom Uncurry-Curry  
ide-unity Id-def)  
**also have** 2: ... = (eval a b · (exp a b ⊗ a) · (exp a b ⊗ l[a])) ·  
a[exp a b,  $\mathcal{I}$ , a]  
**using** assms interchange l-ide-simp **by** auto  
**also have** ... = Uncurry[a, b] (exp a b) · (exp a b ⊗ l[a]) · a[exp a b,  $\mathcal{I}$ , a]  
**using** comp-assoc **by** simp  
**also have** ... = Uncurry a b r[exp a b]  
**using** assms triangle ide-exp 2 comp-assoc **by** auto  
**finally have** Uncurry[a, b]  
(Curry[exp a b ⊗  $\mathcal{I}$ , a, b]  
((eval a b · (exp a b ⊗ eval a a) ·  
a[exp a b, exp a a, a]) ·  
((exp a b ⊗ Id a) ⊗ a))) =  
Uncurry[a, b] r[exp a b]  
**by** blast  
**hence** Curry[exp a b ⊗  $\mathcal{I}$ , a, b]  
((eval a b · (exp a b ⊗ eval a a) · a[exp a b, exp a a, a]) ·  
((exp a b ⊗ Id a) ⊗ a)) =  
r[exp a b]  
**using** assms 1 Curry-Uncurry runit-in-hom **by** force  
**thus** ?thesis  
**by** presburger  
**qed**  
**finally have** r[exp a b] ·  
(Comp a a b · (exp a b ⊗ Id a) ⊗  $\mathcal{I}$ ) · inv a[exp a b,  $\mathcal{I}$ ,  $\mathcal{I}$ ] =  
r[exp a b] · (exp a b ⊗  $\iota$ )  
**by** blast  
**hence** (Comp a a b · (exp a b ⊗ Id a) ⊗  $\mathcal{I}$ ) · inv a[exp a b,  $\mathcal{I}$ ,  $\mathcal{I}$ ] =  
exp a b ⊗  $\iota$   
**using** assms ide-exp iso-cancel-left [of r[exp a b]] iso-runit **by** fastforce  
**thus** ?thesis  
**by** (metis assms(1–2) 0 R.as-nat-trans.naturality1 comp-assoc-assoc'(2)  
ide-exp ide-unity in-homE comp-assoc)  
**qed**  
**qed**  
**qed**

**lemma** Comp-unit-left:

**assumes** ide a **and** ide b **and** ide c

**shows** «Comp a b b · (Id b ⊗ exp a b) :  $\mathcal{I}$  ⊗ exp a b → exp a b»

**and** Comp a b b · (Id b ⊗ exp a b) = l[exp a b]

**proof** –

**show** 0: «Comp a b b · (Id b ⊗ exp a b) :  $\mathcal{I}$  ⊗ exp a b → exp a b»

**using** assms ide-exp **by** simp

**show** Comp a b b · (Id b ⊗ exp a b) = l[exp a b]

**proof** (intro lunit-eqI)

**show** «Comp a b b · (Id b ⊗ exp a b) :  $\mathcal{I}$  ⊗ exp a b → exp a b»

**by fact**  
**show**  $\mathcal{I} \otimes \text{Comp } a \ b \ b \cdot (\text{Id } b \otimes \text{exp } a \ b) = (\iota \otimes \text{exp } a \ b) \cdot \mathfrak{a}^{-1}[\mathcal{I}, \mathcal{I}, \text{exp } a \ b]$   
**proof** –  
**have**  $l[\text{exp } a \ b] \cdot (\mathcal{I} \otimes \text{Comp } a \ b \ b \cdot (\text{Id } b \otimes \text{exp } a \ b)) \cdot \mathfrak{a}[\mathcal{I}, \mathcal{I}, \text{exp } a \ b] =$   
 $l[\text{exp } a \ b] \cdot ((\mathcal{I} \otimes \text{Comp } a \ b \ b) \cdot (\mathcal{I} \otimes \text{Id } b \otimes \text{exp } a \ b)) \cdot \mathfrak{a}[\mathcal{I}, \mathcal{I}, \text{exp } a \ b]$   
**using** *assms 0 interchange [of  $\mathcal{I} \ \mathcal{I} \ \text{Comp } a \ b \ b \ \text{Id } b \otimes \text{exp } a \ b]$  by auto*  
**also have**  $\dots = (l[\text{exp } a \ b] \cdot (\mathcal{I} \otimes \text{Comp } a \ b \ b)) \cdot$   
 $(\mathcal{I} \otimes \text{Id } b \otimes \text{exp } a \ b) \cdot \mathfrak{a}[\mathcal{I}, \mathcal{I}, \text{exp } a \ b]$   
**using** *comp-assoc by simp*  
**also have**  $\dots = (\text{Comp } a \ b \ b \cdot l[\text{exp } b \ b \otimes \text{exp } a \ b]) \cdot (\mathcal{I} \otimes \text{Id } b \otimes \text{exp } a \ b) \cdot$   
 $\mathfrak{a}[\mathcal{I}, \mathcal{I}, \text{exp } a \ b]$   
**using** *assms lunit-naturality*  
**by** (*metis 0 Comp-simps(1-2) cod-comp in-homE*)  
**also have**  $\dots = \text{Comp } a \ b \ b \cdot$   
 $(l[\text{exp } b \ b \otimes \text{exp } a \ b] \cdot (\mathcal{I} \otimes \text{Id } b \otimes \text{exp } a \ b)) \cdot$   
 $\mathfrak{a}[\mathcal{I}, \mathcal{I}, \text{exp } a \ b]$   
**using** *comp-assoc by simp*  
**also have**  $\dots =$   
 $(\text{Comp } a \ b \ b \cdot (\text{Id } b \otimes \text{exp } a \ b)) \cdot l[\mathcal{I} \otimes \text{exp } a \ b] \cdot \mathfrak{a}[\mathcal{I}, \mathcal{I}, \text{exp } a \ b]$   
**using** *assms 0 lunit-naturality calculation in-homE comp-assoc by metis*  
**also have**  $\dots = (\text{Comp } a \ b \ b \cdot (\text{Id } b \otimes \text{exp } a \ b)) \cdot (\iota \otimes \text{exp } a \ b)$   
**using** *assms(1-2) ide-exp ide-unity lunit-eqI lunit-tensor' unit-in-hom-ax*  
*unit-triangle(2)*  
**by** *presburger*  
**also have**  $\dots = l[\text{exp } a \ b] \cdot (\iota \otimes \text{exp } a \ b)$   
**proof** (*unfold Comp-def*)  
**have** (*Curry* $[\text{exp } b \ b \otimes \text{exp } a \ b, a, b]$   
 $(\text{eval } b \ b \cdot (\text{exp } b \ b \otimes \text{eval } a \ b)) \cdot \mathfrak{a}[\text{exp } b \ b, \text{exp } a \ b, a]) \cdot$   
 $(\text{Id } b \otimes \text{exp } a \ b)) \cdot$   
 $(\iota \otimes \text{exp } a \ b) =$   
 $\text{Curry}[\mathcal{I} \otimes \text{exp } a \ b, a, b]$   
 $((\text{eval } b \ b \cdot (\text{exp } b \ b \otimes \text{eval } a \ b)) \cdot \mathfrak{a}[\text{exp } b \ b, \text{exp } a \ b, a]) \cdot$   
 $((\text{Id } b \otimes \text{exp } a \ b) \otimes a)) \cdot$   
 $(\iota \otimes \text{exp } a \ b)$   
**proof** –  
**have**  $\ll \text{eval } b \ b \cdot (\text{exp } b \ b \otimes \text{eval } a \ b) \cdot \mathfrak{a}[\text{exp } b \ b, \text{exp } a \ b, a]$   
 $: (\text{exp } b \ b \otimes \text{exp } a \ b) \otimes a \rightarrow b \gg$   
**using** *assms ide-exp tensor-in-hom ide-in-hom ide-exp eval-in-hom<sub>ECMC</sub>*  
**by force**  
**moreover have**  $\ll \text{Id } b \otimes \text{exp } a \ b : \mathcal{I} \otimes \text{exp } a \ b \rightarrow \text{exp } b \ b \otimes \text{exp } a \ b \gg$   
**using** *assms ide-exp by force*  
**ultimately show** *?thesis*  
**using** *assms comp-Curry-arr by force*  
**qed**  
**also have**  $\dots = \text{Curry}[\mathcal{I} \otimes \text{exp } a \ b, a, b]$   
 $(\text{eval } b \ b \cdot ((\text{exp } b \ b \otimes \text{eval } a \ b) \cdot (\text{Id } b \otimes \text{exp } a \ b \otimes a))) \cdot$   
 $\mathfrak{a}[\mathcal{I}, \text{exp } a \ b, a]) \cdot$   
 $(\iota \otimes \text{exp } a \ b)$   
**using** *assms assoc-naturality [of  $\text{Id } b \ \text{exp } a \ b \ a]$  ide-exp comp-assoc*

**by force**  
**also have** ... =  $\text{Curry}[\mathcal{I} \otimes \text{exp } a \ b, a, b]$   
 $((\text{eval } b \ b \cdot (\text{Id } b \otimes \text{Uncurry } a \ b \ (\text{exp } a \ b))) \cdot$   
 $\text{a}[\mathcal{I}, \text{exp } a \ b, a]) \cdot$   
 $(\iota \otimes \text{exp } a \ b)$

**by** (*simp add: assms Uncurry-exp comp-cod-arr comp-assoc interchange*)  
**also have** ... =  $\text{Curry}[\mathcal{I} \otimes \text{exp } a \ b, a, b]$   
 $((\text{eval } b \ b \cdot (\text{Id } b \otimes \text{eval } a \ b)) \cdot \text{a}[\mathcal{I}, \text{exp } a \ b, a]) \cdot$   
 $(\iota \otimes \text{exp } a \ b)$

**using** *assms comp-arr-dom*  
**by** (*metis eval-in-hom<sub>ECMC</sub> in-homE*)  
**also have** ... =  $\text{Curry}[\mathcal{I} \otimes \text{exp } a \ b, a, b]$   
 $((\text{eval } b \ b \cdot (\text{Id } b \otimes b)) \cdot (\mathcal{I} \otimes \text{eval } a \ b)) \cdot \text{a}[\mathcal{I}, \text{exp } a \ b, a]) \cdot$   
 $(\iota \otimes \text{exp } a \ b)$

**proof** –  
**have**  $\text{Id } b \otimes \text{eval } a \ b = (\text{Id } b \otimes b) \cdot (\mathcal{I} \otimes \text{eval } a \ b)$   
**using** *assms interchange*  
**by** (*metis Id-simps(1–2) comp-arr-dom comp-ide-arr eval-in-hom<sub>ECMC</sub>*  
*ide-in-hom seqI<sup>^</sup>*)  
**thus** *?thesis using comp-assoc by simp*  
**qed**

**also have** ... =  $\text{Curry}[\mathcal{I} \otimes \text{exp } a \ b, a, b]$   
 $((\text{l}[b] \cdot (\mathcal{I} \otimes \text{eval } a \ b)) \cdot \text{a}[\mathcal{I}, \text{exp } a \ b, a]) \cdot$   
 $(\iota \otimes \text{exp } a \ b)$

**using** *assms Id-def Uncurry-Curry lunit-in-hom ide-unity by simp*  
**also have** ... =  $\text{Curry}[\mathcal{I} \otimes \text{exp } a \ b, a, b]$   
 $(\text{eval } a \ b \cdot \text{l}[\text{exp } a \ b \otimes a] \cdot \text{a}[\mathcal{I}, \text{exp } a \ b, a]) \cdot$   
 $(\iota \otimes \text{exp } a \ b)$

**using** *assms lunit-naturality eval-in-hom<sub>ECMC</sub> in-homE lunit-naturality*  
*comp-assoc*  
**by** *metis*  
**also have** ... =  $\text{Curry}[\mathcal{I} \otimes \text{exp } a \ b, a, b]$   
 $(\text{Uncurry}[a, b] \ \text{l}[\text{exp } a \ b]) \cdot (\iota \otimes \text{exp } a \ b)$

**using** *assms ide-exp lunit-tensor' by force*  
**also have** ... =  $\text{l}[\text{exp } a \ b] \cdot (\iota \otimes \text{exp } a \ b)$

**using** *assms Curry-Uncurry lunit-in-hom ide-exp by auto*  
**finally show** ( $\text{Curry}[\text{exp } b \ b \otimes \text{exp } a \ b, a, b]$   
 $(\text{eval } b \ b \cdot (\text{exp } b \ b \otimes \text{eval } a \ b)) \cdot \text{a}[\text{exp } b \ b, \text{exp } a \ b, a]) \cdot$   
 $(\text{Id } b \otimes \text{exp } a \ b)) \cdot$   
 $(\iota \otimes \text{exp } a \ b) =$   
 $\text{l}[\text{exp } a \ b] \cdot (\iota \otimes \text{exp } a \ b)$

**by blast**  
**qed**

**finally have** 1:  $\text{l}[\text{exp } a \ b] \cdot$   
 $(\mathcal{I} \otimes \text{Comp } a \ b \ b \cdot (\text{Id } b \otimes \text{exp } a \ b)) \cdot \text{a}[\mathcal{I}, \mathcal{I}, \text{exp } a \ b] =$   
 $\text{l}[\text{exp } a \ b] \cdot (\iota \otimes \text{exp } a \ b)$

**by blast**  
**have**  $(\mathcal{I} \otimes \text{Comp } a \ b \ b \cdot (\text{Id } b \otimes \text{exp } a \ b)) \cdot \text{a}[\mathcal{I}, \mathcal{I}, \text{exp } a \ b] =$   
 $(\text{inv } \text{l}[\text{exp } a \ b] \cdot \text{l}[\text{exp } a \ b]) \cdot$

$(\mathcal{I} \otimes \text{Comp } a \ b \ b \cdot (\text{Id } b \otimes \text{exp } a \ b)) \cdot a[\mathcal{I}, \mathcal{I}, \text{exp } a \ b]$   
**using** *assms comp-cod-arr* **by** *simp*  
**also have**  $\dots = (\text{inv } l[\text{exp } a \ b] \cdot l[\text{exp } a \ b]) \cdot (\iota \otimes \text{exp } a \ b)$   
**using** *1 comp-assoc* **by** *simp*  
**also have**  $\dots = \iota \otimes \text{exp } a \ b$   
**using** *assms comp-cod-arr* [of  $\iota \otimes \text{exp } a \ b$   $l^{-1}[\text{exp } a \ b] \cdot l[\text{exp } a \ b]$ ] *arrI*  
**by** *auto*  
**finally have**  $(\mathcal{I} \otimes \text{Comp } a \ b \ b \cdot (\text{Id } b \otimes \text{exp } a \ b)) \cdot a[\mathcal{I}, \mathcal{I}, \text{exp } a \ b] =$   
 $\iota \otimes \text{exp } a \ b$   
**by** *blast*  
**thus** *?thesis*  
**using** *assms(1-2) 0 Las-nat-trans.naturality1 comp-assoc-assoc'(1)*  
*ide-exp ide-unity in-homE comp-assoc*  
**by** *metis*  
**qed**  
**qed**  
**qed**

**lemma** *Comp-assoc<sub>ECMC</sub>*:

**assumes** *ide a and ide b and ide c and ide d*

**shows**  $\langle\langle \text{Comp } a \ b \ d \cdot (\text{Comp } b \ c \ d \otimes \text{exp } a \ b) :$

$(\text{exp } c \ d \otimes \text{exp } b \ c) \otimes \text{exp } a \ b \rightarrow \text{exp } a \ d \rangle\rangle$

**and**  $\text{Comp } a \ b \ d \cdot (\text{Comp } b \ c \ d \otimes \text{exp } a \ b) =$

$\text{Comp } a \ c \ d \cdot (\text{exp } c \ d \otimes \text{Comp } a \ b \ c) \cdot a[\text{exp } c \ d, \text{exp } b \ c, \text{exp } a \ b]$

**proof** –

**show**  $\langle\langle \text{Comp } a \ b \ d \cdot (\text{Comp } b \ c \ d \otimes \text{exp } a \ b) :$

$(\text{exp } c \ d \otimes \text{exp } b \ c) \otimes \text{exp } a \ b \rightarrow \text{exp } a \ d \rangle\rangle$

**using** *assms* **by** *auto*

**show**  $\text{Comp } a \ b \ d \cdot (\text{Comp } b \ c \ d \otimes \text{exp } a \ b) =$

$\text{Comp } a \ c \ d \cdot (\text{exp } c \ d \otimes \text{Comp } a \ b \ c) \cdot a[\text{exp } c \ d, \text{exp } b \ c, \text{exp } a \ b]$

**proof** –

**have** *1*:  $\text{Uncurry}[a, d] (\text{Comp } a \ c \ d \cdot (\text{exp } c \ d \otimes \text{Comp } a \ b \ c) \cdot$   
 $a[\text{exp } c \ d, \text{exp } b \ c, \text{exp } a \ b]) =$

$\text{Uncurry}[a, d] (\text{Comp } a \ b \ d \cdot (\text{Comp } b \ c \ d \otimes \text{exp } a \ b))$

**proof** –

**have**  $\text{Uncurry}[a, d]$

$(\text{Comp } a \ c \ d \cdot (\text{exp } c \ d \otimes \text{Comp } a \ b \ c) \cdot$

$a[\text{exp } c \ d, \text{exp } b \ c, \text{exp } a \ b]) =$

$\text{Uncurry}[a, d]$

$(\text{Curry}[\text{exp } c \ d \otimes \text{exp } a \ c, a, d]$

$(\text{eval } c \ d \cdot (\text{exp } c \ d \otimes \text{eval } a \ c) \cdot a[\text{exp } c \ d, \text{exp } a \ c, a]) \cdot$

$(\text{exp } c \ d \otimes \text{Comp } a \ b \ c) \cdot a[\text{exp } c \ d, \text{exp } b \ c, \text{exp } a \ b])$

**using** *Comp-def* **by** *simp*

**also have**  $\dots = \text{Uncurry}[a, d]$

$(\text{Curry}[(\text{exp } c \ d \otimes \text{exp } b \ c) \otimes \text{exp } a \ b, a, d]$

$((\text{eval } c \ d \cdot (\text{exp } c \ d \otimes \text{eval } a \ c) \cdot a[\text{exp } c \ d, \text{exp } a \ c, a]) \cdot$

$((\text{exp } c \ d \otimes \text{Comp } a \ b \ c) \cdot$

$a[\text{exp } c \ d, \text{exp } b \ c, \text{exp } a \ b] \otimes a))$

**using** *assms*

*comp-Curry-arr*  
 $[of\ a\ (exp\ c\ d\ \otimes\ Comp\ a\ b\ c) \cdot a[exp\ c\ d,\ exp\ b\ c,\ exp\ a\ b]$   
 $(exp\ c\ d\ \otimes\ exp\ b\ c) \otimes\ exp\ a\ b\ exp\ c\ d\ \otimes\ exp\ a\ c$   
 $eval\ c\ d \cdot (exp\ c\ d\ \otimes\ eval\ a\ c) \cdot a[exp\ c\ d,\ exp\ a\ c,\ a]\ d]$

**by auto**  
**also have** ... =  $eval\ c\ d \cdot (exp\ c\ d\ \otimes\ eval\ a\ c) \cdot$   
 $(a[exp\ c\ d,\ exp\ a\ c,\ a] \cdot ((exp\ c\ d\ \otimes\ Comp\ a\ b\ c) \otimes\ a)) \cdot$   
 $(a[exp\ c\ d,\ exp\ b\ c,\ exp\ a\ b] \otimes\ a)$

**using assms Uncurry-Curry ide-exp interchange comp-assoc by simp**  
**also have** ... =  $eval\ c\ d \cdot$   
 $(exp\ c\ d\ \otimes\ eval\ a\ c) \cdot$   
 $(a[exp\ c\ d,\ exp\ a\ c,\ a] \cdot$   
 $((exp\ c\ d\ \otimes$   
 $Curry[exp\ b\ c\ \otimes\ exp\ a\ b,\ a,\ c]$   
 $(eval\ b\ c \cdot (exp\ b\ c\ \otimes\ eval\ a\ b) \cdot a[exp\ b\ c,\ exp\ a\ b,\ a]))$   
 $\otimes\ a)) \cdot$   
 $(a[exp\ c\ d,\ exp\ b\ c,\ exp\ a\ b] \otimes\ a)$

**unfolding Comp-def by simp**  
**also have** ... =  $eval\ c\ d \cdot$   
 $(exp\ c\ d\ \otimes\ eval\ a\ c) \cdot$   
 $((exp\ c\ d\ \otimes$   
 $Curry[exp\ b\ c\ \otimes\ exp\ a\ b,\ a,\ c]$   
 $(eval\ b\ c \cdot (exp\ b\ c\ \otimes\ eval\ a\ b) \cdot a[exp\ b\ c,\ exp\ a\ b,\ a])$   
 $\otimes\ a) \cdot$   
 $a[exp\ c\ d,\ exp\ b\ c\ \otimes\ exp\ a\ b,\ a] \cdot$   
 $(a[exp\ c\ d,\ exp\ b\ c,\ exp\ a\ b] \otimes\ a)$

**using assms assoc-naturality [of exp c d - a] Comp-def Comp-simps(1-3)**  
*ide-exp ide-char*

**by (metis (no-types, lifting) mem-Collect-eq)**  
**also have** ... =  $eval\ c\ d \cdot$   
 $((exp\ c\ d\ \otimes\ eval\ a\ c) \cdot$   
 $(exp\ c\ d\ \otimes$   
 $Curry[exp\ b\ c\ \otimes\ exp\ a\ b,\ a,\ c]$   
 $(eval\ b\ c \cdot (exp\ b\ c\ \otimes\ eval\ a\ b) \cdot a[exp\ b\ c,\ exp\ a\ b,\ a])$   
 $\otimes\ a)) \cdot$   
 $a[exp\ c\ d,\ exp\ b\ c\ \otimes\ exp\ a\ b,\ a] \cdot$   
 $(a[exp\ c\ d,\ exp\ b\ c,\ exp\ a\ b] \otimes\ a)$

**using comp-assoc by simp**  
**also have** ... =  $eval\ c\ d \cdot$   
 $(exp\ c\ d\ \otimes$   
 $Uncurry[a,\ c]$   
 $(Curry[exp\ b\ c\ \otimes\ exp\ a\ b,\ a,\ c]$   
 $(eval\ b\ c \cdot (exp\ b\ c\ \otimes\ eval\ a\ b) \cdot$   
 $a[exp\ b\ c,\ exp\ a\ b,\ a]))) \cdot$   
 $a[exp\ c\ d,\ exp\ b\ c\ \otimes\ exp\ a\ b,\ a] \cdot$   
 $(a[exp\ c\ d,\ exp\ b\ c,\ exp\ a\ b] \otimes\ a)$

**using assms Comp-def Comp-in-hom interchange by auto**  
**also have** ... =  $eval\ c\ d \cdot$   
 $(exp\ c\ d\ \otimes\ (eval\ b\ c \cdot (exp\ b\ c\ \otimes\ eval\ a\ b) \cdot$

$a[\text{exp } b \ c, \text{exp } a \ b, a]) \cdot$   
 $a[\text{exp } c \ d, \text{exp } b \ c \otimes \text{exp } a \ b, a] \cdot$   
 $(a[\text{exp } c \ d, \text{exp } b \ c, \text{exp } a \ b] \otimes a)$

**using** *assms ide-exp tensor-in-hom ide-in-hom ide-exp eval-in-hom<sub>ECMC</sub>*  
*assoc-in-hom Uncurry-Curry*

**by force**

**also have**  $\dots = \text{eval } c \ d \cdot$   
 $((\text{exp } c \ d \otimes \text{eval } b \ c) \cdot$   
 $(\text{exp } c \ d \otimes \text{exp } b \ c \otimes \text{eval } a \ b) \cdot$   
 $(\text{exp } c \ d \otimes a[\text{exp } b \ c, \text{exp } a \ b, a])) \cdot$   
 $a[\text{exp } c \ d, \text{exp } b \ c \otimes \text{exp } a \ b, a] \cdot$   
 $(a[\text{exp } c \ d, \text{exp } b \ c, \text{exp } a \ b] \otimes a)$

**using** *assms ide-exp tensor-in-hom interchange* **by auto**

**also have**  $\dots = \text{eval } c \ d \cdot$   
 $(\text{exp } c \ d \otimes \text{eval } b \ c) \cdot$   
 $(\text{exp } c \ d \otimes \text{exp } b \ c \otimes \text{eval } a \ b) \cdot$   
 $(\text{exp } c \ d \otimes a[\text{exp } b \ c, \text{exp } a \ b, a]) \cdot$   
 $a[\text{exp } c \ d, \text{exp } b \ c \otimes \text{exp } a \ b, a] \cdot$   
 $(a[\text{exp } c \ d, \text{exp } b \ c, \text{exp } a \ b] \otimes a)$

**using** *comp-assoc* **by simp**

**finally have**  $*$ :  $\text{Uncurry}[a, d] (\text{Comp } a \ c \ d \cdot (\text{exp } c \ d \otimes \text{Comp } a \ b \ c) \cdot$   
 $a[\text{exp } c \ d, \text{exp } b \ c, \text{exp } a \ b]) =$   
 $\text{eval } c \ d \cdot$   
 $(\text{exp } c \ d \otimes \text{eval } b \ c) \cdot$   
 $(\text{exp } c \ d \otimes \text{exp } b \ c \otimes \text{eval } a \ b) \cdot$   
 $(\text{exp } c \ d \otimes a[\text{exp } b \ c, \text{exp } a \ b, a]) \cdot$   
 $a[\text{exp } c \ d, \text{exp } b \ c \otimes \text{exp } a \ b, a] \cdot$   
 $(a[\text{exp } c \ d, \text{exp } b \ c, \text{exp } a \ b] \otimes a)$

**by blast**

**have**  $\text{Uncurry}[a, d] (\text{Comp } a \ b \ d \cdot (\text{Comp } b \ c \ d \otimes \text{exp } a \ b)) =$   
 $\text{Uncurry}[a, d]$   
 $(\text{Curry}[\text{exp } b \ d \otimes \text{exp } a \ b, a, d]$   
 $(\text{eval } b \ d \cdot (\text{exp } b \ d \otimes \text{eval } a \ b) \cdot a[\text{exp } b \ d, \text{exp } a \ b, a]) \cdot$   
 $(\text{Comp } b \ c \ d \otimes \text{exp } a \ b))$

**using** *Comp-def* **by simp**

**also have**  $\dots = \text{Uncurry}[a, d]$   
 $(\text{Curry}[(\text{exp } c \ d \otimes \text{exp } b \ c) \otimes \text{exp } a \ b, a, d]$   
 $(\text{eval } b \ d \cdot (\text{exp } b \ d \otimes \text{eval } a \ b) \cdot a[\text{exp } b \ d, \text{exp } a \ b, a]) \cdot$   
 $((\text{Comp } b \ c \ d \otimes \text{exp } a \ b) \otimes a))$

**proof** –

**have**  $\ll \text{Comp } b \ c \ d \otimes \text{exp } a \ b :$   
 $(\text{exp } c \ d \otimes \text{exp } b \ c) \otimes \text{exp } a \ b \rightarrow \text{exp } b \ d \otimes \text{exp } a \ b \gg$

**using** *assms ide-exp* **by force**

**moreover have**  $\ll \text{eval } b \ d \cdot (\text{exp } b \ d \otimes \text{eval } a \ b) \cdot a[\text{exp } b \ d, \text{exp } a \ b, a]$   
 $: (\text{exp } b \ d \otimes \text{exp } a \ b) \otimes a \rightarrow d \gg$

**using** *assms ide-exp tensor-in-hom ide-in-hom ide-exp eval-in-hom<sub>ECMC</sub>*  
**by force**

**ultimately show** *?thesis*

**using** *comp-Curry-arr assms comp-assoc* **by auto**

**qed**  
**also have** ... =  $eval\ b\ d \cdot (exp\ b\ d \otimes eval\ a\ b) \cdot a[exp\ b\ d, exp\ a\ b, a] \cdot$   
 $((Comp\ b\ c\ d \otimes exp\ a\ b) \otimes a)$   
**using** *assms ide-exp Uncurry-Curry* **by force**  
**also have** ... =  $eval\ b\ d \cdot$   
 $((exp\ b\ d \otimes eval\ a\ b) \cdot (Comp\ b\ c\ d \otimes exp\ a\ b \otimes a)) \cdot$   
 $a[exp\ c\ d \otimes exp\ b\ c, exp\ a\ b, a]$   
**using** *assms ide-exp comp-assoc*  
*assoc-naturality [of Comp b c d exp a b a]*  
**by force**  
**also have** ... =  $eval\ b\ d \cdot (Comp\ b\ c\ d \otimes eval\ a\ b) \cdot$   
 $a[exp\ c\ d \otimes exp\ b\ c, exp\ a\ b, a]$   
**by** (*simp add: assms comp-arr-dom comp-cod-arr interchange*)  
**also have** ... =  $eval\ b\ d \cdot$   
 $(Curry[exp\ c\ d \otimes exp\ b\ c, b, d]$   
 $(eval\ c\ d \cdot (exp\ c\ d \otimes eval\ b\ c) \cdot a[exp\ c\ d, exp\ b\ c, b])$   
 $\otimes eval\ a\ b) \cdot$   
 $a[exp\ c\ d \otimes exp\ b\ c, exp\ a\ b, a]$   
**unfolding** *Comp-def* **by simp**  
**also have** ... =  $eval\ b\ d \cdot$   
 $((Curry[exp\ c\ d \otimes exp\ b\ c, b, d]$   
 $(eval\ c\ d \cdot (exp\ c\ d \otimes eval\ b\ c) \cdot a[exp\ c\ d, exp\ b\ c, b])$   
 $\otimes b) \cdot$   
 $((exp\ c\ d \otimes exp\ b\ c) \otimes eval\ a\ b)) \cdot$   
 $a[exp\ c\ d \otimes exp\ b\ c, exp\ a\ b, a]$   
**by** (*metis (no-types, lifting) assms Comp-def Comp-simps(1-2)*  
*comp-arr-dom comp-cod-arr eval-simps(1,3) interchange*)  
**also have** ... =  $Uncurry[b, d]$   
 $(Curry[exp\ c\ d \otimes exp\ b\ c, b, d]$   
 $(eval\ c\ d \cdot (exp\ c\ d \otimes eval\ b\ c) \cdot a[exp\ c\ d, exp\ b\ c, b])) \cdot$   
 $((exp\ c\ d \otimes exp\ b\ c) \otimes eval\ a\ b) \cdot$   
 $a[exp\ c\ d \otimes exp\ b\ c, exp\ a\ b, a]$   
**using** *comp-assoc* **by simp**  
**also have** ... =  $eval\ c\ d \cdot$   
 $(exp\ c\ d \otimes eval\ b\ c) \cdot$   
 $(a[exp\ c\ d, exp\ b\ c, b] \cdot$   
 $((exp\ c\ d \otimes exp\ b\ c) \otimes eval\ a\ b)) \cdot$   
 $a[exp\ c\ d \otimes exp\ b\ c, exp\ a\ b, a]$   
**using** *assms ide-exp Uncurry-Curry comp-assoc* **by auto**  
**also have** ... =  $eval\ c\ d \cdot$   
 $(exp\ c\ d \otimes eval\ b\ c) \cdot$   
 $((exp\ c\ d \otimes exp\ b\ c \otimes eval\ a\ b) \cdot$   
 $a[exp\ c\ d, exp\ b\ c, exp\ a\ b \otimes a]) \cdot$   
 $a[exp\ c\ d \otimes exp\ b\ c, exp\ a\ b, a]$   
**using** *assoc-naturality [of exp c d exp b c eval a b]*  
**by** (*metis assms arr-cod cod-cod Curry-in-hom dom-dom eval-in-hom<sub>ECMC</sub>*  
*ide-exp in-homE*)  
**also have** ... =  $eval\ c\ d \cdot$   
 $(exp\ c\ d \otimes eval\ b\ c) \cdot$

```

      (exp c d ⊗ exp b c ⊗ eval a b) ·
      a[exp c d, exp b c, exp a b ⊗ a] ·
      a[exp c d ⊗ exp b c, exp a b, a]
    using comp-assoc by simp
  finally have **: Uncurry[a, d] (Comp a b d · (Comp b c d ⊗ exp a b)) =
    eval c d ·
    (exp c d ⊗ eval b c) ·
    (exp c d ⊗ exp b c ⊗ eval a b) ·
    a[exp c d, exp b c, exp a b ⊗ a] ·
    a[exp c d ⊗ exp b c, exp a b, a]

  by blast
  show ?thesis
  using * ** assms ide-exp pentagon by force
qed
have Comp a b d · (Comp b c d ⊗ exp a b) =
  Curry[(exp c d ⊗ exp b c) ⊗ exp a b, a, d]
  (Uncurry[a, d] (Comp a b d · (Comp b c d ⊗ exp a b)))
  using assms ide-exp Curry-Uncurry by fastforce
also have ... = Curry[(exp c d ⊗ exp b c) ⊗ exp a b, a, d]
  (Uncurry[a, d] (Comp a c d · (exp c d ⊗ Comp a b c) ·
    a[exp c d, exp b c, exp a b]))
  using 1 by simp
also have ... = Comp a c d · (exp c d ⊗ Comp a b c) ·
  a[exp c d, exp b c, exp a b]
  using assms ide-exp Curry-Uncurry by simp
finally show Comp a b d · (Comp b c d ⊗ exp a b) =
  Comp a c d · (exp c d ⊗ Comp a b c) · a[exp c d, exp b c, exp a b]
  by blast
qed
qed
end

end

```

## 1.4 Cartesian Closed Monoidal Categories

A *cartesian closed monoidal category* is a cartesian monoidal category that is a closed monoidal category with respect to a chosen product. This is not quite the same thing as a cartesian closed category, because a cartesian monoidal category (being a monoidal category) has chosen structure (the tensor, associators, and unitors), whereas we have defined a cartesian closed category to be an abstract category satisfying certain properties that are expressed without assuming any chosen structure.

```

theory CartesianClosedMonoidalCategory
imports Category3.CartesianClosedCategory MonoidalCategory.CartesianMonoidalCategory
  ClosedMonoidalCategory

```



**begin**

**locale** *cartesian-closed-monoidal-category* =  
  *cartesian-monoidal-category* +  
  *closed-monoidal-category*

**locale** *elementary-cartesian-closed-monoidal-category* =  
  *cartesian-monoidal-category* +  
  *elementary-closed-monoidal-category*

**begin**

**lemmas** *prod-eq-tensor* [*simp*]

**end**

The following is the main purpose for the current theory: to show that a cartesian closed category with chosen structure determines a cartesian closed monoidal category.

**context** *elementary-cartesian-closed-category*  
**begin**

**interpretation** *CMC: cartesian-monoidal-category*  $C \text{ Prod } \alpha \iota$   
  **using** *extends-to-cartesian-monoidal-category<sub>ECC</sub>* **by** *blast*

**interpretation** *CMC: closed-monoidal-category*  $C \text{ Prod } \alpha \iota$   
  **using** *CMC.T.extensionality interchange left-adjoint-prod*  
  **by** *unfold-locales*  
  (*auto simp add: left-adjoint-functor.ex-terminal-arrow*)

**lemma** *extends-to-closed-monoidal-category<sub>ECCC</sub>*:  
  **shows** *closed-monoidal-category*  $C \text{ Prod } \alpha \iota$   
  ..

**lemma** *extends-to-cartesian-closed-monoidal-category<sub>ECCC</sub>*:  
  **shows** *cartesian-closed-monoidal-category*  $C \text{ Prod } \alpha \iota$   
  ..

**interpretation** *CMC: elementary-monoidal-category*  
   $C \text{ CMC.tensor CMC.unital CMC.lunit CMC.runit CMC.assoc}$   
  **using** *CMC.induces-elementary-monoidal-category* **by** *blast*

**interpretation** *CMC: elementary-closed-monoidal-category*  
   $C \text{ Prod } \alpha \iota \text{ exp eval curry}$   
  **using** *eval-in-hom-ax curry-in-hom uncurry-curry-ax curry-uncurry-ax*  
  **by** *unfold-locales auto*

**lemma** *extends-to-elementary-closed-monoidal-category<sub>ECCC</sub>*:  
  **shows** *elementary-closed-monoidal-category*  $C \text{ Prod } \alpha \iota \text{ exp eval curry}$   
  ..

**lemma** *extends-to-elementary-cartesian-closed-monoidal-category*<sub>ECCC</sub>:  
**shows** *elementary-cartesian-closed-monoidal-category*  $C$  *Prod*  $\alpha$   $\iota$  *exp eval curry*  
 ..

**end**

**context** *elementary-cartesian-closed-monoidal-category*  
**begin**

**interpretation** *elementary-monoidal-category*  $C$  *tensor unity lunit runit assoc*  
**using** *induces-elementary-monoidal-category* **by** *blast*

The following fact is not used in the present article, but it is a natural and likely useful lemma for which I constructed a proof at one point. The proof requires cartesianness; I suspect this is essential, but I am not absolutely certain of it.

**lemma** *isomorphic-exp-prod*:  
**assumes** *ide a and ide b and ide x*  
**shows**  $\langle \text{Curry}[\text{exp } x (a \otimes b), x, a] (\mathfrak{p}_1[a, b] \cdot \text{eval } x (a \otimes b)),$   
 $\text{Curry}[\text{exp } x (a \otimes b), x, b] (\mathfrak{p}_0[a, b] \cdot \text{eval } x (a \otimes b)) \rangle$   
 $: \text{exp } x (a \otimes b) \rightarrow \text{exp } x a \otimes \text{exp } x b$   
**(is**  $\langle ?C, ?D \rangle : \text{exp } x (a \otimes b) \rightarrow \text{exp } x a \otimes \text{exp } x b$  **)**  
**and**  $\langle \text{Curry}[\text{exp } x a \otimes \text{exp } x b, x, a \otimes b]$   
 $\langle \text{eval } x a \cdot \langle \mathfrak{p}_1[\text{exp } x a, \text{exp } x b] \cdot \mathfrak{p}_1[\text{exp } x a \otimes \text{exp } x b, x],$   
 $\mathfrak{p}_0[\text{exp } x a \otimes \text{exp } x b, x] \rangle,$   
 $\text{eval } x b \cdot \langle \mathfrak{p}_0[\text{exp } x a, \text{exp } x b] \cdot \mathfrak{p}_1[\text{exp } x a \otimes \text{exp } x b, x],$   
 $\mathfrak{p}_0[\text{exp } x a \otimes \text{exp } x b, x] \rangle \rangle$   
 $: \text{exp } x a \otimes \text{exp } x b \rightarrow \text{exp } x (a \otimes b) \rangle$   
**(is**  $\langle \text{Curry}[\text{exp } x a \otimes \text{exp } x b, x, a \otimes b] \langle ?A, ?B \rangle$   
 $: \text{exp } x a \otimes \text{exp } x b \rightarrow \text{exp } x (a \otimes b) \rangle$  **)**  
**and** *inverse-arrows*  
 $\langle \text{Curry}[\text{exp } x a \otimes \text{exp } x b, x, a \otimes b]$   
 $\langle \text{eval } x a \cdot \langle \mathfrak{p}_1[\text{exp } x a, \text{exp } x b] \cdot \mathfrak{p}_1[\text{exp } x a \otimes \text{exp } x b, x],$   
 $\mathfrak{p}_0[\text{exp } x a \otimes \text{exp } x b, x] \rangle,$   
 $\text{eval } x b \cdot \langle \mathfrak{p}_0[\text{exp } x a, \text{exp } x b] \cdot \mathfrak{p}_1[\text{exp } x a \otimes \text{exp } x b, x],$   
 $\mathfrak{p}_0[\text{exp } x a \otimes \text{exp } x b, x] \rangle \rangle \rangle$   
 $\langle \text{Curry}[\text{exp } x (a \otimes b), x, a] (\mathfrak{p}_1[a, b] \cdot \text{eval } x (a \otimes b)),$   
 $\text{Curry}[\text{exp } x (a \otimes b), x, b] (\mathfrak{p}_0[a, b] \cdot \text{eval } x (a \otimes b)) \rangle$   
**and** *isomorphic*  $(\text{exp } x (a \otimes b)) (\text{exp } x a \otimes \text{exp } x b)$   
**proof** –  
**have**  $A: \langle ?A : (\text{exp } x a \otimes \text{exp } x b) \otimes x \rightarrow a \rangle$   
**using** *assms by auto*  
**have**  $B: \langle ?B : (\text{exp } x a \otimes \text{exp } x b) \otimes x \rightarrow b \rangle$   
**using** *assms by auto*  
**have**  $AB: \langle \langle ?A, ?B \rangle : (\text{exp } x a \otimes \text{exp } x b) \otimes x \rightarrow a \otimes b \rangle$   
**by** *(metis A B ECC.tuple-in-hom prod-eq-tensor)*  
**have**  $C: \langle ?C : \text{exp } x (a \otimes b) \rightarrow \text{exp } x a \rangle$   
**using** *assms by auto*

**have**  $D$ : « $?D : \text{exp } x (a \otimes b) \rightarrow \text{exp } x b$ »  
**using** *assms by auto*  
**show**  $CD$ : « $\langle ?C, ?D \rangle : \text{exp } x (a \otimes b) \rightarrow \text{exp } x a \otimes \text{exp } x b$ »  
**using**  $C D$  **by** *fastforce*  
**show**  $1$ : « $\text{Curry}[\text{exp } x a \otimes \text{exp } x b, x, a \otimes b] \langle ?A, ?B \rangle$   
 $: (\text{exp } x a \otimes \text{exp } x b) \rightarrow \text{exp } x (a \otimes b)$ »  
**by** (*simp add: AB assms(1-3) Curry-in-hom*)  
**show** *inverse-arrows* ( $\text{Curry}[\text{exp } x a \otimes \text{exp } x b, x, a \otimes b] \langle ?A, ?B \rangle \langle ?C, ?D \rangle$ )  
**proof**  
**show** *ide* ( $\text{Curry}[\text{exp } x a \otimes \text{exp } x b, x, a \otimes b] \langle ?A, ?B \rangle \cdot \langle ?C, ?D \rangle$ )  
**proof** –  
**have**  $\text{Curry}[\text{exp } x a \otimes \text{exp } x b, x, a \otimes b] \langle ?A, ?B \rangle \cdot \langle ?C, ?D \rangle =$   
 $\text{Curry}[\text{exp } x (a \otimes b), x, a \otimes b] (\langle ?A, ?B \rangle \cdot (\langle ?C, ?D \rangle \otimes x))$   
**using** *assms AB CD comp-Curry-arr by presburger*  
**also have**  $\dots = \text{Curry}[\text{exp } x (a \otimes b), x, a \otimes b]$   
 $\langle ?A \cdot (\langle ?C, ?D \rangle \otimes x), ?B \cdot (\langle ?C, ?D \rangle \otimes x) \rangle$   
**proof** –  
**have** *span*  $?A ?B$   
**using**  $A B$  **by** *fastforce*  
**moreover have** *arr* ( $\langle ?C, ?D \rangle \otimes x$ )  
**using** *assms CD by auto*  
**moreover have** *dom*  $?A = \text{cod } (\langle ?C, ?D \rangle \otimes x)$   
**by** (*metis A CD assms(3) cod-tensor ide-char in-homE*)  
**ultimately show** *?thesis*  
**using** *assms ECC.comp-tuple-arr by metis*  
**qed**  
**also have**  $\dots = \text{Curry}[\text{exp } x (a \otimes b), x, a \otimes b]$   
 $\langle \text{Uncurry}[x, a] ?C, \text{eval } x b \cdot (?D \otimes x) \rangle$   
**proof** –  
**have**  $?A \cdot (\langle ?C, ?D \rangle \otimes x) = \text{Uncurry}[x, a] ?C$   
**proof** –  
**have**  $?A \cdot (\langle ?C, ?D \rangle \otimes x) =$   
 $\text{eval } x a \cdot$   
 $\langle \mathfrak{p}_1[\text{exp } x a, \text{exp } x b] \cdot \mathfrak{p}_1[\text{exp } x a \otimes \text{exp } x b, x] \cdot (\langle ?C, ?D \rangle \otimes x),$   
 $\mathfrak{p}_0[\text{exp } x a \otimes \text{exp } x b, x] \cdot (\langle ?C, ?D \rangle \otimes x) \rangle$   
**using** *assms ECC.comp-tuple-arr comp-assoc by simp*  
**also have**  $\dots = \text{eval } x a \cdot$   
 $\langle ?C \cdot \mathfrak{p}_1[\text{exp } x (a \otimes b), x], x \cdot \mathfrak{p}_0[\text{exp } x (a \otimes b), x] \rangle$   
**using** *assms ECC.pr-naturality(1-2) by auto*  
**also have**  $\dots = \text{eval } x a \cdot (?C \otimes x) \cdot$   
 $\langle \mathfrak{p}_1[\text{exp } x (a \otimes b), x], \mathfrak{p}_0[\text{exp } x (a \otimes b), x] \rangle$   
**using** *assms*  
 $\text{ECC.prod-tuple}$   
 $[of \ \mathfrak{p}_1[\text{exp } x (a \otimes b), x] \ \mathfrak{p}_0[\text{exp } x (a \otimes b), x] \ ?C \ x]$   
**by** *simp*  
**also have**  $\dots = \text{Uncurry}[x, a] ?C$   
**using** *assms C ECC.tuple-pr comp-arr-ide comp-arr-dom by auto*  
**finally show** *?thesis by blast*  
**qed**

**moreover have**  $?B \cdot (\langle ?C, ?D \rangle \otimes x) = \text{Uncurry}[x, b] ?D$   
**proof** –  
**have**  $?B \cdot (\langle ?C, ?D \rangle \otimes x) =$   
 $\text{eval } x \ b \cdot$   
 $\langle \mathfrak{p}_0[\text{exp } x \ a, \text{exp } x \ b] \cdot \mathfrak{p}_1[\text{exp } x \ a \otimes \text{exp } x \ b, x] \cdot (\langle ?C, ?D \rangle \otimes x),$   
 $\mathfrak{p}_0[\text{exp } x \ a \otimes \text{exp } x \ b, x] \cdot (\langle ?C, ?D \rangle \otimes x) \rangle$   
**using** *assms*  $\text{ECC.comp-tuple-arr comp-assoc}$  **by** *simp*  
**also have**  $\dots = \text{eval } x \ b \cdot$   
 $\langle ?D \cdot \mathfrak{p}_1[\text{exp } x \ (a \otimes b), x], x \cdot \mathfrak{p}_0[\text{exp } x \ (a \otimes b), x] \rangle$   
**using** *assms*  $C \ \text{ECC.pr-naturality}(1-2)$  **by** *auto*  
**also have**  $\dots = \text{eval } x \ b \cdot (?D \otimes x) \cdot$   
 $\langle \mathfrak{p}_1[\text{exp } x \ (a \otimes b), x], \mathfrak{p}_0[\text{exp } x \ (a \otimes b), x] \rangle$   
**using** *assms*  
 $\text{ECC.prod-tuple}$   
 $[of \ \mathfrak{p}_1[\text{exp } x \ (a \otimes b), x] \ \mathfrak{p}_0[\text{exp } x \ (a \otimes b), x] \ ?D \ x]$   
**by** *simp*  
**also have**  $\dots = \text{Uncurry}[x, b] ?D$   
**using** *assms*  $C \ \text{ECC.tuple-pr comp-arr-ide comp-arr-dom}$  **by** *auto*  
**finally show**  $?thesis$  **by** *blast*  
**qed**  
**ultimately show**  $?thesis$  **by** *simp*  
**qed**  
**also have**  $\dots = \text{Curry}[\text{exp } x \ (a \otimes b), x, a \otimes b]$   
 $(\langle \mathfrak{p}_1[a, b] \cdot \text{eval } x \ (a \otimes b), \mathfrak{p}_0[a, b] \cdot \text{eval } x \ (a \otimes b) \rangle)$   
**using** *assms*  $\text{Uncurry-Curry}$  **by** *auto*  
**also have**  $\dots = \text{Curry}[\text{exp } x \ (a \otimes b), x, a \otimes b]$   
 $(\langle \mathfrak{p}_1[a, b], \mathfrak{p}_0[a, b] \rangle \cdot \text{eval } x \ (a \otimes b))$   
**using** *assms*  $\text{ECC.comp-tuple-arr}$   $[of \ \mathfrak{p}_1[a, b] \ \mathfrak{p}_0[a, b] \ \text{eval } x \ (a \otimes b)]$   
**by** *simp*  
**also have**  $\dots = \text{Curry}[\text{exp } x \ (a \otimes b), x, a \otimes b] (\text{eval } x \ (a \otimes b))$   
**using** *assms*  $\text{comp-cod-arr}$  **by** *simp*  
**also have**  $\dots = \text{exp } x \ (a \otimes b)$   
**using** *assms*  $\text{Curry-Uncurry ide-exp ide-in-hom tensor-preserves-ide}$   
 $\text{Uncurry-exp}$   
**by** *metis*  
**finally have**  $\text{Curry}[\text{exp } x \ a \otimes \text{exp } x \ b, x, a \otimes b] \langle ?A, ?B \rangle \cdot \langle ?C, ?D \rangle =$   
 $\text{exp } x \ (a \otimes b)$   
**by** *blast*  
**thus**  $?thesis$   
**using** *assms*  $\text{ide-exp tensor-preserves-ide}$  **by** *presburger*  
**qed**  
**show** *ide*  $(\langle ?C, ?D \rangle \cdot \text{Curry}[\text{exp } x \ a \otimes \text{exp } x \ b, x, a \otimes b] \langle ?A, ?B \rangle)$   
**proof** –  
**have**  $?2: \text{span } \mathfrak{p}_1[\text{exp } x \ a \otimes \text{exp } x \ b, x] \ \mathfrak{p}_0[\text{exp } x \ a \otimes \text{exp } x \ b, x]$   
**using** *assms* **by** *simp*  
**have**  $?3: \text{seq } x \ \mathfrak{p}_0[\text{exp } x \ a \otimes \text{exp } x \ b, x]$   
**using** *assms* **by** *simp*  
**have**  $\langle ?C, ?D \rangle \cdot \text{Curry}[\text{exp } x \ a \otimes \text{exp } x \ b, x, a \otimes b] \langle ?A, ?B \rangle =$   
 $\langle ?C \cdot \text{Curry}[\text{exp } x \ a \otimes \text{exp } x \ b, x, a \otimes b] \langle ?A, ?B \rangle,$

$?D \cdot \text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a \ \otimes \ b] \langle ?A, ?B \rangle$   
**using** *assms C D 1 ECC.comp-tuple-arr* **by** (*metis in-homE*)  
**also have** ... =  $\langle \mathfrak{p}_1[\text{exp } x \ a, \ \text{exp } x \ b], \ \mathfrak{p}_0[\text{exp } x \ a, \ \text{exp } x \ b] \rangle$   
**proof** –  
**have**  $\text{Curry}[\text{exp } x \ (a \ \otimes \ b), x, a] (\mathfrak{p}_1[a, b] \cdot \text{eval } x \ (a \ \otimes \ b)) \cdot$   
 $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a \ \otimes \ b] \langle ?A, ?B \rangle =$   
 $\mathfrak{p}_1[\text{exp } x \ a, \ \text{exp } x \ b]$   
**proof** –  
**have**  $\text{Curry}[\text{exp } x \ (a \ \otimes \ b), x, a] (\mathfrak{p}_1[a, b] \cdot \text{eval } x \ (a \ \otimes \ b)) \cdot$   
 $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a \ \otimes \ b] \langle ?A, ?B \rangle =$   
 $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a]$   
 $((\mathfrak{p}_1[a, b] \cdot \text{eval } x \ (a \ \otimes \ b)) \cdot$   
 $(\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a \ \otimes \ b] \langle ?A, ?B \rangle \otimes x))$   
**using** *assms 1 comp-Curry-arr* **by** *auto*  
**also have** ... =  $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a]$   
 $(\mathfrak{p}_1[a, b] \cdot \text{eval } x \ (a \ \otimes \ b) \cdot$   
 $(\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a \ \otimes \ b] \langle ?A, ?B \rangle \otimes x))$   
**using** *comp-assoc* **by** *simp*  
**also have** ... =  $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a] (\mathfrak{p}_1[a, b] \cdot \langle ?A, ?B \rangle)$   
**using** *assms AB Uncurry-Curry ide-exp tensor-preserves-ide* **by** *simp*  
**also have** ... =  $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a]$   
 $(\text{eval } x \ a \cdot$   
 $\langle \mathfrak{p}_1[\text{exp } x \ a, \ \text{exp } x \ b] \cdot \mathfrak{p}_1[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x],$   
 $\mathfrak{p}_0[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x] \rangle)$   
**using** *assms A B ECC.pr-tuple(1)* **by** *fastforce*  
**also have** ... =  $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a]$   
 $(\text{eval } x \ a \cdot (\mathfrak{p}_1[\text{exp } x \ a, \ \text{exp } x \ b] \otimes x) \cdot$   
 $\langle \mathfrak{p}_1[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x],$   
 $\mathfrak{p}_0[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x] \rangle)$   
**proof** –  
**have** *seq*  $\mathfrak{p}_1[\text{exp } x \ a, \ \text{exp } x \ b] \ \mathfrak{p}_1[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x]$   
**using** *assms* **by** *auto*  
**thus** *?thesis*  
**using** *assms 2 3 prod-eq-tensor comp-ide-arr ECC.prod-tuple*  
**by** *metis*  
**qed**  
**also have** ... =  $\text{Curry} (\text{exp } x \ a \ \otimes \ \text{exp } x \ b) \ x \ a$   
 $(\text{eval } x \ a \cdot (\mathfrak{p}_1[\text{exp } x \ a, \ \text{exp } x \ b] \otimes x))$   
**using** *assms comp-arr-dom* **by** *simp*  
**also have** ... =  $\mathfrak{p}_1[\text{exp } x \ a, \ \text{exp } x \ b]$   
**using** *assms Curry-Uncurry* **by** *simp*  
**finally show** *?thesis* **by** *blast*  
**qed**  
**moreover have**  $\text{Curry}[\text{exp } x \ (a \ \otimes \ b), x, b] (\mathfrak{p}_0[a, b] \cdot \text{eval } x \ (a \ \otimes \ b)) \cdot$   
 $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a \ \otimes \ b] \langle ?A, ?B \rangle =$   
 $\mathfrak{p}_0[\text{exp } x \ a, \ \text{exp } x \ b]$   
**proof** –  
**have**  $\text{Curry}[\text{exp } x \ (a \ \otimes \ b), x, b] (\mathfrak{p}_0[a, b] \cdot \text{eval } x \ (a \ \otimes \ b)) \cdot$   
 $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a \ \otimes \ b] \langle ?A, ?B \rangle =$

$$\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x, \ b]$$

$$((\mathfrak{p}_0[a, b] \cdot \text{eval } x \ (a \ \otimes \ b)) \cdot$$

$$(\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x, \ a \ \otimes \ b] \langle ?A, ?B \rangle \otimes x))$$

**proof** –

**have** « $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x, \ a \ \otimes \ b] \langle ?A, ?B \rangle$   
 $: \text{exp } x \ a \ \otimes \ \text{exp } x \ b \rightarrow \text{exp } x \ (a \ \otimes \ b)$ »

**using** 1 **by** *blast*

**moreover have** « $\mathfrak{p}_0[a, b] \cdot \text{eval } x \ (a \ \otimes \ b) : \text{exp } x \ (a \ \otimes \ b) \otimes x \rightarrow b$ »

**using** *assms*

**by** (*metis* (*no-types, lifting*) *ECC.pr0-in-hom' ECC.pr-simps(2)*  
*comp-in-homI eval-in-hom<sub>EMC</sub> prod-eq-tensor tensor-preserves-ide*)

**ultimately show** *?thesis*

**using** *assms comp-Curry-arr* **by** *simp*

**qed**

**also have** ... =  $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x, \ b]$   
 $(\mathfrak{p}_0[a, b] \cdot$   
 $\text{Uncurry}[x, \ a \ \otimes \ b]$   
 $(\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x, \ a \ \otimes \ b] \langle ?A, ?B \rangle))$

**using** *comp-assoc* **by** *simp*

**also have** ... =  $\text{Curry} \ (\text{exp } x \ a \ \otimes \ \text{exp } x \ b) \ x \ b \ (\mathfrak{p}_0[a, b] \cdot \langle ?A, ?B \rangle)$

**using** *assms AB Uncurry-Curry ide-exp tensor-preserves-ide* **by** *simp*

**also have** ... =  $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x, \ b]$   
 $(\text{eval } x \ b \cdot$   
 $\langle \mathfrak{p}_0[\text{exp } x \ a, \ \text{exp } x \ b] \cdot \mathfrak{p}_1[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x],$   
 $\mathfrak{p}_0[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x] \rangle)$

**using** *assms A B* **by** *fastforce*

**also have** ... =  $\text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x, \ b]$   
 $(\text{eval } x \ b \cdot (\mathfrak{p}_0[\text{exp } x \ a, \ \text{exp } x \ b] \otimes x) \cdot$   
 $\langle \mathfrak{p}_1[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x],$   
 $\mathfrak{p}_0[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x] \rangle)$

**proof** –

**have** *seq*  $\mathfrak{p}_0[\text{exp } x \ a, \ \text{exp } x \ b] \ \mathfrak{p}_1[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x]$

**using** *assms* **by** *auto*

**thus** *?thesis*

**using** *assms 2 3 prod-eq-tensor comp-ide-arr ECC.prod-tuple*  
**by** *metis*

**qed**

**also have** ... =  $\text{Curry} \ (\text{exp } x \ a \ \otimes \ \text{exp } x \ b) \ x \ b$   
 $(\text{Uncurry}[x, \ b] \ \mathfrak{p}_0[\text{exp } x \ a, \ \text{exp } x \ b])$

**proof** –

**have**  $(\mathfrak{p}_0[\text{exp } x \ a, \ \text{exp } x \ b] \otimes x) \cdot$   
 $\langle \mathfrak{p}_1[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x], \ \mathfrak{p}_0[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, \ x] \rangle =$   
 $\mathfrak{p}_0[\text{exp } x \ a, \ \text{exp } x \ b] \otimes x$

**using** *assms comp-arr-ide ECC.tuple-pr* **by** *auto*

**thus** *?thesis* **by** *simp*

**qed**

**also have** ... =  $\mathfrak{p}_0[\text{exp } x \ a, \ \text{exp } x \ b]$

**using** *assms Curry-Uncurry* **by** *simp*

**finally show** *?thesis* **by** *blast*

```

      qed
      ultimately show ?thesis by simp
    qed
    also have ... =  $\text{exp } x \ a \ \otimes \ \text{exp } x \ b$ 
      using assms ECC.tuple-pr by simp
    finally have  $\langle ?C, ?D \rangle \cdot \text{Curry}[\text{exp } x \ a \ \otimes \ \text{exp } x \ b, x, a \ \otimes \ b] \langle ?A, ?B \rangle =$ 
       $\text{exp } x \ a \ \otimes \ \text{exp } x \ b$ 
      by blast
    thus ?thesis
      using assms tensor-preserves-ide by simp
  qed
qed
thus isomorphic ( $\text{exp } x \ (a \ \otimes \ b)$ ) ( $\text{exp } x \ a \ \otimes \ \text{exp } x \ b$ )
  unfolding isomorphic-def
  using CD by blast
qed

end

end

```

## Chapter 2

# Enriched Categories

The notion of an enriched category [1] generalizes the concept of category by replacing the hom-sets of an ordinary category by objects of an arbitrary monoidal category  $M$ . The choice, for each object  $a$ , of a distinguished element  $id\ a : a \rightarrow a$  as an identity, is replaced by an arrow  $Id\ a : \mathcal{I} \rightarrow Hom\ a\ a$  of  $M$ . The composition operation is similarly replaced by a family of arrows  $Comp\ a\ b\ c : Hom\ B\ C \otimes Hom\ A\ B \rightarrow Hom\ A\ C$  of  $M$ . The identity and composition are required to satisfy unit and associativity laws which are expressed as commutative diagrams in  $M$ .

```
theory EnrichedCategory
imports ClosedMonoidalCategory
begin
```

```
  context monoidal-category
  begin
```

```
    abbreviation  $\iota'$  ( $\langle \iota^{-1} \rangle$ )
    where  $\iota' \equiv inv\ \iota$ 
```

```
  end
```

```
  context elementary-symmetric-monoidal-category
  begin
```

```
    lemma sym-unit:
    shows  $\iota \cdot s[\mathcal{I}, \mathcal{I}] = \iota$ 
    by (simp add:  $\iota$ -def unitor-coherence unitor-coincidence(2))
```

```
    lemma sym-inv-unit:
    shows  $s[\mathcal{I}, \mathcal{I}] \cdot inv\ \iota = inv\ \iota$ 
    using sym-unit
    by (metis MC.unit-is-iso arr-inv cod-inv comp-arr-dom comp-cod-arr)
```



*iso-cancel-left iso-is-arr)*

end

## 2.1 Basic Definitions

```

locale enriched-category =
  monoidal-category +
fixes Obj :: 'o set
and Hom :: 'o  $\Rightarrow$  'o  $\Rightarrow$  'a
and Id :: 'o  $\Rightarrow$  'a
and Comp :: 'o  $\Rightarrow$  'o  $\Rightarrow$  'o  $\Rightarrow$  'a
assumes ide-Hom [intro, simp]:  $\llbracket a \in \text{Obj}; b \in \text{Obj} \rrbracket \Longrightarrow \text{ide } (\text{Hom } a \ b)$ 
  and Id-in-hom [intro]:  $a \in \text{Obj} \Longrightarrow \llbracket \text{Id } a : \mathcal{I} \rightarrow \text{Hom } a \ a \rrbracket$ 
  and Comp-in-hom [intro]:  $\llbracket a \in \text{Obj}; b \in \text{Obj}; c \in \text{Obj} \rrbracket \Longrightarrow$ 
     $\llbracket \text{Comp } a \ b \ c : \text{Hom } b \ c \otimes \text{Hom } a \ b \rightarrow \text{Hom } a \ c \rrbracket$ 
  and Comp-Hom-Id:  $\llbracket a \in \text{Obj}; b \in \text{Obj} \rrbracket \Longrightarrow$ 
     $\text{Comp } a \ a \ b \cdot (\text{Hom } a \ b \otimes \text{Id } a) = \text{r}[\text{Hom } a \ b]$ 
  and Comp-Id-Hom:  $\llbracket a \in \text{Obj}; b \in \text{Obj} \rrbracket \Longrightarrow$ 
     $\text{Comp } a \ b \ b \cdot (\text{Id } b \otimes \text{Hom } a \ b) = \text{l}[\text{Hom } a \ b]$ 
  and Comp-assoc:  $\llbracket a \in \text{Obj}; b \in \text{Obj}; c \in \text{Obj}; d \in \text{Obj} \rrbracket \Longrightarrow$ 
     $\text{Comp } a \ b \ d \cdot (\text{Comp } b \ c \ d \otimes \text{Hom } a \ b) =$ 
     $\text{Comp } a \ c \ d \cdot (\text{Hom } c \ d \otimes \text{Comp } a \ b \ c) \cdot$ 
     $\text{a}[\text{Hom } c \ d, \text{Hom } b \ c, \text{Hom } a \ b]$ 

```

A functor from an enriched category  $A$  to an enriched category  $B$  consists of an object map  $F_o : \text{Obj}_A \rightarrow \text{Obj}_B$  and a map  $F_a$  that assigns to each pair of objects  $a \ b$  in  $\text{Obj}_A$  an arrow  $F_a \ a \ b : \text{Hom}_A \ a \ b \rightarrow \text{Hom}_B \ (F_o \ a) \ (F_o \ b)$  of the underlying monoidal category, subject to equations expressing that identities and composition are preserved.

```

locale enriched-functor =
  monoidal-category C T  $\alpha$   $\iota$  +
  A: enriched-category C T  $\alpha$   $\iota$  ObjA HomA IdA CompA +
  B: enriched-category C T  $\alpha$   $\iota$  ObjB HomB IdB CompB
for C :: 'm  $\Rightarrow$  'm  $\Rightarrow$  'm (infixr  $\langle \cdot \rangle$  55)
and T :: 'm  $\times$  'm  $\Rightarrow$  'm
and  $\alpha$  :: 'm  $\times$  'm  $\times$  'm  $\Rightarrow$  'm
and  $\iota$  :: 'm
and ObjA :: 'a set
and HomA :: 'a  $\Rightarrow$  'a  $\Rightarrow$  'm
and IdA :: 'a  $\Rightarrow$  'm
and CompA :: 'a  $\Rightarrow$  'a  $\Rightarrow$  'a  $\Rightarrow$  'm
and ObjB :: 'b set
and HomB :: 'b  $\Rightarrow$  'b  $\Rightarrow$  'm
and IdB :: 'b  $\Rightarrow$  'm
and CompB :: 'b  $\Rightarrow$  'b  $\Rightarrow$  'b  $\Rightarrow$  'm
and Fo :: 'a  $\Rightarrow$  'b
and Fa :: 'a  $\Rightarrow$  'a  $\Rightarrow$  'm +

```

**assumes extensionality:**  $a \notin \text{Obj}_A \vee b \notin \text{Obj}_A \implies F_a a b = \text{null}$   
**assumes preserves-Obj [intro]:**  $a \in \text{Obj}_A \implies F_o a \in \text{Obj}_B$   
**and preserves-Hom:**  $\llbracket a \in \text{Obj}_A; b \in \text{Obj}_A \rrbracket \implies$   
 $\quad \langle F_a a b : \text{Hom}_A a b \rightarrow \text{Hom}_B (F_o a) (F_o b) \rangle$   
**and preserves-Id:**  $a \in \text{Obj}_A \implies F_a a a \cdot \text{Id}_A a = \text{Id}_B (F_o a)$   
**and preserves-Comp:**  $\llbracket a \in \text{Obj}_A; b \in \text{Obj}_A; c \in \text{Obj}_A \rrbracket \implies$   
 $\quad \text{Comp}_B (F_o a) (F_o b) (F_o c) \cdot T (F_a b c, F_a a b) =$   
 $\quad F_a a c \cdot \text{Comp}_A a b c$

**locale fully-faithful-enriched-functor =**  
*enriched-functor +*  
**assumes locally-iso:**  $\llbracket a \in \text{Obj}_A; b \in \text{Obj}_A \rrbracket \implies \text{iso} (F_a a b)$

A natural transformation from an an enriched functor  $F = (F_o, F_a)$  to an enriched functor  $G = (G_o, G_a)$  consists of a map  $\tau$  that assigns to each object  $a \in \text{Obj}_A$  a “component at  $a$ ”, which is an arrow  $\tau a : \mathcal{I} \rightarrow \text{Hom}_B (F_o a) (G_o a)$ , subject to an equation that expresses the naturality condition.

**locale enriched-natural-transformation =**  
*monoidal-category C T  $\alpha$   $\iota$  +*  
*A: enriched-category C T  $\alpha$   $\iota$   $\text{Obj}_A \text{Hom}_A \text{Id}_A \text{Comp}_A +$*   
*B: enriched-category C T  $\alpha$   $\iota$   $\text{Obj}_B \text{Hom}_B \text{Id}_B \text{Comp}_B +$*   
*F: enriched-functor C T  $\alpha$   $\iota$*   
 $\quad \text{Obj}_A \text{Hom}_A \text{Id}_A \text{Comp}_A \text{Obj}_B \text{Hom}_B \text{Id}_B \text{Comp}_B F_o F_a +$   
*G: enriched-functor C T  $\alpha$   $\iota$*   
 $\quad \text{Obj}_A \text{Hom}_A \text{Id}_A \text{Comp}_A \text{Obj}_B \text{Hom}_B \text{Id}_B \text{Comp}_B G_o G_a$   
**for**  $C :: 'm \Rightarrow 'm \Rightarrow 'm$  (**infixr**  $\langle \cdot \rangle$  55)  
**and**  $T :: 'm \times 'm \Rightarrow 'm$   
**and**  $\alpha :: 'm \times 'm \times 'm \Rightarrow 'm$   
**and**  $\iota :: 'm$   
**and**  $\text{Obj}_A :: 'a \text{ set}$   
**and**  $\text{Hom}_A :: 'a \Rightarrow 'a \Rightarrow 'm$   
**and**  $\text{Id}_A :: 'a \Rightarrow 'm$   
**and**  $\text{Comp}_A :: 'a \Rightarrow 'a \Rightarrow 'a \Rightarrow 'm$   
**and**  $\text{Obj}_B :: 'b \text{ set}$   
**and**  $\text{Hom}_B :: 'b \Rightarrow 'b \Rightarrow 'm$   
**and**  $\text{Id}_B :: 'b \Rightarrow 'm$   
**and**  $\text{Comp}_B :: 'b \Rightarrow 'b \Rightarrow 'b \Rightarrow 'm$   
**and**  $F_o :: 'a \Rightarrow 'b$   
**and**  $F_a :: 'a \Rightarrow 'a \Rightarrow 'm$   
**and**  $G_o :: 'a \Rightarrow 'b$   
**and**  $G_a :: 'a \Rightarrow 'a \Rightarrow 'm$   
**and**  $\tau :: 'a \Rightarrow 'm +$   
**assumes extensionality:**  $a \notin \text{Obj}_A \implies \tau a = \text{null}$   
**and component-in-hom [intro]:**  $a \in \text{Obj}_A \implies \langle \tau a : \mathcal{I} \rightarrow \text{Hom}_B (F_o a) (G_o a) \rangle$   
**and naturality:**  $\llbracket a \in \text{Obj}_A; b \in \text{Obj}_A \rrbracket \implies$   
 $\quad \text{Comp}_B (F_o a) (F_o b) (G_o b) \cdot (\tau b \otimes F_a a b) \cdot \text{l}^{-1}[\text{Hom}_A a b] =$   
 $\quad \text{Comp}_B (F_o a) (G_o a) (G_o b) \cdot (G_a a b \otimes \tau a) \cdot \text{r}^{-1}[\text{Hom}_A a b]$

### 2.1.1 Self-Enrichment

**context** *elementary-closed-monoidal-category*  
**begin**

Every closed monoidal category  $M$  admits a structure of enriched category, where the exponentials in  $M$  itself serve as the “hom-objects” (cf. [1] Section 1.6). Essentially all the work in proving this theorem has already been done in *EnrichedCategoryBasics.ClosedMonoidalCategory*.

**interpretation** *closed-monoidal-category*  
**using** *is-closed-monoidal-category* **by** *blast*

**interpretation** *EC: enriched-category C T α ι ⟨Collect ide⟩ exp Id Comp*  
**using** *Id-in-hom Comp-in-hom Comp-unit-right Comp-unit-left Comp-assoc<sub>ECMC</sub>(2)*  
**by** *unfold-locales auto*

**theorem** *is-enriched-in-itself:*  
**shows** *enriched-category C T α ι (Collect ide) exp Id Comp*  
 ..

The following mappings define a bijection between  $hom\ a\ b$  and  $hom\ \mathcal{I}$  ( $exp\ a\ b$ ). These have functorial properties which are encountered repeatedly.

**definition** *UP* ( $\langle \cdot^\uparrow \rangle$  [100] 100)  
**where**  $t^\uparrow \equiv$  *if arr t then Curry[ $\mathcal{I}$ , dom t, cod t] (t · 1[dom t]) else null*

**definition** *DN*  
**where**  $DN\ a\ b\ t \equiv$  *if arr t then Uncurry[a, b] t · 1<sup>-1</sup>[a] else null*

**abbreviation** *DN'* ( $\langle \cdot^\downarrow[-, -] \rangle$  [100] 99)  
**where**  $t^\downarrow[a, b] \equiv DN\ a\ b\ t$

**lemma** *UP-DN:*  
**shows** [*intro*]:  $arr\ t \implies \langle t^\uparrow : \mathcal{I} \rightarrow exp\ (dom\ t)\ (cod\ t) \rangle$   
**and** [*intro*]:  $\llbracket ide\ a; ide\ b; \langle t : \mathcal{I} \rightarrow exp\ a\ b \rangle \rrbracket \implies \langle t^\downarrow[a, b] : a \rightarrow b \rangle$   
**and** [*simp*]:  $arr\ t \implies (t^\uparrow)^\downarrow[dom\ t, cod\ t] = t$   
**and** [*simp*]:  $\llbracket ide\ a; ide\ b; \langle t : \mathcal{I} \rightarrow exp\ a\ b \rangle \rrbracket \implies (t^\downarrow[a, b])^\uparrow = t$   
**using** *UP-def DN-def Uncurry-Curry Curry-Uncurry [of  $\mathcal{I}\ a\ b\ t$ ]*  
*comp-assoc comp-arr-dom*  
**by** *auto*

**lemma** *UP-simps* [*simp*]:  
**assumes**  $arr\ t$   
**shows**  $arr\ (t^\uparrow)$  **and**  $dom\ (t^\uparrow) = \mathcal{I}$  **and**  $cod\ (t^\uparrow) = exp\ (dom\ t)\ (cod\ t)$   
**using** *assms UP-DN* **by** *auto*

**lemma** *DN-simps* [*simp*]:  
**assumes**  $ide\ a$  **and**  $ide\ b$  **and**  $arr\ t$  **and**  $dom\ t = \mathcal{I}$  **and**  $cod\ t = exp\ a\ b$   
**shows**  $arr\ (t^\downarrow[a, b])$  **and**  $dom\ (t^\downarrow[a, b]) = a$  **and**  $cod\ (t^\downarrow[a, b]) = b$

**using** *assms UP-DN DN-def* **by** *auto*

**lemma** *UP-ide*:

**assumes** *ide a*

**shows**  $a^\uparrow = Id\ a$

**using** *assms Id-def comp-cod-arr UP-def* **by** *auto*

**lemma** *DN-Id*:

**assumes** *ide a*

**shows**  $(Id\ a)^\downarrow[a, a] = a$

**using** *assms Uncurry-Curry lunit-in-hom Id-def DN-def* **by** *auto*

**lemma** *UP-comp*:

**assumes** *seq t u*

**shows**  $(t \cdot u)^\uparrow = Comp\ (dom\ u)\ (cod\ u)\ (cod\ t) \cdot (t^\uparrow \otimes u^\uparrow) \cdot \iota^{-1}$

**proof** –

**have**  $Comp\ (dom\ u)\ (cod\ u)\ (cod\ t) \cdot (t^\uparrow \otimes u^\uparrow) \cdot \iota^{-1} =$   
 $(Curry[exp\ (cod\ u)\ (cod\ t) \otimes exp\ (dom\ u)\ (cod\ u),\ dom\ u,\ cod\ t]$   
 $(eval\ (cod\ u)\ (cod\ t) \cdot$   
 $(exp\ (cod\ u)\ (cod\ t) \otimes eval\ (dom\ u)\ (cod\ u)) \cdot$   
 $a[exp\ (cod\ u)\ (cod\ t),\ exp\ (dom\ u)\ (cod\ u),\ dom\ u] \cdot$   
 $(t^\uparrow \otimes u^\uparrow)) \cdot \iota^{-1}$

**unfolding** *Comp-def*

**using** *assms comp-assoc* **by** *simp*

**also have** ... =

$(Curry[\mathcal{I} \otimes \mathcal{I},\ dom\ u,\ cod\ t]$   
 $((eval\ (cod\ u)\ (cod\ t) \cdot$   
 $(exp\ (cod\ u)\ (cod\ t) \otimes eval\ (dom\ u)\ (cod\ u)) \cdot$   
 $a[exp\ (cod\ u)\ (cod\ t),\ exp\ (dom\ u)\ (cod\ u),\ dom\ u] \cdot$   
 $((t^\uparrow \otimes u^\uparrow) \otimes dom\ u))) \cdot \iota^{-1}$

**using** *assms*

*comp-Curry-arr*

[*of*  $dom\ u\ t^\uparrow \otimes u^\uparrow$

$\mathcal{I} \otimes \mathcal{I}\ exp\ (cod\ u)\ (cod\ t) \otimes exp\ (dom\ u)\ (cod\ u)$

$eval\ (cod\ u)\ (cod\ t) \cdot$

$(exp\ (cod\ u)\ (cod\ t) \otimes eval\ (dom\ u)\ (cod\ u)) \cdot$

$a[exp\ (cod\ u)\ (cod\ t),\ exp\ (dom\ u)\ (cod\ u),\ dom\ u]$

$cod\ t]$

**by** *fastforce*

**also have** ... =

$Curry[\mathcal{I} \otimes \mathcal{I},\ dom\ u,\ cod\ t]$   
 $(eval\ (cod\ u)\ (cod\ t) \cdot$   
 $((exp\ (cod\ u)\ (cod\ t) \otimes eval\ (dom\ u)\ (cod\ u)) \cdot$   
 $(t^\uparrow \otimes u^\uparrow \otimes dom\ u)) \cdot a[\mathcal{I}, \mathcal{I},\ dom\ u]) \cdot \iota^{-1}$

**using** *assms assoc-naturality* [*of*  $t^\uparrow\ u^\uparrow\ dom\ u]$  *comp-assoc* **by** *auto*

**also have** ... =

$Curry[\mathcal{I} \otimes \mathcal{I},\ dom\ u,\ cod\ t]$   
 $(eval\ (cod\ u)\ (cod\ t) \cdot$   
 $(exp\ (cod\ u)\ (cod\ t) \cdot t^\uparrow \otimes Uncurry[dom\ u,\ cod\ u]\ (u^\uparrow)) \cdot$

$a[\mathcal{I}, \mathcal{I}, \text{dom } u] \cdot \iota^{-1}$

**using** *assms comp-cod-arr UP-DN interchange by auto*

**also have** ... =

$$\begin{aligned} & \text{Curry}[\mathcal{I} \otimes \mathcal{I}, \text{dom } u, \text{cod } t] \\ & \quad (\text{eval } (\text{cod } u) (\text{cod } t) \cdot \\ & \quad \quad (\text{exp } (\text{cod } u) (\text{cod } t) \cdot t^\dagger \otimes u \cdot l[\text{dom } u]) \cdot \\ & \quad \quad a[\mathcal{I}, \mathcal{I}, \text{dom } u] \cdot \iota^{-1} \end{aligned}$$

**using** *assms Uncurry-Curry UP-def by auto*

**also have** ... =

$$\begin{aligned} & \text{Curry}[\mathcal{I} \otimes \mathcal{I}, \text{dom } u, \text{cod } t] \\ & \quad (\text{eval } (\text{cod } u) (\text{cod } t) \cdot \\ & \quad \quad (t^\dagger \otimes u \cdot l[\text{dom } u]) \cdot a[\mathcal{I}, \mathcal{I}, \text{dom } u] \cdot \iota^{-1} \end{aligned}$$

**using** *assms comp-cod-arr by auto*

**also have** ... =

$$\begin{aligned} & \text{Curry}[\mathcal{I} \otimes \mathcal{I}, \text{dom } u, \text{cod } t] \\ & \quad (\text{eval } (\text{cod } u) (\text{cod } t) \cdot \\ & \quad \quad ((t^\dagger \otimes \text{cod } u) \cdot (\mathcal{I} \otimes u \cdot l[\text{dom } u])) \cdot \\ & \quad \quad a[\mathcal{I}, \mathcal{I}, \text{dom } u] \cdot \iota^{-1} \end{aligned}$$

**using** *assms comp-arr-dom [of t^\dagger \mathcal{I} comp-cod-arr [of u \cdot l[dom u] cod u] interchange [of t^\dagger \mathcal{I} cod u u \cdot l[dom u]]*

**by** *auto*

**also have** ... =

$$\begin{aligned} & \text{Curry}[\mathcal{I}, \text{dom } u, \text{cod } t] \\ & \quad ((\text{eval } (\text{cod } u) (\text{cod } t) \cdot \\ & \quad \quad ((t^\dagger \otimes \text{cod } u) \cdot (\mathcal{I} \otimes u \cdot l[\text{dom } u])) \cdot \\ & \quad \quad a[\mathcal{I}, \mathcal{I}, \text{dom } u] \cdot (\iota^{-1} \otimes \text{dom } u)) \end{aligned}$$

**proof** –

**have**  $\langle \iota^{-1} : \mathcal{I} \rightarrow \mathcal{I} \otimes \mathcal{I} \rangle$

**using** *inv-in-hom unit-is-iso by blast*

**thus** *?thesis*

**using** *assms comp-Curry-arr by fastforce*

**qed**

**also have** ... =

$$\begin{aligned} & \text{Curry}[\mathcal{I}, \text{dom } u, \text{cod } t] \\ & \quad ((\text{Uncurry}[\text{cod } u, \text{cod } t] (t^\dagger)) \cdot (\mathcal{I} \otimes u \cdot l[\text{dom } u]) \cdot \\ & \quad \quad a[\mathcal{I}, \mathcal{I}, \text{dom } u] \cdot (\iota^{-1} \otimes \text{dom } u)) \end{aligned}$$

**using** *comp-assoc by simp*

**also have** ... =  $\text{Curry}[\mathcal{I}, \text{dom } u, \text{cod } t] (\text{Uncurry}[\text{cod } u, \text{cod } t] (t^\dagger) \cdot (\mathcal{I} \otimes u))$

**proof** –

**have**  $(\mathcal{I} \otimes u \cdot l[\text{dom } u]) \cdot a[\mathcal{I}, \mathcal{I}, \text{dom } u] \cdot (\iota^{-1} \otimes \text{dom } u) =$

$$((\mathcal{I} \otimes u) \cdot (\mathcal{I} \otimes l[\text{dom } u])) \cdot a[\mathcal{I}, \mathcal{I}, \text{dom } u] \cdot (\iota^{-1} \otimes \text{dom } u)$$

**using** *assms by auto*

**also have** ... =  $(\mathcal{I} \otimes u) \cdot (\mathcal{I} \otimes l[\text{dom } u]) \cdot a[\mathcal{I}, \mathcal{I}, \text{dom } u] \cdot (\iota^{-1} \otimes \text{dom } u)$

**using** *comp-assoc by simp*

**also have** ... =  $(\mathcal{I} \otimes u) \cdot (\mathcal{I} \otimes l[\text{dom } u]) \cdot (\mathcal{I} \otimes l^{-1}[\text{dom } u])$

**proof** –

**have**  $a[\mathcal{I}, \mathcal{I}, \text{dom } u] \cdot (\iota^{-1} \otimes \text{dom } u) = \mathcal{I} \otimes l^{-1}[\text{dom } u]$

**proof** –

**have**  $a[\mathcal{I}, \mathcal{I}, \text{dom } u] \cdot (\iota^{-1} \otimes \text{dom } u) =$

```

      inv ((ι ⊗ dom u) · a-1[ $\mathcal{I}$ ,  $\mathcal{I}$ , dom u])
    using assms inv-inv inv-comp [of a-1[ $\mathcal{I}$ ,  $\mathcal{I}$ , dom u] ι ⊗ dom u]
      inv-tensor [of ι dom u]
    by (metis ide-dom ide-is-iso ide-unity inv-ide iso-assoc iso-inv-iso
        iso-is-arr lunit-char(2) seqE tensor-preserves-iso triangle
        unit-is-iso unitor-coincidence(2))
    also have ... = inv (I ⊗ l[dom u])
      using assms lunit-char [of dom u] by auto
    also have ... = I ⊗ l-1[dom u]
      using assms inv-tensor by auto
    finally show ?thesis by blast
  qed
  thus ?thesis by simp
qed
also have ... = (I ⊗ u) · (I ⊗ dom u)
  using assms
  by (metis comp-ide-self comp-lunit-lunit'(1) dom-comp ideD(1)
      ide-dom ide-unity interchange)
also have ... = I ⊗ u
  using assms by blast
finally have (I ⊗ u · l[dom u]) · a[ $\mathcal{I}$ ,  $\mathcal{I}$ , dom u] · (ι-1 ⊗ dom u) = I ⊗ u
  by blast
thus ?thesis by argo
qed
also have ... = Curry[ $\mathcal{I}$ , dom u, cod t] ((t · l[cod u]) · (I ⊗ u))
  using assms Uncurry-Curry UP-def by auto
also have ... = Curry[ $\mathcal{I}$ , dom u, cod t] (t · u · l[dom u])
  using assms comp-assoc lunit-naturality by auto
also have ... = (t · u)↑
  using assms comp-assoc UP-def by simp
finally show ?thesis by simp
qed
end

```

## 2.2 Underlying Category, Functor, and Natural Transformation

### 2.2.1 Underlying Category

The underlying category (*cf.* [1] Section 1.3) of an enriched category has as its arrows from  $a$  to  $b$  the arrows  $\mathcal{I} \rightarrow \text{Hom } a \ b$  of  $M$  (*i.e.* the points of  $\text{Hom } a \ b$ ). The identity at  $a$  is  $\text{Id } a$ . The composition of arrows  $f$  and  $g$  is given by the formula:  $\text{Comp } a \ b \ c \cdot (g \otimes f) \cdot \iota^{-1}$ .

```

locale underlying-category =
  M: monoidal-category +
  A: enriched-category
begin

```

**sublocale** *concrete-category* *Obj*  $\langle \lambda a b. M.hom \mathcal{I} (Hom a b) \rangle \langle Id \rangle$   
 $\langle \lambda c b a g f. Comp a b c \cdot (g \otimes f) \cdot \iota^{-1} \rangle$

**proof**

**show**  $\bigwedge a. a \in Obj \implies Id a \in M.hom \mathcal{I} (Hom a a)$

**using** *A.Id-in-hom* **by** *blast*

**show** *1*:  $\bigwedge a b c f g.$

$\llbracket a \in Obj; b \in Obj; c \in Obj;$

$f \in M.hom \mathcal{I} (Hom a b); g \in M.hom \mathcal{I} (Hom b c) \rrbracket$

$\implies Comp a b c \cdot (g \otimes f) \cdot \iota^{-1} \in M.hom \mathcal{I} (Hom a c)$

**using** *A.Comp-in-hom* *M.inv-in-hom* *M.unit-is-iso* *M.comp-in-homI*  
*M.unit-in-hom*

**apply** *auto*[*1*]

**apply** (*intro* *M.comp-in-homI*)

**by** *auto*

**show**  $\bigwedge a b f. \llbracket a \in Obj; b \in Obj; f \in M.hom \mathcal{I} (Hom a b) \rrbracket$

$\implies Comp a a b \cdot (f \otimes Id a) \cdot \iota^{-1} = f$

**proof** –

**fix** *a b f*

**assume** *a*:  $a \in Obj$  **and** *b*:  $b \in Obj$  **and** *f*:  $f \in M.hom \mathcal{I} (Hom a b)$

**show**  $Comp a a b \cdot (f \otimes Id a) \cdot \iota^{-1} = f$

**proof** –

**have**  $Comp a a b \cdot (f \otimes Id a) \cdot \iota^{-1} = (Comp a a b \cdot (f \otimes Id a)) \cdot \iota^{-1}$

**using** *M.comp-assoc* **by** *simp*

**also have**  $\dots = (Comp a a b \cdot (Hom a b \otimes Id a) \cdot (f \otimes \mathcal{I})) \cdot \iota^{-1}$

**using** *a f M.comp-arr-dom* *M.comp-cod-arr* *A.Id-in-hom*

*M.in-homE* *M.interchange mem-Collect-eq*

**by** *metis*

**also have**  $\dots = (r[Hom a b] \cdot (f \otimes \mathcal{I})) \cdot \iota^{-1}$

**using** *a b f A.Comp-Hom-Id* *M.comp-assoc* **by** *metis*

**also have**  $\dots = (f \cdot r[\mathcal{I}]) \cdot \iota^{-1}$

**using** *f M.runit-naturality* **by** *fastforce*

**also have**  $\dots = f \cdot \iota \cdot \iota^{-1}$

**by** (*simp add: M.unitor-coincidence(2)* *M.comp-assoc*)

**also have**  $\dots = f$

**using** *f M.comp-arr-dom* *M.comp-arr-inv'* *M.unit-is-iso* **by** *auto*

**finally show**  $Comp a a b \cdot (f \otimes Id a) \cdot \iota^{-1} = f$  **by** *blast*

**qed**

**qed**

**show**  $\bigwedge a b f. \llbracket a \in Obj; b \in Obj; f \in M.hom \mathcal{I} (Hom a b) \rrbracket$

$\implies Comp a b b \cdot (Id b \otimes f) \cdot \iota^{-1} = f$

**proof** –

**fix** *a b f*

**assume** *a*:  $a \in Obj$  **and** *b*:  $b \in Obj$  **and** *f*:  $f \in M.hom \mathcal{I} (Hom a b)$

**show**  $Comp a b b \cdot (Id b \otimes f) \cdot \iota^{-1} = f$

**proof** –

**have**  $Comp a b b \cdot (Id b \otimes f) \cdot \iota^{-1} = (Comp a b b \cdot (Id b \otimes f)) \cdot \iota^{-1}$

**using** *M.comp-assoc* **by** *simp*

**also have**  $\dots = (Comp a b b \cdot (Id b \otimes Hom a b) \cdot (\mathcal{I} \otimes f)) \cdot \iota^{-1}$

```

using a b f M.comp-arr-dom M.comp-cod-arr A.Id-in-hom
M.in-homE M.interchange mem-Collect-eq
by metis
also have ... = (l[Hom a b] · (I ⊗ f)) · ι-1
using a b A.Comp-Id-Hom M.comp-assoc by metis
also have ... = (f · l[I]) · ι-1
using a b f M.lunit-naturality [of f] by auto
also have ... = f · ι · ι-1
by (simp add: M.unitor-coincidence(1) M.comp-assoc)
also have ... = f
using M.comp-arr-dom M.comp-arr-inv' M.unit-is-iso f by auto
finally show Comp a b b · (Id b ⊗ f) · ι-1 = f by blast
qed
qed
show ∧ a b c d f g h.
  [a ∈ Obj; b ∈ Obj; c ∈ Obj; d ∈ Obj;
   f ∈ M.hom I (Hom a b); g ∈ M.hom I (Hom b c);
   h ∈ M.hom I (Hom c d)]
  ⇒ Comp a c d · (h ⊗ Comp a b c · (g ⊗ f) · ι-1) · ι-1 =
     Comp a b d · (Comp b c d · (h ⊗ g) · ι-1 ⊗ f) · ι-1
proof -
  fix a b c d f g h
  assume a: a ∈ Obj and b: b ∈ Obj and c: c ∈ Obj and d: d ∈ Obj
  assume f: f ∈ M.hom I (Hom a b) and g: g ∈ M.hom I (Hom b c)
  and h: h ∈ M.hom I (Hom c d)
  have Comp a c d · (h ⊗ Comp a b c · (g ⊗ f) · ι-1) · ι-1 =
    Comp a c d ·
      ((Hom c d ⊗ Comp a b c) · (h ⊗ (g ⊗ f) · ι-1)) · ι-1
  using a b c d f g h 1 M.interchange A.ide-Hom M.comp-ide-arr M.comp-cod-arr
    M.in-homE mem-Collect-eq
  by metis
  also have ... = Comp a c d ·
    ((Hom c d ⊗ Comp a b c) ·
     (a[Hom c d, Hom b c, Hom a b] ·
      a-1[Hom c d, Hom b c, Hom a b])) ·
    (h ⊗ (g ⊗ f) · ι-1) · ι-1
  proof -
    have (Hom c d ⊗ Comp a b c) ·
      (a[Hom c d, Hom b c, Hom a b] ·
       a-1[Hom c d, Hom b c, Hom a b]) =
      Hom c d ⊗ Comp a b c
    using a b c d
  by (metis A.Comp-in-hom A.ide-Hom M.comp-arr-ide
    M.comp-assoc-assoc'(1) M.ide-in-hom M.interchange M.seqI'
    M.tensor-preserves-ide)
  thus ?thesis
  using M.comp-assoc by force
qed
also have ... = (Comp a c d · (Hom c d ⊗ Comp a b c) ·

```



$$\begin{aligned}
& a[\text{Hom } c \ d, \text{Hom } b \ c, \text{Hom } a \ b]) \cdot \\
& (\text{a}^{-1}[\text{Hom } c \ d, \text{Hom } b \ c, \text{Hom } a \ b] \cdot \\
& (h \otimes (g \otimes f) \cdot \iota^{-1})) \cdot \\
& \iota^{-1}
\end{aligned}$$

**using** *M.comp-assoc* **by** *auto*  
**also have** ... = (*Comp a b d* · (*Comp b c d* ⊗ *Hom a b*)) ·  
( $\text{a}^{-1}[\text{Hom } c \ d, \text{Hom } b \ c, \text{Hom } a \ b] \cdot (h \otimes (g \otimes f) \cdot \iota^{-1})) \cdot \iota^{-1}$ )  
**using** *a b c d A.Comp-assoc* **by** *auto*  
**also have** ... = (*Comp a b d* · (*Comp b c d* ⊗ *Hom a b*)) ·  
( $\text{a}^{-1}[\text{Hom } c \ d, \text{Hom } b \ c, \text{Hom } a \ b] \cdot (h \otimes (g \otimes f))) \cdot$   
( $\mathcal{I} \otimes \iota^{-1}) \cdot \iota^{-1}$ )  
**proof** –  
**have**  $h \otimes (g \otimes f) \cdot \iota^{-1} = (h \otimes (g \otimes f)) \cdot (\mathcal{I} \otimes \iota^{-1})$   
**proof** –  
**have** *M.seq h I*  
**using** *h by auto*  
**moreover have** *M.seq (g ⊗ f) ι<sup>-1</sup>*  
**using** *f g M.inv-in-hom M.unit-is-iso* **by** *blast*  
**ultimately show** *?thesis*  
**using** *a b c d f g h M.interchange M.comp-arr-ide M.ide-unity* **by** *metis*  
**qed**  
**thus** *?thesis*  
**using** *M.comp-assoc* **by** *simp*  
**qed**  
**also have** ... = (*Comp a b d* · (*Comp b c d* ⊗ *Hom a b*)) ·  
( $(h \otimes g) \otimes f \cdot \text{a}^{-1}[\mathcal{I}, \mathcal{I}, \mathcal{I}] \cdot (\mathcal{I} \otimes \iota^{-1}) \cdot \iota^{-1}$ )  
**using** *f g h M.assoc'-naturality*  
**by** (*metis M.comp-assoc M.in-homE mem-Collect-eq*)  
**also have** ... = (*Comp a b d* · (*Comp b c d* ⊗ *Hom a b*)) ·  
( $((h \otimes g) \otimes f) \cdot (\iota^{-1} \otimes \mathcal{I})) \cdot \iota^{-1}$ )  
**proof** –  
**have**  $\text{a}^{-1}[\mathcal{I}, \mathcal{I}, \mathcal{I}] \cdot (\mathcal{I} \otimes \iota^{-1}) \cdot \iota^{-1} = (\iota^{-1} \otimes \mathcal{I}) \cdot \iota^{-1}$   
**using** *M.untor-coincidence*  
**by** (*metis (full-types) M.L.preserves-inv M.L.preserves-iso*  
*M.R.preserves-inv M.arrI M.arr-tensor M.comp-assoc M.ideD(1)*  
*M.ide-unity M.inv-comp M.iso-assoc M.unit-in-hom-ax*  
*M.unit-is-iso M.unit-triangle(1)*)  
**thus** *?thesis*  
**using** *M.comp-assoc* **by** *simp*  
**qed**  
**also have** ... = *Comp a b d* ·  
( $(\text{Comp } b \ c \ d \otimes \text{Hom } a \ b) \cdot ((h \otimes g) \cdot \iota^{-1} \otimes f)) \cdot \iota^{-1}$ )  
**proof** –  
**have**  $((h \otimes g) \otimes f) \cdot (\iota^{-1} \otimes \mathcal{I}) = (h \otimes g) \cdot \iota^{-1} \otimes f$   
**proof** –  
**have** *M.seq (h ⊗ g) ι<sup>-1</sup>*  
**using** *g h M.inv-in-hom M.unit-is-iso* **by** *blast*  
**moreover have** *M.seq f I*  
**using** *M.ide-in-hom M.ide-unity f* **by** *blast*

```

ultimately show ?thesis
  using f g h M.interchange M.comp-arr-ide M.ide-unity by metis
qed
thus ?thesis
  using M.comp-assoc by auto
qed
also have ... = Comp a b d · (Comp b c d · (h ⊗ g) · ι-1 ⊗ f) · ι-1
  using b c d f g h 1 M.in-homE mem-Collect-eq M.comp-cod-arr
  M.interchange A.ide-Hom M.comp-ide-arr
  by metis
finally show Comp a c d · (h ⊗ Comp a b c · (g ⊗ f) · ι-1) · ι-1 =
  Comp a b d · (Comp b c d · (h ⊗ g) · ι-1 ⊗ f) · ι-1
  by blast
qed
qed

```

```

abbreviation comp (infixr ‹·› 55)
where comp ≡ COMP

```

```

lemma hom-char:
assumes a ∈ Obj and b ∈ Obj
shows hom (MkIde a) (MkIde b) = MkArr a b ‹ M.hom ℐ (Hom a b) ›
proof
show hom (MkIde a) (MkIde b) ⊆ MkArr a b ‹ M.hom ℐ (Hom a b) ›
proof
fix t
assume t: t ∈ hom (MkIde a) (MkIde b)
have t = MkArr a b (Map t)
  using t MkArr-Map dom-char cod-char by fastforce
moreover have Map t ∈ M.hom ℐ (Hom a b)
  using t arr-char dom-char cod-char by fastforce
ultimately show t ∈ MkArr a b ‹ M.hom ℐ (Hom a b) › by simp
qed
show MkArr a b ‹ M.hom ℐ (Hom a b) › ⊆ hom (MkIde a) (MkIde b)
  using assms MkArr-in-hom by blast
qed

```

end

## 2.2.2 Underlying Functor

The underlying functor of an enriched functor  $F : A \rightarrow B$  takes an arrow  $\langle\langle f : a \rightarrow a' \rangle\rangle$  of the underlying category  $A_0$  (i.e. an arrow  $\langle\mathcal{I} \rightarrow \text{Hom } a \ a'\rangle$  of  $M$ ) to the arrow  $\langle\langle F_a \ a \ a' \cdot f : F_o \ a \rightarrow F_o \ a' \rangle\rangle$  of  $B_0$  (i.e. the arrow  $\langle\langle F_a \ a \ a' \cdot f : \mathcal{I} \rightarrow \text{Hom } (F_o \ a) \ (F_o \ a') \rangle\rangle$  of  $M$ ).

```

locale underlying-functor =
  enriched-functor
begin

```

**sublocale**  $A_0$ : *underlying-category*  $C$   $T$   $\alpha$   $\iota$   $Obj_A$   $Hom_A$   $Id_A$   $Comp_A$  ..  
**sublocale**  $B_0$ : *underlying-category*  $C$   $T$   $\alpha$   $\iota$   $Obj_B$   $Hom_B$   $Id_B$   $Comp_B$  ..

**notation**  $A_0.comp$  (**infixr**  $\langle \cdot_{A_0} \rangle$  55)

**notation**  $B_0.comp$  (**infixr**  $\langle \cdot_{B_0} \rangle$  55)

**definition**  $map_0$

**where**  $map_0 f =$  (if  $A_0.arr f$   
then  $B_0.MkArr (F_o (A_0.Dom f)) (F_o (A_0.Cod f))$   
 $(F_a (A_0.Dom f) (A_0.Cod f) \cdot A_0.Map f)$   
else  $B_0.null$ )

**sublocale functor**  $A_0.comp$   $B_0.comp$   $map_0$

**proof**

**fix**  $f$

**show**  $\neg A_0.arr f \implies map_0 f = B_0.null$

**using**  $map_0-def$  **by**  $simp$

**show**  $1: \bigwedge f. A_0.arr f \implies B_0.arr (map_0 f)$

**proof** –

**fix**  $f$

**assume**  $f: A_0.arr f$

**have**  $B_0.arr (B_0.MkArr (F_o (A_0.Dom f)) (F_o (A_0.Cod f))$   
 $(F_a (A_0.Dom f) (A_0.Cod f) \cdot A_0.Map f))$

**using**  $f$  *preserves-Hom*  $A_0.Dom-in-Obj$   $A_0.Cod-in-Obj$   $A_0.arrE$

**by** ( $metis$  ( $mono-tags$ ,  $lifting$ )  $B_0.arr-MkArr$   $comp-in-homI$   
 $mem-Collect-eq$  *preserves-Obj*)

**thus**  $B_0.arr (map_0 f)$

**using**  $f$   $map_0-def$  **by**  $simp$

**qed**

**show**  $A_0.arr f \implies B_0.dom (map_0 f) = map_0 (A_0.dom f)$

**using**  $1$   $A_0.dom-char$   $B_0.dom-char$  *preserves-Id*  $A_0.arr-dom-iff-arr$   
 $map_0-def$   $A_0.Dom-in-Obj$

**by** *auto*

**show**  $A_0.arr f \implies B_0.cod (map_0 f) = map_0 (A_0.cod f)$

**using**  $1$   $A_0.cod-char$   $B_0.cod-char$  *preserves-Id*  $A_0.arr-cod-iff-arr$   
 $map_0-def$   $A_0.Cod-in-Obj$

**by** *auto*

**fix**  $g$

**assume**  $fg: A_0.seq g f$

**show**  $map_0 (g \cdot_{A_0} f) = map_0 g \cdot_{B_0} map_0 f$

**proof** –

**have**  $B_0.MkArr (F_o (A_0.Dom (g \cdot_{A_0} f))) (F_o (B_0.Cod (g \cdot_{A_0} f)))$   
 $(F_a (A_0.Dom (g \cdot_{A_0} f))$

$(B_0.Cod (g \cdot_{A_0} f)) \cdot B_0.Map (g \cdot_{A_0} f)) =$

$B_0.MkArr (F_o (A_0.Dom g)) (F_o (B_0.Cod g))$

$(F_a (A_0.Dom g) (B_0.Cod g) \cdot B_0.Map g) \cdot_{B_0}$

$B_0.MkArr (F_o (A_0.Dom f)) (F_o (B_0.Cod f))$

$(F_a (A_0.Dom f) (B_0.Cod f) \cdot B_0.Map f)$

**proof** –

**have** 2:  $B_0.arr (B_0.MkArr (F_o (A_0.Dom f)) (F_o (A_0.Dom g))$   
 $(F_a (A_0.Dom f) (A_0.Cod f) \cdot A_0.Map f))$   
**using** *fg 1 A<sub>0</sub>.seq-char map<sub>0</sub>-def* **by** *auto*  
**have** 3:  $B_0.arr (B_0.MkArr (F_o (A_0.Dom g)) (F_o (A_0.Cod g))$   
 $(F_a (A_0.Dom g) (A_0.Cod g) \cdot A_0.Map g))$   
**using** *fg 1 A<sub>0</sub>.seq-char map<sub>0</sub>-def* **by** *metis*  
**have**  $B_0.MkArr (F_o (A_0.Dom g)) (F_o (B_0.Cod g))$   
 $(F_a (A_0.Dom g) (B_0.Cod g) \cdot B_0.Map g) \cdot_{B_0}$   
 $B_0.MkArr (F_o (A_0.Dom f)) (F_o (B_0.Cod f))$   
 $(F_a (A_0.Dom f) (B_0.Cod f) \cdot B_0.Map f) =$   
 $B_0.MkArr (F_o (A_0.Dom f)) (F_o (A_0.Cod g))$   
 $(Comp_B (F_o (A_0.Dom f)) (F_o (A_0.Dom g)) (F_o (A_0.Cod g))) \cdot$   
 $(F_a (A_0.Dom g) (A_0.Cod g) \cdot A_0.Map g \otimes$   
 $F_a (A_0.Dom f) (A_0.Cod f) \cdot A_0.Map f) \cdot$   
 $\iota^{-1})$   
**using** *fg 2 3 A<sub>0</sub>.seq-char B<sub>0</sub>.comp-MkArr* **by** *simp*  
**moreover**  
**have**  $Comp_B (F_o (A_0.Dom f)) (F_o (A_0.Dom g)) (F_o (A_0.Cod g)) \cdot$   
 $(F_a (A_0.Dom g) (A_0.Cod g) \cdot A_0.Map g \otimes$   
 $F_a (A_0.Dom f) (A_0.Cod f) \cdot A_0.Map f) \cdot \iota^{-1} =$   
 $F_a (A_0.Dom (g \cdot_{A_0} f)) (B_0.Cod (g \cdot_{A_0} f)) \cdot B_0.Map (g \cdot_{A_0} f)$   
**proof** –  
**have**  $Comp_B (F_o (A_0.Dom f)) (F_o (A_0.Dom g)) (F_o (A_0.Cod g)) \cdot$   
 $(F_a (A_0.Dom g) (A_0.Cod g) \cdot A_0.Map g \otimes$   
 $F_a (A_0.Dom f) (A_0.Cod f) \cdot A_0.Map f) \cdot \iota^{-1} =$   
 $Comp_B (F_o (A_0.Dom f)) (F_o (A_0.Dom g)) (F_o (A_0.Cod g)) \cdot$   
 $((F_a (A_0.Dom g) (A_0.Cod g) \otimes F_a (A_0.Dom f) (A_0.Cod f))) \cdot$   
 $(A_0.Map g \otimes A_0.Map f) \cdot \iota^{-1}$   
**using** *fg preserves-Hom*  
*interchange [of F<sub>a</sub> (A<sub>0</sub>.Dom g) (A<sub>0</sub>.Cod g) A<sub>0</sub>.Map g*  
 $F_a (A_0.Dom f) (A_0.Cod f) A_0.Map f]$   
**by** (*metis A<sub>0</sub>.arrE A<sub>0</sub>.seqE seqI' mem-Collect-eq*)  
**also have** ... =  
 $(Comp_B (F_o (A_0.Dom f)) (F_o (A_0.Dom g)) (F_o (A_0.Cod g)) \cdot$   
 $(F_a (A_0.Dom g) (A_0.Cod g) \otimes F_a (A_0.Dom f) (A_0.Cod f))) \cdot$   
 $(A_0.Map g \otimes A_0.Map f) \cdot \iota^{-1}$   
**using** *comp-assoc* **by** *auto*  
**also have** ... =  $(F_a (A_0.Dom f) (B_0.Cod g) \cdot$   
 $Comp_A (A_0.Dom f) (A_0.Dom g) (B_0.Cod g)) \cdot$   
 $(A_0.Map g \otimes A_0.Map f) \cdot \iota^{-1}$   
**using** *fg A<sub>0</sub>.seq-char preserves-Comp A<sub>0</sub>.Dom-in-Obj A<sub>0</sub>.Cod-in-Obj*  
**by** *auto*  
**also have** ... =  $F_a (A_0.Dom (g \cdot_{A_0} f)) (B_0.Cod (g \cdot_{A_0} f)) \cdot$   
 $Comp_A (A_0.Dom f) (A_0.Dom g) (B_0.Cod g) \cdot$   
 $(A_0.Map g \otimes A_0.Map f) \cdot \iota^{-1}$   
**using** *fg comp-assoc A<sub>0</sub>.seq-char* **by** *simp*  
**also have** ... =  $F_a (A_0.Dom (g \cdot_{A_0} f)) (B_0.Cod (g \cdot_{A_0} f)) \cdot$   
 $B_0.Map (g \cdot_{A_0} f)$   
**using** *A<sub>0</sub>.Map-comp A<sub>0</sub>.seq-char fg* **by** *presburger*

```

    finally show ?thesis by blast
  qed
  ultimately show ?thesis
    using A0.seq-char fg by auto
  qed
  thus ?thesis
    using fg map0-def B0.comp-MkArr by auto
  qed
qed

```

```

proposition is-functor:
shows functor A0.comp B0.comp map0
..

```

end

### 2.2.3 Underlying Natural Transformation

The natural transformation underlying an enriched natural transformation  $\tau$  has components that are essentially those of  $\tau$ , except that we have to bother ourselves about coercions between types.

```

locale underlying-natural-transformation =
  enriched-natural-transformation
begin

```

```

sublocale A0: underlying-category C T  $\alpha$   $\iota$  ObjA HomA IdA CompA ..
sublocale B0: underlying-category C T  $\alpha$   $\iota$  ObjB HomB IdB CompB ..
sublocale F0: underlying-functor C T  $\alpha$   $\iota$ 
  ObjA HomA IdA CompA ObjB HomB IdB CompB Fo Fa ..
sublocale G0: underlying-functor C T  $\alpha$   $\iota$ 
  ObjA HomA IdA CompA ObjB HomB IdB CompB Go Ga ..

```

```

definition mapobj

```

```

where mapobj a  $\equiv$ 
  B0.MkArr (B0.Dom (F0.map0 a)) (B0.Dom (G0.map0 a))
  ( $\tau$  (A0.Dom a))

```

```

sublocale  $\tau$ : NaturalTransformation.transformation-by-components
  A0.comp B0.comp F0.map0 G0.map0 mapobj

```

```

proof

```

```

  show  $\bigwedge a. A_0.ide\ a \implies B_0.in-hom\ (map_{obj}\ a)\ (F_0.map_0\ a)\ (G_0.map_0\ a)$ 

```

```

    unfolding mapobj-def

```

```

    using A0.Dom-in-Obj B0.ide-charCC F0.map0-def G0.map0-def

```

```

      F0.preserves-ide G0.preserves-ide component-in-hom

```

```

    by auto

```

```

  show  $\bigwedge f. A_0.arr\ f \implies$ 

```

```

    mapobj (A0.cod f)  $\cdot_{B_0}$  F0.map0 f =

```

```

    G0.map0 f  $\cdot_{B_0}$  mapobj (A0.dom f)

```

```

proof –

```

```

fix f
assume f: A0.arr f
show mapobj (A0.cod f) ·B0 F0.map0 f =
      G0.map0 f ·B0 mapobj (A0.dom f)
proof (intro B0.arr-eqI)
  show 1: B0.seq (mapobj (A0.cod f)) (F0.map0 f)
    using A0.ide-cod
      ⟨∧ a. A0.ide a ⇒
        B0.in-hom (mapobj a) (F0.map0 a) (G0.map0 a)⟩ f
    by blast
  show 2: B0.seq (G0.map0 f) (mapobj (A0.dom f))
    using A0.ide-dom
      ⟨∧ a. A0.ide a ⇒
        B0.in-hom (mapobj a) (F0.map0 a) (G0.map0 a)⟩ f
    by blast
  show B0.Dom (mapobj (A0.cod f) ·B0 F0.map0 f) =
      B0.Dom (G0.map0 f ·B0 mapobj (A0.dom f))
    using f 1 2 B0.comp-char [of mapobj (A0.cod f) F0.map0 f]
      B0.comp-char [of G0.map0 f mapobj (A0.dom f)]
      F0.map0-def G0.map0-def mapobj-def
    by simp
  show B0.Cod (mapobj (A0.cod f) ·B0 F0.map0 f) =
      B0.Cod (G0.map0 f ·B0 mapobj (A0.dom f))
    using f 1 2 B0.comp-char [of mapobj (A0.cod f) F0.map0 f]
      B0.comp-char [of G0.map0 f mapobj (A0.dom f)]
      F0.map0-def G0.map0-def mapobj-def
    by simp
  show B0.Map (mapobj (A0.cod f) ·B0 F0.map0 f) =
      B0.Map (G0.map0 f ·B0 mapobj (A0.dom f))
  proof –
    have CompB (Fo (A0.Dom f)) (Fo (A0.Cod f)) (Go (A0.Cod f)) ·
      (τ (A0.Cod f) ⊗ Fa (A0.Dom f) (A0.Cod f) · A0.Map f) · ι-1 =
      CompB (Fo (A0.Dom f)) (Go (A0.Dom f)) (Go (A0.Cod f)) ·
      (Ga (A0.Dom f) (A0.Cod f) · A0.Map f ⊗ τ (A0.Dom f)) · ι-1
    proof –
      have CompB (Fo (A0.Dom f)) (Fo (A0.Cod f)) (Go (A0.Cod f)) ·
        (τ (A0.Cod f) ⊗ Fa (A0.Dom f) (A0.Cod f) · A0.Map f) · ι-1 =
        CompB (Fo (A0.Dom f)) (Fo (A0.Cod f)) (Go (A0.Cod f)) ·
        ((τ (A0.Cod f) ⊗ Fa (A0.Dom f) (A0.Cod f)) · (I ⊗ A0.Map f)) ·
          ι-1
      proof –
        have τ (A0.Cod f) ⊗ Fa (A0.Dom f) (A0.Cod f) · A0.Map f =
          (τ (A0.Cod f) ⊗ Fa (A0.Dom f) (A0.Cod f)) · (I ⊗ A0.Map f)
        proof –
          have seq (τ (A0.Cod f)) I
            using f seqI component-in-hom
            by (metis (no-types, lifting) A0.Cod-in-Obj ide-char
              ide-unity in-homE)
          moreover have seq (Fa (A0.Dom f) (B0.Cod f)) (B0.Map f)

```

**using**  $f$   $A_0$ .Map-in-Hom  $A_0$ .Cod-in-Obj  $A_0$ .Dom-in-Obj  
 $F$ .preserves-Hom in-homE  
**by** blast  
**ultimately show** ?thesis  
**using**  $f$  component-in-hom interchange comp-arr-dom **by** auto  
**qed**  
**thus** ?thesis **by** simp  
**qed**  
**also have** ... =  

$$\text{Comp}_B (F_o (A_0.\text{Dom } f)) (F_o (A_0.\text{Cod } f)) (G_o (A_0.\text{Cod } f)) \cdot$$

$$((\tau (B_0.\text{Cod } f) \otimes F_a (A_0.\text{Dom } f) (B_0.\text{Cod } f)) \cdot$$

$$(l^{-1}[\text{Hom}_A (A_0.\text{Dom } f) (B_0.\text{Cod } f)] \cdot$$

$$l[\text{Hom}_A (A_0.\text{Dom } f) (B_0.\text{Cod } f)]) \cdot$$

$$(\mathcal{I} \otimes B_0.\text{Map } f)) \cdot \iota^{-1}$$
**proof** –  
**have**  $(l^{-1}[\text{Hom}_A (A_0.\text{Dom } f) (B_0.\text{Cod } f)] \cdot$   
 $l[\text{Hom}_A (A_0.\text{Dom } f) (B_0.\text{Cod } f)]) \cdot$   
 $(\mathcal{I} \otimes B_0.\text{Map } f) =$   
 $\mathcal{I} \otimes B_0.\text{Map } f$   
**using**  $f$  comp-lunit-lunit'(2)  
**by** (metis (no-types, lifting) A.ide-Hom  $A_0$ .arrE comp-cod-arr  
comp-ide-self ideD(1) ide-unity interchange in-homE  
mem-Collect-eq)  
**thus** ?thesis **by** simp  
**qed**  
**also have** ... =  

$$(\text{Comp}_B (F_o (A_0.\text{Dom } f)) (F_o (B_0.\text{Cod } f)) (G_o (B_0.\text{Cod } f)) \cdot$$

$$(\tau (B_0.\text{Cod } f) \otimes F_a (A_0.\text{Dom } f) (B_0.\text{Cod } f)) \cdot$$

$$l^{-1}[\text{Hom}_A (A_0.\text{Dom } f) (B_0.\text{Cod } f)]) \cdot$$

$$l[\text{Hom}_A (A_0.\text{Dom } f) (B_0.\text{Cod } f)] \cdot (\mathcal{I} \otimes B_0.\text{Map } f)) \cdot \iota^{-1}$$
**using** comp-assoc **by** simp  
**also have** ... =  

$$\text{Comp}_B (F_o (A_0.\text{Dom } f)) (G_o (A_0.\text{Dom } f)) (G_o (B_0.\text{Cod } f)) \cdot$$

$$(G_a (A_0.\text{Dom } f) (B_0.\text{Cod } f) \otimes \tau (A_0.\text{Dom } f)) \cdot$$

$$r^{-1}[\text{Hom}_A (A_0.\text{Dom } f) (B_0.\text{Cod } f)] \cdot$$

$$(l[\text{Hom}_A (A_0.\text{Dom } f) (B_0.\text{Cod } f)] \cdot (\mathcal{I} \otimes B_0.\text{Map } f)) \cdot \iota^{-1}$$
**using**  $f$   $A_0$ .Cod-in-Obj  $A_0$ .Dom-in-Obj naturality comp-assoc **by** simp  
**also have** ... =  

$$\text{Comp}_B (F_o (A_0.\text{Dom } f)) (G_o (A_0.\text{Dom } f)) (G_o (B_0.\text{Cod } f)) \cdot$$

$$(G_a (A_0.\text{Dom } f) (B_0.\text{Cod } f) \otimes \tau (A_0.\text{Dom } f)) \cdot$$

$$r^{-1}[\text{Hom}_A (A_0.\text{Dom } f) (B_0.\text{Cod } f)] \cdot (B_0.\text{Map } f \cdot l[\mathcal{I}]) \cdot \iota^{-1}$$
**using**  $f$  lunit-naturality  $A_0$ .Map-in-Hom **by** force  
**also have** ... =  

$$\text{Comp}_B (F_o (A_0.\text{Dom } f)) (G_o (A_0.\text{Dom } f)) (G_o (B_0.\text{Cod } f)) \cdot$$

$$(G_a (A_0.\text{Dom } f) (B_0.\text{Cod } f) \otimes \tau (A_0.\text{Dom } f)) \cdot$$

$$r^{-1}[\text{Hom}_A (A_0.\text{Dom } f) (B_0.\text{Cod } f)] \cdot B_0.\text{Map } f$$
**proof** –  
**have**  $\iota \cdot \iota^{-1} = \mathcal{I}$   
**using** comp-arr-inv' unit-is-iso **by** blast

**moreover have**  $\langle\langle B_0.Map\ f : \mathcal{I} \rightarrow Hom_A (A_0.Dom\ f) (B_0.Cod\ f) \rangle\rangle$   
**using**  $f\ A_0.Map\text{-in-Hom}$  **by** *blast*  
**ultimately show** *?thesis*  
**using**  $f\ comp\text{-arr-dom}\ unitor\text{-coincidence}(1)\ comp\text{-assoc}$  **by** *auto*  
**qed**  
**also have** ... =  

$$Comp_B (F_o (A_0.Dom\ f)) (G_o (A_0.Dom\ f)) (G_o (B_0.Cod\ f)) \cdot$$

$$(G_a (A_0.Dom\ f) (B_0.Cod\ f) \otimes \tau (A_0.Dom\ f)) \cdot$$

$$(B_0.Map\ f \otimes \mathcal{I}) \cdot r^{-1}[\mathcal{I}]$$
**using**  $f\ runit'\text{-naturality}\ A_0.Map\text{-in-Hom}$  **by** *force*  
**also have** ... =  

$$Comp_B (F_o (A_0.Dom\ f)) (G_o (A_0.Dom\ f)) (G_o (B_0.Cod\ f)) \cdot$$

$$((G_a (A_0.Dom\ f) (B_0.Cod\ f) \otimes \tau (A_0.Dom\ f)) \cdot$$

$$(B_0.Map\ f \otimes \mathcal{I})) \cdot \iota^{-1}$$
**using** *unitor-coincidence comp-assoc* **by** *simp*  
**also have** ... =  

$$Comp_B (F_o (A_0.Dom\ f)) (G_o (A_0.Dom\ f)) (G_o (B_0.Cod\ f)) \cdot$$

$$(G_a (A_0.Dom\ f) (B_0.Cod\ f) \cdot A_0.Map\ f \otimes \tau (A_0.Dom\ f)) \cdot \iota^{-1}$$
**proof** –  
**have** *seq*  $(G_a (A_0.Dom\ f) (B_0.Cod\ f)) (B_0.Map\ f)$   
**using**  $f\ A_0.Map\text{-in-Hom}\ A_0.Cod\text{-in-Obj}\ A_0.Dom\text{-in-Obj}\ G.preserves\text{-Hom}$   
**by** *fast*  
**moreover have** *seq*  $(\tau (A_0.Dom\ f)) \mathcal{I}$   
**using**  $f\ seqI\ component\text{-in-hom}$   
**by** *(metis (no-types, lifting) A\_0.Dom-in-Obj ide-char ide-unity in-homE)*  
**ultimately show** *?thesis*  
**using**  $f\ comp\text{-arr-dom}\ interchange$  **by** *auto*  
**qed**  
**finally show** *?thesis* **by** *simp*  
**qed**  
**thus** *?thesis*  
**using**  $f\ 1\ 2\ B_0.comp\text{-char}\ [of\ map_{obj}\ (A_0.cod\ f)\ F_0.map_0\ f]$   
 $B_0.comp\text{-char}\ [of\ G_0.map_0\ f\ map_{obj}\ (A_0.dom\ f)]$   
 $F_0.map_0\text{-def}\ G_0.map_0\text{-def}\ map_{obj}\text{-def}$   
**by** *simp*  
**qed**  
**qed**  
**qed**  
**qed**  
**proposition** *is-natural-transformation:*  
**shows** *natural-transformation*  $A_0.comp\ B_0.comp\ F_0.map_0\ G_0.map_0\ \tau.map$   
..  
**end**



## 2.2.4 Self-Enriched Case

Here we show that a closed monoidal category  $C$ , regarded as a category enriched in itself, it is isomorphic to its own underlying category. This is useful, because it is somewhat less cumbersome to work directly in the category  $C$  than in the higher-type version that results from the underlying category construction. Kelly often regards these two categories as identical.

```

locale self-enriched-category =
  elementary-closed-monoidal-category +
  enriched-category C T α ι ⟨Collect ide⟩ exp Id Comp
begin

  sublocale UC: underlying-category C T α ι ⟨Collect ide⟩ exp Id Comp ..

  abbreviation toUC
  where toUC g ≡ if arr g
            then UC.MkArr (dom g) (cod g) (g†)
            else UC.null

  lemma toUC-simps [simp]:
  assumes arr f
  shows UC.arr (toUC f)
  and UC.dom (toUC f) = toUC (dom f)
  and UC.cod (toUC f) = toUC (cod f)
    using assms UC.arr-char UC.dom-char UC.cod-char UP-def
    comp-cod-arr Id-def
  by auto

  lemma toUC-in-hom [intro]:
  assumes arr f
  shows UC.in-hom (toUC f) (UC.MkIde (dom f)) (UC.MkIde (cod f))
    using assms toUC-simps by fastforce

  sublocale toUC: functor C UC.comp toUC
    using toUC-simps UP-comp UC.COMP-def
    by unfold-locales auto

  abbreviation frmUC
  where frmUC g ≡ if UC.arr g
            then (UC.Map g)↓[UC.Dom g, UC.Cod g]
            else null

  lemma frmUC-simps [simp]:
  assumes UC.arr f
  shows arr (frmUC f)
  and dom (frmUC f) = frmUC (UC.dom f)
  and cod (frmUC f) = frmUC (UC.cod f)
    using assms UC.arr-char UC.dom-char UC.cod-char Uncurry-Curry
    Id-def lunit-in-hom DN-def

```

by auto

**lemma** *frmUC-in-hom* [intro]:  
**assumes** *UC.in-hom f a b*  
**shows**  $\langle \text{frmUC } f : \text{frmUC } a \rightarrow \text{frmUC } b \rangle$   
**using** *assms frmUC-simps* **by** *blast*

**lemma** *DN-Map-comp*:  
**assumes** *UC.seq g f*  
**shows**  $(\text{UC.Map } (\text{UC.comp } g f))^\downarrow[\text{UC.Dom } f, \text{UC.Cod } g] =$   
 $(\text{UC.Map } g)^\downarrow[\text{UC.Dom } g, \text{UC.Cod } g] \cdot$   
 $(\text{UC.Map } f)^\downarrow[\text{UC.Dom } f, \text{UC.Cod } f]$

**proof** –  
**have**  $(\text{UC.Map } (\text{UC.comp } g f))^\downarrow[\text{UC.Dom } f, \text{UC.Cod } g] =$   
 $((\text{UC.Map } (\text{UC.comp } g f))^\downarrow[\text{UC.Dom } f, \text{UC.Cod } g])^\uparrow$   
 $\downarrow[\text{UC.Dom } f, \text{UC.Cod } g]$   
**using** *assms UC.arr-char UC.seq-char [of g f]* **by** *fastforce*  
**also have**  $\dots = ((\text{UC.Map } g)^\downarrow[\text{UC.Dom } g, \text{UC.Cod } g] \cdot$   
 $(\text{UC.Map } f)^\downarrow[\text{UC.Dom } f, \text{UC.Cod } f])^\uparrow$   
 $\downarrow[\text{UC.Dom } f, \text{UC.Cod } g]$

**proof** –  
**have**  $((\text{UC.Map } (\text{UC.comp } g f))^\downarrow[\text{UC.Dom } f, \text{UC.Cod } g])^\uparrow =$   
 $\text{UC.Map } (\text{UC.comp } g f)$   
**using** *assms UC.arr-char UC.seq-char [of g f]* **by** *fastforce*  
**also have**  $\dots = \text{Comp } (\text{UC.Dom } f) (\text{UC.Dom } g) (\text{UC.Cod } g) \cdot$   
 $(\text{UC.Map } g \otimes \text{UC.Map } f) \cdot \iota^{-1}$   
**using** *assms UC.Map-comp UC.seq-char* **by** *blast*  
**also have**  $\dots = \text{Comp } (\text{UC.Dom } f) (\text{UC.Dom } g) (\text{UC.Cod } g) \cdot$   
 $((\text{UC.Map } g)^\downarrow[\text{UC.Dom } g, \text{UC.Cod } g])^\uparrow \otimes$   
 $((\text{UC.Map } f)^\downarrow[\text{UC.Dom } f, \text{UC.Cod } f])^\uparrow) \cdot \iota^{-1}$   
**using** *assms UC.seq-char UC.arr-char* **by** *auto*  
**also have**  $\dots = ((\text{UC.Map } g)^\downarrow[\text{UC.Dom } g, \text{UC.Cod } g] \cdot$   
 $(\text{UC.Map } f)^\downarrow[\text{UC.Dom } f, \text{UC.Cod } f])^\uparrow$

**proof** –  
**have**  $\text{dom } ((\text{UC.Map } f)^\downarrow[\text{UC.Dom } f, \text{UC.Cod } f]) = \text{UC.Dom } f$   
**using** *assms DN-Id UC.Dom-in-Obj frmUC-simps(2)* **by** *auto*  
**moreover have**  $\text{cod } ((\text{UC.Map } f)^\downarrow[\text{UC.Dom } f, \text{UC.Cod } f]) = \text{UC.Cod } f$   
**using** *assms DN-Id UC.Cod-in-Obj frmUC-simps(3)* **by** *auto*  
**moreover have**  $\text{seq } ((\text{UC.Map } g)^\downarrow[\text{UC.Cod } f, \text{UC.Cod } g])$   
 $((\text{UC.Map } f)^\downarrow[\text{UC.Dom } f, \text{UC.Cod } f])$   
**using** *assms frmUC-simps(1–3) UC.seq-char*  
**apply** (*intro seqI*)  
**apply** *auto[3]*  
**by** *metis+*  
**ultimately show** *?thesis*  
**using** *assms UP-comp UP-DN(2) UC.arr-char UC.seq-char*  
*in-homE seqI*  
**by** *auto*  
**qed**

**finally show** *?thesis* **by** *simp*  
**qed**  
**also have**  $\dots = (UC.Map\ g)^\downarrow[UC.Dom\ g,\ UC.Cod\ g] \cdot (UC.Map\ f)^\downarrow[UC.Dom\ f,\ UC.Cod\ f]$   
**proof** –  
**have**  $2: seq\ ((UC.Map\ g)^\downarrow[UC.Dom\ g,\ UC.Cod\ g])\ ((UC.Map\ f)^\downarrow[UC.Dom\ f,\ UC.Cod\ f])$   
**using** *assms frmUC-simps(1-3) UC.seq-char*  
**apply** (*elim UC.seqE, intro seqI*)  
**apply** *auto[3]*  
**by** *metis+*  
**moreover have**  $dom\ ((UC.Map\ g)^\downarrow[UC.Dom\ g,\ UC.Cod\ g]) \cdot (UC.Map\ f)^\downarrow[UC.Dom\ f,\ UC.Cod\ f] = UC.Dom\ f$   
**using** *assms 2 UC.Dom-comp UC.arr-char [of f] by auto*  
**moreover have**  $cod\ ((UC.Map\ g)^\downarrow[UC.Dom\ g,\ UC.Cod\ g]) \cdot (UC.Map\ f)^\downarrow[UC.Dom\ f,\ UC.Cod\ f] = UC.Cod\ g$   
**using** *assms 2 UC.Cod-comp UC.arr-char [of g] by auto*  
**ultimately show** *?thesis*  
**using** *assms*  
 $UP-DN(3)\ [of\ (UC.Map\ g)^\downarrow[UC.Dom\ g,\ UC.Cod\ g] \cdot (UC.Map\ f)^\downarrow[UC.Dom\ f,\ UC.Cod\ f]]$   
**by** *simp*  
**qed**  
**finally show** *?thesis* **by** *blast*  
**qed**

**sublocale** *frmUC: functor UC.comp C frmUC*

**proof**

**show**  $\bigwedge f. \neg UC.arr\ f \implies frmUC\ f = null$   
**by** *simp*  
**show**  $\bigwedge f. UC.arr\ f \implies arr\ (frmUC\ f)$   
**using** *UC.arr-char frmUC-simps(1) by blast*  
**show**  $\bigwedge f. UC.arr\ f \implies dom\ (frmUC\ f) = frmUC\ (UC.dom\ f)$   
**using** *frmUC-simps(2) by blast*  
**show**  $\bigwedge f. UC.arr\ f \implies cod\ (frmUC\ f) = frmUC\ (UC.cod\ f)$   
**using** *frmUC-simps(3) by blast*  
**fix** *f g*  
**assume** *fg: UC.seq g f*  
**show**  $frmUC\ (UC.comp\ g\ f) = frmUC\ g \cdot frmUC\ f$   
**using** *fg UC.seq-char DN-Map-comp by auto*  
**qed**

**sublocale** *inverse-functors UC.comp C toUC frmUC*

**proof**

**show**  $frmUC \circ toUC = map$   
**using** *extensionality comp-arr-dom comp-assoc Uncurry-Curry by auto*  
**interpret** *to-frm: composite-functor UC.comp C UC.comp frmUC toUC ..*

```

show  $toUC \circ frmUC = UC.map$ 
proof
  fix  $f$ 
  show  $(toUC \circ frmUC) f = UC.map f$ 
  proof (cases  $UC.arr f$ )
    show  $\neg UC.arr f \implies ?thesis$ 
      using  $UC.extensionality$  by auto
    assume  $f: UC.arr f$ 
    show  $?thesis$ 
    proof (intro  $UC.arr-eqI$ )
      show  $UC.arr ((toUC \circ frmUC) f)$ 
        using  $f$  by blast
      show  $UC.arr (UC.map f)$ 
        using  $f$  by blast
      show  $UC.Dom ((toUC \circ frmUC) f) = UC.Dom (UC.map f)$ 
        using  $f UC.Dom-in-Obj frmUC.preserves-arr UC.arr-char [of f]$ 
        by auto
      show  $UC.Cod (to-frm.map f) = UC.Cod (UC.map f)$ 
        using  $f UC.arr-char [of f]$  by auto
      show  $UC.Map (to-frm.map f) = UC.Map (UC.map f)$ 
        using  $f UP-DN UC.arr-char [of f]$  by auto
    qed
  qed
qed
qed
qed

lemma inverse-functors-toUC-frmUC:
shows inverse-functors  $UC.comp C toUC frmUC$ 
  ..

corollary enriched-category-isomorphic-to-underlying-category:
shows isomorphic-categories  $UC.comp C$ 
  using inverse-functors-toUC-frmUC
  by unfold-locales blast

end

```

## 2.3 Opposite of an Enriched Category

Construction of the opposite of an enriched category (*cf.* [1] (1.19)) requires that the underlying monoidal category be symmetric, in order to introduce the required “twist” in the definition of composition.

```

locale opposite-enriched-category =
  symmetric-monoidal-category +
  EC: enriched-category
begin

```

```

  interpretation elementary-symmetric-monoidal-category

```

*C tensor unity lunit runit assoc sym*  
**using** *induces-elementary-symmetric-monoidal-category<sub>CMC</sub>* **by** *blast*

**abbreviation** *(input)*  $Hom_{op}$   
**where**  $Hom_{op} a b \equiv Hom b a$

**abbreviation**  $Comp_{op}$   
**where**  $Comp_{op} a b c \equiv Comp c b a \cdot s[Hom c b, Hom b a]$

**sublocale** *enriched-category*  $C T \alpha \iota Obj Hom_{op} Id Comp_{op}$   
**proof**

**show**  $*$ :  $\bigwedge a b. \llbracket a \in Obj; b \in Obj \rrbracket \implies ide (Hom b a)$   
**using** *EC.ide-Hom* **by** *blast*  
**show**  $\bigwedge a. a \in Obj \implies \llbracket Id a : \mathcal{I} \rightarrow Hom a a \rrbracket$   
**using** *EC.Id-in-hom* **by** *blast*  
**show**  $**$ :  $\bigwedge a b c. \llbracket a \in Obj; b \in Obj; c \in Obj \rrbracket \implies$   
 $\llbracket Comp_{op} a b c : Hom c b \otimes Hom b a \rightarrow Hom c a \rrbracket$   
**using** *sym-in-hom EC.ide-Hom EC.Comp-in-hom* **by** *auto*  
**show**  $\bigwedge a b. \llbracket a \in Obj; b \in Obj \rrbracket \implies$   
 $Comp_{op} a a b \cdot (Hom b a \otimes Id a) = r[Hom b a]$

**proof** –  
**fix**  $a b$   
**assume**  $a: a \in Obj$  **and**  $b: b \in Obj$   
**have**  $Comp_{op} a a b \cdot (Hom b a \otimes Id a) =$   
 $Comp b a a \cdot s[Hom b a, Hom a a] \cdot (Hom b a \otimes Id a)$   
**using** *comp-assoc* **by** *simp*  
**also have**  $\dots = Comp b a a \cdot (Id a \otimes Hom b a) \cdot s[Hom b a, \mathcal{I}]$   
**using**  $a b$  *sym-naturality [of Hom b a Id a]* *sym-in-hom*  
 $EC.Id-in-hom EC.ide-Hom$   
**by** *fastforce*  
**also have**  $\dots = (Comp b a a \cdot (Id a \otimes Hom b a)) \cdot s[Hom b a, \mathcal{I}]$   
**using** *comp-assoc* **by** *simp*  
**also have**  $\dots = l[Hom b a] \cdot s[Hom b a, \mathcal{I}]$   
**using**  $a b$  *EC.Comp-Id-Hom* **by** *simp*  
**also have**  $\dots = r[Hom b a]$   
**using**  $a b$  *unitor-coherence EC.ide-Hom* **by** *presburger*  
**finally show**  $Comp_{op} a a b \cdot (Hom b a \otimes Id a) = r[Hom b a]$   
**by** *blast*

**qed**  
**show**  $\bigwedge a b. \llbracket a \in Obj; b \in Obj \rrbracket \implies$   
 $Comp_{op} a b b \cdot (Id b \otimes Hom b a) = l[Hom b a]$

**proof** –  
**fix**  $a b$   
**assume**  $a: a \in Obj$  **and**  $b: b \in Obj$   
**have**  $Comp_{op} a b b \cdot (Id b \otimes Hom b a) =$   
 $Comp b b a \cdot s[Hom b b, Hom b a] \cdot (Id b \otimes Hom b a)$   
**using** *comp-assoc* **by** *simp*  
**also have**  $\dots = Comp b b a \cdot (Hom b a \otimes Id b) \cdot s[\mathcal{I}, Hom b a]$

**using**  $a\ b$  *sym-naturality* [of  $Id\ b\ Hom\ b\ a$ ] *sym-in-hom*  
 $EC.Id-in-hom\ EC.ide-Hom$   
**by** *force*  
**also have**  $\dots = (Comp\ b\ b\ a \cdot (Hom\ b\ a \otimes Id\ b)) \cdot s[\mathcal{I}, Hom\ b\ a]$   
**using** *comp-assoc* **by** *simp*  
**also have**  $\dots = r[Hom\ b\ a] \cdot s[\mathcal{I}, Hom\ b\ a]$   
**using**  $a\ b$  *EC.Comp-Hom-Id* **by** *simp*  
**also have**  $\dots = l[Hom\ b\ a]$   
**proof** –  
  
**have**  $r[Hom\ b\ a] \cdot s[\mathcal{I}, Hom\ b\ a] =$   
 $(l[Hom\ b\ a] \cdot s[Hom\ b\ a, \mathcal{I}]) \cdot s[\mathcal{I}, Hom\ b\ a]$   
**using**  $a\ b$  *unitor-coherence* *EC.ide-Hom* **by** *simp*  
**also have**  $\dots = l[Hom\ b\ a] \cdot s[Hom\ b\ a, \mathcal{I}] \cdot s[\mathcal{I}, Hom\ b\ a]$   
**using** *comp-assoc* **by** *simp*  
**also have**  $\dots = l[Hom\ b\ a]$   
**using**  $a\ b$  *comp-arr-dom comp-arr-inv sym-inverse* **by** *simp*  
**finally show** *?thesis* **by** *blast*  
**qed**  
**finally show**  $Comp_{op}\ a\ b\ b \cdot (Id\ b \otimes Hom\ b\ a) = l[Hom\ b\ a]$   
**by** *blast*  
**qed**  
**show**  $\bigwedge a\ b\ c\ d. [a \in Obj; b \in Obj; c \in Obj; d \in Obj] \implies$   
 $Comp_{op}\ a\ b\ d \cdot (Comp_{op}\ b\ c\ d \otimes Hom\ b\ a) =$   
 $Comp_{op}\ a\ c\ d \cdot (Hom\ d\ c \otimes Comp_{op}\ a\ b\ c) \cdot$   
 $a[Hom\ d\ c, Hom\ c\ b, Hom\ b\ a]$   
**proof** –  
**fix**  $a\ b\ c\ d$   
**assume**  $a: a \in Obj$  **and**  $b: b \in Obj$  **and**  $c: c \in Obj$  **and**  $d: d \in Obj$   
**have**  $Comp_{op}\ a\ b\ d \cdot (Comp_{op}\ b\ c\ d \otimes Hom\ b\ a) =$   
 $Comp_{op}\ a\ b\ d \cdot (Comp\ d\ c\ b \otimes Hom\ b\ a) \cdot$   
 $(s[Hom\ d\ c, Hom\ c\ b] \otimes Hom\ b\ a)$   
**using**  $a\ b\ c\ d$  *\*\* interchange comp-ide-arr ide-in-hom seqI'*  
 $EC.ide-Hom$   
**by** *metis*  
**also have**  $\dots = (Comp\ d\ b\ a \cdot$   
 $(s[Hom\ d\ b, Hom\ b\ a] \cdot (Comp\ d\ c\ b \otimes Hom\ b\ a))) \cdot$   
 $(s[Hom\ d\ c, Hom\ c\ b] \otimes Hom\ b\ a))$   
**using** *comp-assoc* **by** *simp*  
**also have**  $\dots = (Comp\ d\ b\ a \cdot$   
 $((Hom\ b\ a \otimes Comp\ d\ c\ b) \cdot$   
 $s[Hom\ c\ b \otimes Hom\ d\ c, Hom\ b\ a]) \cdot$   
 $(s[Hom\ d\ c, Hom\ c\ b] \otimes Hom\ b\ a))$   
**using**  $a\ b\ c\ d$  *sym-naturality* *EC.Comp-in-hom ide-char*  
 $in-homE\ EC.ide-Hom$   
**by** *metis*  
**also have**  $\dots = (Comp\ d\ b\ a \cdot (Hom\ b\ a \otimes Comp\ d\ c\ b)) \cdot$   
 $(s[Hom\ c\ b \otimes Hom\ d\ c, Hom\ b\ a] \cdot$   
 $(s[Hom\ d\ c, Hom\ c\ b] \otimes Hom\ b\ a))$

**using** *comp-assoc by simp*  
**also have** ... =  $(\text{Comp } d \ c \ a \cdot (\text{Comp } c \ b \ a \otimes \text{Hom } d \ c)) \cdot$   
 $a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot$   
 $(s[\text{Hom } c \ b \otimes \text{Hom } d \ c, \text{Hom } b \ a] \cdot$   
 $(s[\text{Hom } d \ c, \text{Hom } c \ b] \otimes \text{Hom } b \ a))$

**proof** –

**have**  $\text{Comp } d \ b \ a \cdot (\text{Hom } b \ a \otimes \text{Comp } d \ c \ b) =$   
 $(\text{Comp } d \ b \ a \cdot (\text{Hom } b \ a \otimes \text{Comp } d \ c \ b)) \cdot$   
 $(\text{Hom } b \ a \otimes \text{Hom } c \ b \otimes \text{Hom } d \ c)$   
**using** *a b c d EC.Comp-in-hom arrI comp-in-homI ide-in-hom*  
*tensor-in-hom EC.ide-Hom*

**proof** –

**have** *seq*  $(\text{Comp } d \ b \ a) (\text{Hom } b \ a \otimes \text{Comp } d \ c \ b)$   
**using** *a b c d EC.Comp-in-hom arrI comp-in-homI ide-in-hom*  
*tensor-in-hom EC.ide-Hom*  
**by** *meson*  
**moreover have** *dom*  $(\text{Comp } d \ b \ a \cdot (\text{Hom } b \ a \otimes \text{Comp } d \ c \ b)) =$   
 $(\text{Hom } b \ a \otimes \text{Hom } c \ b \otimes \text{Hom } d \ c)$   
**using** *a b c d EC.Comp-in-hom dom-comp dom-tensor ideD(1-2)*  
*in-homE calculation EC.ide-Hom*  
**by** *metis*  
**ultimately show** *?thesis*  
**using** *a b c d EC.Comp-in-hom comp-arr-dom by metis*

**qed**

**also have** ... =  
 $(\text{Comp } d \ b \ a \cdot (\text{Hom } b \ a \otimes \text{Comp } d \ c \ b)) \cdot$   
 $a[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c]$   
**using** *a b c d comp-assoc-assoc'(1) EC.ide-Hom by simp*

**also have** ... =  $(\text{Comp } d \ b \ a \cdot (\text{Hom } b \ a \otimes \text{Comp } d \ c \ b)) \cdot$   
 $a[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot$   
 $a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c]$   
**using** *comp-assoc by simp*

**also have** ... =  $(\text{Comp } d \ c \ a \cdot (\text{Comp } c \ b \ a \otimes \text{Hom } d \ c)) \cdot$   
 $a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c]$   
**using** *a b c d EC.Comp-assoc by simp*

**also have** ... =  $\text{Comp } d \ c \ a \cdot (\text{Comp } c \ b \ a \otimes \text{Hom } d \ c) \cdot$   
 $a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c]$   
**using** *comp-assoc by simp*

**finally have**  $\text{Comp } d \ b \ a \cdot (\text{Hom } b \ a \otimes \text{Comp } d \ c \ b) =$   
 $\text{Comp } d \ c \ a \cdot (\text{Comp } c \ b \ a \otimes \text{Hom } d \ c) \cdot$   
 $a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c]$   
**by** *blast*  
**thus** *?thesis by simp*

**qed**

**also have** ... =  $(\text{Comp } d \ c \ a \cdot (\text{Comp } c \ b \ a \otimes \text{Hom } d \ c)) \cdot$   
 $(a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot$   
 $s[\text{Hom } c \ b \otimes \text{Hom } d \ c, \text{Hom } b \ a] \cdot$   
 $(s[\text{Hom } d \ c, \text{Hom } c \ b] \otimes \text{Hom } b \ a))$   
**using** *comp-assoc by simp*

**finally have LHS:**  $(\text{Comp } d \ b \ a \cdot \text{s}[\text{Hom } d \ b, \text{Hom } b \ a]) \cdot$   
 $(\text{Comp } d \ c \ b \cdot \text{s}[\text{Hom } d \ c, \text{Hom } c \ b] \otimes \text{Hom } b \ a) =$   
 $(\text{Comp } d \ c \ a \cdot (\text{Comp } c \ b \ a \otimes \text{Hom } d \ c)) \cdot$   
 $(\text{a}^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot$   
 $\text{s}[\text{Hom } c \ b \otimes \text{Hom } d \ c, \text{Hom } b \ a] \cdot$   
 $(\text{s}[\text{Hom } d \ c, \text{Hom } c \ b] \otimes \text{Hom } b \ a))$

**by blast**

**have**  $\text{Comp}_{op} \ a \ c \ d \cdot (\text{Hom } d \ c \otimes \text{Comp}_{op} \ a \ b \ c) \cdot$   
 $\text{a}[\text{Hom } d \ c, \text{Hom } c \ b, \text{Hom } b \ a] =$   
 $\text{Comp } d \ c \ a \cdot$   
 $(\text{s}[\text{Hom } d \ c, \text{Hom } c \ a] \cdot$   
 $(\text{Hom } d \ c \otimes \text{Comp } c \ b \ a \cdot \text{s}[\text{Hom } c \ b, \text{Hom } b \ a])) \cdot$   
 $\text{a}[\text{Hom } d \ c, \text{Hom } c \ b, \text{Hom } b \ a]$

**using comp-assoc by simp**

**also have ... =**  
 $\text{Comp } d \ c \ a \cdot$   
 $((\text{Comp } c \ b \ a \cdot \text{s}[\text{Hom } c \ b, \text{Hom } b \ a] \otimes \text{Hom } d \ c) \cdot$   
 $\text{s}[\text{Hom } d \ c, \text{Hom } c \ b \otimes \text{Hom } b \ a]) \cdot$   
 $\text{a}[\text{Hom } d \ c, \text{Hom } c \ b, \text{Hom } b \ a]$

**using a b c d \*\* sym-naturality ide-char in-homE EC.ide-Hom**

**by metis**

**also have ... =**  
 $\text{Comp } d \ c \ a \cdot$   
 $((((\text{Comp } c \ b \ a \otimes \text{Hom } d \ c) \cdot (\text{s}[\text{Hom } c \ b, \text{Hom } b \ a] \otimes \text{Hom } d \ c)) \cdot$   
 $\text{s}[\text{Hom } d \ c, \text{Hom } c \ b \otimes \text{Hom } b \ a]) \cdot$   
 $\text{a}[\text{Hom } d \ c, \text{Hom } c \ b, \text{Hom } b \ a]$

**using a b c d \*\* interchange comp-arr-dom ideD(1-2)**  
*in-homE EC.ide-Hom*

**by metis**

**also have ... =**  $(\text{Comp } d \ c \ a \cdot (\text{Comp } c \ b \ a \otimes \text{Hom } d \ c)) \cdot$   
 $((\text{s}[\text{Hom } c \ b, \text{Hom } b \ a] \otimes \text{Hom } d \ c) \cdot$   
 $\text{s}[\text{Hom } d \ c, \text{Hom } c \ b \otimes \text{Hom } b \ a] \cdot$   
 $\text{a}[\text{Hom } d \ c, \text{Hom } c \ b, \text{Hom } b \ a])$

**using comp-assoc by simp**

**also have ... =**  $(\text{Comp } d \ c \ a \cdot (\text{Comp } c \ b \ a \otimes \text{Hom } d \ c)) \cdot$   
 $(\text{a}^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot$   
 $\text{s}[\text{Hom } c \ b \otimes \text{Hom } d \ c, \text{Hom } b \ a] \cdot$   
 $(\text{s}[\text{Hom } d \ c, \text{Hom } c \ b] \otimes \text{Hom } b \ a))$

**proof –**

**have**  $(\text{s}[\text{Hom } c \ b, \text{Hom } b \ a] \otimes \text{Hom } d \ c) \cdot$   
 $\text{s}[\text{Hom } d \ c, \text{Hom } c \ b \otimes \text{Hom } b \ a] \cdot$   
 $\text{a}[\text{Hom } d \ c, \text{Hom } c \ b, \text{Hom } b \ a] =$   
 $\text{a}^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot$   
 $\text{s}[\text{Hom } c \ b \otimes \text{Hom } d \ c, \text{Hom } b \ a] \cdot$   
 $(\text{s}[\text{Hom } d \ c, \text{Hom } c \ b] \otimes \text{Hom } b \ a)$

**proof –**

**have 1:**  $\text{s}[\text{Hom } d \ c, \text{Hom } c \ b \otimes \text{Hom } b \ a] \cdot$   
 $\text{a}[\text{Hom } d \ c, \text{Hom } c \ b, \text{Hom } b \ a] =$   
 $\text{a}^{-1}[\text{Hom } c \ b, \text{Hom } b \ a, \text{Hom } d \ c] \cdot$



$$(Hom\ c\ b \otimes s[Hom\ d\ c, Hom\ b\ a]) \cdot$$

$$a[Hom\ c\ b, Hom\ d\ c, Hom\ b\ a] \cdot$$

$$(s[Hom\ d\ c, Hom\ c\ b] \otimes Hom\ b\ a)$$

**proof** –

**have**  $s[Hom\ d\ c, Hom\ c\ b \otimes Hom\ b\ a] \cdot$   
 $a[Hom\ d\ c, Hom\ c\ b, Hom\ b\ a] =$   
 $(a^{-1}[Hom\ c\ b, Hom\ b\ a, Hom\ d\ c] \cdot$   
 $a[Hom\ c\ b, Hom\ b\ a, Hom\ d\ c]) \cdot$   
 $s[Hom\ d\ c, Hom\ c\ b \otimes Hom\ b\ a] \cdot$   
 $a[Hom\ d\ c, Hom\ c\ b, Hom\ b\ a]$

**using**  $a\ b\ c\ d\ comp\text{-}assoc\text{-}assoc'(2)\ comp\text{-}cod\text{-}arr$  **by** *simp*

**also have** ... =  
 $a^{-1}[Hom\ c\ b, Hom\ b\ a, Hom\ d\ c] \cdot$   
 $a[Hom\ c\ b, Hom\ b\ a, Hom\ d\ c] \cdot$   
 $s[Hom\ d\ c, Hom\ c\ b \otimes Hom\ b\ a] \cdot$   
 $a[Hom\ d\ c, Hom\ c\ b, Hom\ b\ a]$

**using** *comp-assoc* **by** *simp*

**also have** ... =  
 $a^{-1}[Hom\ c\ b, Hom\ b\ a, Hom\ d\ c] \cdot$   
 $(Hom\ c\ b \otimes s[Hom\ d\ c, Hom\ b\ a]) \cdot$   
 $a[Hom\ c\ b, Hom\ d\ c, Hom\ b\ a] \cdot$   
 $(s[Hom\ d\ c, Hom\ c\ b] \otimes Hom\ b\ a)$

**using**  $a\ b\ c\ d\ assoc\text{-}coherence\ EC.ide\text{-}Hom$  **by** *auto*

**finally show** *?thesis* **by** *blast*

**qed**

**have** 2:  $(s[Hom\ c\ b, Hom\ b\ a] \otimes Hom\ d\ c) \cdot$   
 $a^{-1}[Hom\ c\ b, Hom\ b\ a, Hom\ d\ c] \cdot$   
 $(Hom\ c\ b \otimes s[Hom\ d\ c, Hom\ b\ a]) =$   
 $a^{-1}[Hom\ b\ a, Hom\ c\ b, Hom\ d\ c] \cdot$   
 $s[Hom\ c\ b \otimes Hom\ d\ c, Hom\ b\ a] \cdot$   
 $inv\ a[Hom\ c\ b, Hom\ d\ c, Hom\ b\ a]$

**proof** –

**have**  $(s[Hom\ c\ b, Hom\ b\ a] \otimes Hom\ d\ c) \cdot$   
 $a^{-1}[Hom\ c\ b, Hom\ b\ a, Hom\ d\ c] \cdot$   
 $(Hom\ c\ b \otimes s[Hom\ d\ c, Hom\ b\ a]) =$   
 $inv\ ((Hom\ c\ b \otimes s[Hom\ b\ a, Hom\ d\ c]) \cdot$   
 $a[Hom\ c\ b, Hom\ b\ a, Hom\ d\ c] \cdot$   
 $(s[Hom\ b\ a, Hom\ c\ b] \otimes Hom\ d\ c))$

**proof** –

**have**  $inv\ ((Hom\ c\ b \otimes s[Hom\ b\ a, Hom\ d\ c]) \cdot$   
 $a[Hom\ c\ b, Hom\ b\ a, Hom\ d\ c] \cdot$   
 $(s[Hom\ b\ a, Hom\ c\ b] \otimes Hom\ d\ c)) =$   
 $inv\ (a[Hom\ c\ b, Hom\ b\ a, Hom\ d\ c] \cdot$   
 $(s[Hom\ b\ a, Hom\ c\ b] \otimes Hom\ d\ c)) \cdot$   
 $inv\ (Hom\ c\ b \otimes s[Hom\ b\ a, Hom\ d\ c])$

**using**  $a\ b\ c\ d\ EC.ide\text{-}Hom$   
 $inv\text{-}comp$  [of  $a[Hom\ c\ b, Hom\ b\ a, Hom\ d\ c] \cdot$   
 $(s[Hom\ b\ a, Hom\ c\ b] \otimes Hom\ d\ c)$   
 $Hom\ c\ b \otimes s[Hom\ b\ a, Hom\ d\ c]$ ]

**by** *fastforce*  
**also have** ... =  
 $(\text{inv } (s[\text{Hom } b \ a, \text{Hom } c \ b] \otimes \text{Hom } d \ c) \cdot$   
 $\quad a^{-1}[\text{Hom } c \ b, \text{Hom } b \ a, \text{Hom } d \ c]) \cdot$   
 $\quad \text{inv } (\text{Hom } c \ b \otimes s[\text{Hom } b \ a, \text{Hom } d \ c])$   
**using** *a b c d EC.ide-Hom inv-comp* **by** *simp*  
**also have** ... =  
 $((s[\text{Hom } c \ b, \text{Hom } b \ a] \otimes \text{Hom } d \ c) \cdot$   
 $\quad a^{-1}[\text{Hom } c \ b, \text{Hom } b \ a, \text{Hom } d \ c]) \cdot$   
 $\quad (\text{Hom } c \ b \otimes s[\text{Hom } d \ c, \text{Hom } b \ a])$   
  
**using** *a b c d sym-inverse inverse-unique*  
**apply** *auto[1]*  
**by** (*metis \**)  
**finally show** *?thesis*  
**using** *comp-assoc* **by** *simp*  
**qed**  
**also have** ... =  
 $\text{inv } (a[\text{Hom } c \ b, \text{Hom } d \ c, \text{Hom } b \ a] \cdot$   
 $\quad s[\text{Hom } b \ a, \text{Hom } c \ b \otimes \text{Hom } d \ c] \cdot$   
 $\quad a[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c])$   
**using** *a b c d assoc-coherence EC.ide-Hom* **by** *auto*  
**also have** ... =  
 $a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot$   
 $\text{inv } s[\text{Hom } b \ a, \text{Hom } c \ b \otimes \text{Hom } d \ c] \cdot$   
 $a^{-1}[\text{Hom } c \ b, \text{Hom } d \ c, \text{Hom } b \ a]$   
**using** *a b c d EC.ide-Hom inv-comp inv-tensor comp-assoc*  
*isos-compose*  
**by** *auto*  
**also have** ... =  
 $a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot$   
 $s[\text{Hom } c \ b \otimes \text{Hom } d \ c, \text{Hom } b \ a] \cdot$   
 $a^{-1}[\text{Hom } c \ b, \text{Hom } d \ c, \text{Hom } b \ a]$   
**using** *a b c d sym-inverse inv-is-inverse inverse-unique*  
**by** (*metis tensor-preserves-ide EC.ide-Hom*)  
**finally show** *?thesis* **by** *blast*  
**qed**  
**hence**  $(s[\text{Hom } c \ b, \text{Hom } b \ a] \otimes \text{Hom } d \ c) \cdot$   
 $a^{-1}[\text{Hom } c \ b, \text{Hom } b \ a, \text{Hom } d \ c] \cdot$   
 $(\text{Hom } c \ b \otimes s[\text{Hom } d \ c, \text{Hom } b \ a]) \cdot$   
 $a[\text{Hom } c \ b, \text{Hom } d \ c, \text{Hom } b \ a] =$   
 $a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot$   
 $s[\text{Hom } c \ b \otimes \text{Hom } d \ c, \text{Hom } b \ a] \cdot$   
 $\text{inv } a[\text{Hom } c \ b, \text{Hom } d \ c, \text{Hom } b \ a] \cdot$   
 $a[\text{Hom } c \ b, \text{Hom } d \ c, \text{Hom } b \ a]$   
**by** (*metis comp-assoc*)  
**hence**  $\exists: (s[\text{Hom } c \ b, \text{Hom } b \ a] \otimes \text{Hom } d \ c) \cdot$   
 $a^{-1}[\text{Hom } c \ b, \text{Hom } b \ a, \text{Hom } d \ c] \cdot$   
 $(\text{Hom } c \ b \otimes s[\text{Hom } d \ c, \text{Hom } b \ a]) \cdot$

$$\begin{aligned} & a[\text{Hom } c \ b, \text{Hom } d \ c, \text{Hom } b \ a] = \\ & a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot \\ & s[\text{Hom } c \ b \otimes \text{Hom } d \ c, \text{Hom } b \ a] \\ \text{using } & a \ b \ c \ \text{comp-arr-dom } d \ \text{by fastforce} \\ \text{have } & (s[\text{Hom } c \ b, \text{Hom } b \ a] \otimes \text{Hom } d \ c) \cdot \\ & s[\text{Hom } d \ c, \text{Hom } c \ b \otimes \text{Hom } b \ a] \cdot \\ & a[\text{Hom } d \ c, \text{Hom } c \ b, \text{Hom } b \ a] = \\ & (s[\text{Hom } c \ b, \text{Hom } b \ a] \otimes \text{Hom } d \ c) \cdot \\ & a^{-1}[\text{Hom } c \ b, \text{Hom } b \ a, \text{Hom } d \ c] \cdot \\ & (\text{Hom } c \ b \otimes s[\text{Hom } d \ c, \text{Hom } b \ a]) \cdot \\ & a[\text{Hom } c \ b, \text{Hom } d \ c, \text{Hom } b \ a] \cdot \\ & (s[\text{Hom } d \ c, \text{Hom } c \ b] \otimes \text{Hom } b \ a) \\ \text{using } & 1 \ \text{by simp} \\ \text{also have } & \dots = \\ & ((s[\text{Hom } c \ b, \text{Hom } b \ a] \otimes \text{Hom } d \ c) \cdot \\ & a^{-1}[\text{Hom } c \ b, \text{Hom } b \ a, \text{Hom } d \ c] \cdot \\ & (\text{Hom } c \ b \otimes s[\text{Hom } d \ c, \text{Hom } b \ a]) \cdot \\ & a[\text{Hom } c \ b, \text{Hom } d \ c, \text{Hom } b \ a]) \cdot \\ & (s[\text{Hom } d \ c, \text{Hom } c \ b] \otimes \text{Hom } b \ a) \\ \text{using } & \text{comp-assoc by simp} \\ \text{also have } & \dots = \\ & (a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot \\ & s[\text{Hom } c \ b \otimes \text{Hom } d \ c, \text{Hom } b \ a]) \cdot \\ & (s[\text{Hom } d \ c, \text{Hom } c \ b] \otimes \text{Hom } b \ a) \\ \text{using } & 3 \ \text{by simp} \\ \text{also have } & \dots = \\ & a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot \\ & s[\text{Hom } c \ b \otimes \text{Hom } d \ c, \text{Hom } b \ a] \cdot \\ & (s[\text{Hom } d \ c, \text{Hom } c \ b] \otimes \text{Hom } b \ a) \\ \text{using } & \text{comp-assoc by simp} \\ \text{finally show } & \text{?thesis by simp} \\ \text{qed} \\ \text{thus } & \text{?thesis by auto} \\ \text{qed} \\ \text{finally have } & \text{RHS: } \text{Comp}_{op} \ a \ c \ d \cdot \\ & (\text{Hom } d \ c \otimes \text{Comp}_{op} \ a \ b \ c) \cdot \\ & a[\text{Hom } d \ c, \text{Hom } c \ b, \text{Hom } b \ a] = \\ & (\text{Comp } d \ c \ a \cdot (\text{Comp } c \ b \ a \otimes \text{Hom } d \ c)) \cdot \\ & (a^{-1}[\text{Hom } b \ a, \text{Hom } c \ b, \text{Hom } d \ c] \cdot \\ & s[\text{Hom } c \ b \otimes \text{Hom } d \ c, \text{Hom } b \ a] \cdot \\ & (s[\text{Hom } d \ c, \text{Hom } c \ b] \otimes \text{Hom } b \ a)) \\ \text{by } & \text{blast} \\ \text{show } & \text{Comp}_{op} \ a \ b \ d \cdot (\text{Comp}_{op} \ b \ c \ d \otimes \text{Hom } b \ a) = \\ & \text{Comp}_{op} \ a \ c \ d \cdot (\text{Hom } d \ c \otimes \text{Comp}_{op} \ a \ b \ c) \cdot \\ & a[\text{Hom } d \ c, \text{Hom } c \ b, \text{Hom } b \ a] \\ \text{using } & \text{LHS RHS by simp} \\ \text{qed} \\ \text{qed} \end{aligned}$$

end

### 2.3.1 Relation between $(-^{op})_0$ and $(-_0)^{op}$

Kelly (comment before (1.22)) claims, for a category  $A$  enriched in a symmetric monoidal category, that we have  $(A^{op})_0 = (A_0)^{op}$ . This point becomes somewhat confusing, as it depends on the particular formalization one adopts for the notion of “category”.

As we can see from the next two facts (*Op-UC-hom-char* and *UC-Op-hom-char*), the hom-sets *Op.UC.hom a b* and *UC.Op.hom a b* are both obtained by using *UC.MkArr* to “tag” elements of *hom I (Hom (UC.Dom b) (UC.Dom a))* with *UC.Dom a* and *UC.Dom b*. These two hom-sets are formally distinct if (as is the case for us), the arrows of a category are regarded as containing information about their domain and codomain, so that the hom-sets are disjoint. On the other hand, if one regards a category as a collection of mappings that assign to each pair of objects  $a$  and  $b$  a corresponding set *hom a b*, then the hom-sets *Op.UC.hom a b* and *UC.Op.hom a b* could be arranged to be equal, as Kelly suggests.

**locale** *category-enriched-in-symmetric-monoidal-category = symmetric-monoidal-category + enriched-category*

**begin**

**interpretation** *elementary-symmetric-monoidal-category*

*C tensor unity lunit runit assoc sym*

**using** *induces-elementary-symmetric-monoidal-category<sub>CMC</sub> by blast*

**interpretation** *Op: opposite-enriched-category C T α ι σ Obj Hom Id Comp ..*

**interpretation** *Op<sub>0</sub>: underlying-category C T α ι Obj Op.Hom<sub>op</sub> Id Op.Comp<sub>op</sub>*

..

**interpretation** *UC: underlying-category C T α ι Obj Hom Id Comp ..*

**interpretation** *UC.Op: dual-category UC.comp ..*

**lemma** *Op-UC-hom-char:*

**assumes** *UC.ide a and UC.ide b*

**shows** *Op<sub>0</sub>.hom a b =*

*UC.MkArr (UC.Dom a) (UC.Dom b) ‘*

*hom I (Hom (UC.Dom b) (UC.Dom a))*

**using** *assms Op<sub>0</sub>.hom-char [of UC.Dom a UC.Dom b]*

*UC.ide-char [of a] UC.ide-char [of b] UC.arr-char*

**by force**

**lemma** *UC-Op-hom-char:*

**assumes** *UC.ide a and UC.ide b*

**shows** *UC.Op.hom a b =*

*UC.MkArr (UC.Dom b) (UC.Dom a) ‘*

$hom \mathcal{I} (Hom (UC.Dom b) (UC.Dom a))$   
**using** *assms*  $UC.Op.hom-char UC.hom-char [of UC.Dom b UC.Dom a]$   
 $UC.ide-char_{CC}$   
**by** *simp*

**abbreviation**  $toUCOp$

**where**  $toUCOp f \equiv if Op_0.arr f$   
 $then UC.MkArr (Op_0.Cod f) (Op_0.Dom f) (Op_0.Map f)$   
 $else UC.Op.null$

**sublocale**  $toUCOp$ : functor  $Op_0.comp UC.Op.comp toUCOp$

**proof**

**show**  $\bigwedge f. \neg Op_0.arr f \implies toUCOp f = UC.Op.null$   
**by** *simp*  
**show**  $1: \bigwedge f. Op_0.arr f \implies UC.Op.arr (toUCOp f)$   
**using**  $Op_0.arr-char$  **by** *auto*  
**show**  $\bigwedge f. Op_0.arr f \implies UC.Op.dom (toUCOp f) = toUCOp (Op_0.dom f)$   
**using**  $1$  **by** *simp*  
**show**  $\bigwedge f. Op_0.arr f \implies UC.Op.cod (toUCOp f) = toUCOp (Op_0.cod f)$   
**using**  $1$  **by** *simp*  
**show**  $\bigwedge g f. Op_0.seq g f \implies$   
 $toUCOp (Op_0.comp g f) = UC.Op.comp (toUCOp g) (toUCOp f)$

**proof** –

**fix**  $f g$   
**assume**  $fg: Op_0.seq g f$   
**show**  $toUCOp (Op_0.comp g f) = UC.Op.comp (toUCOp g) (toUCOp f)$   
**proof** (*intro UC.arr-eqI*)

**show**  $UC.arr (toUCOp (Op_0.comp g f))$   
**using**  $1 fg UC.Op.arr-char$  **by** *blast*  
**show**  $2: UC.arr (UC.Op.comp (toUCOp g) (toUCOp f))$   
**using**  $1 Op_0.seq-char UC.seq-char fg$  **by** *force*  
**show**  $Op_0.Dom (toUCOp (Op_0.comp g f)) =$   
 $Op_0.Dom (UC.Op.comp (toUCOp g) (toUCOp f))$   
**using**  $1 2 fg Op_0.seq-char$  **by** *fastforce*  
**show**  $Op_0.Cod (toUCOp (Op_0.comp g f)) =$   
 $Op_0.Cod (UC.Op.comp (toUCOp g) (toUCOp f))$   
**using**  $1 2 fg Op_0.seq-char$  **by** *fastforce*  
**show**  $Op_0.Map (toUCOp (Op_0.comp g f)) =$   
 $Op_0.Map (UC.Op.comp (toUCOp g) (toUCOp f))$

**proof** –

**have**  $Op_0.Map (toUCOp (Op_0.comp g f)) =$   
 $Op.Comp_{op} (UC.Dom f) (UC.Dom g) (UC.Cod g) \cdot$   
 $(UC.Map g \otimes UC.Map f) \cdot \iota^{-1}$

**using**  $1 2 fg Op_0.seq-char$  **by** *auto*

**also have**  $\dots = Comp (Op_0.Cod g) (Op_0.Dom g) (Op_0.Dom f) \cdot$   
 $(s[Hom (Op_0.Cod g) (Op_0.Dom g),$   
 $Hom (Op_0.Dom g) (Op_0.Dom f)]) \cdot$   
 $(Op_0.Map g \otimes Op_0.Map f) \cdot \iota^{-1}$

**using** *comp-assoc* **by** *simp*

**also have** ... =  $Comp (Op_0.Cod\ g) (Op_0.Dom\ g) (Op_0.Dom\ f) \cdot$   
 $((Op_0.Map\ f \otimes Op_0.Map\ g) \cdot s[\mathcal{I}, \mathcal{I}]) \cdot \iota^{-1}$   
**using** *fg Op<sub>0</sub>.seq-char Op<sub>0</sub>.arr-char sym-naturality*  
**by** (*metis (no-types, lifting) in-homE mem-Collect-eq*)  
**also have** ... =  $Comp (Op_0.Cod\ g) (Op_0.Dom\ g) (Op_0.Dom\ f) \cdot$   
 $(Op_0.Map\ f \otimes Op_0.Map\ g) \cdot s[\mathcal{I}, \mathcal{I}] \cdot \iota^{-1}$   
**using** *comp-assoc by simp*  
**also have** ... =  $Comp (Op_0.Cod\ g) (Op_0.Dom\ g) (Op_0.Dom\ f) \cdot$   
 $(Op_0.Map\ f \otimes Op_0.Map\ g) \cdot \iota^{-1}$   
**using** *sym-inv-unit  $\iota$ -def monoidal-category-axioms*  
**by** (*simp add: monoidal-category.unitor-coincidence(1)*)  
**finally have**  $Op_0.Map (toUCOp (Op_0.comp\ g\ f)) =$   
 $Comp (Op_0.Cod\ g) (Op_0.Dom\ g) (Op_0.Dom\ f) \cdot$   
 $(Op_0.Map\ f \otimes Op_0.Map\ g) \cdot \iota^{-1}$   
**by** *blast*  
**also have** ... =  $Op_0.Map (UC.Op.comp (toUCOp\ g) (toUCOp\ f))$   
**using** *fg 2 by auto*  
**finally show** *?thesis by blast*  
**qed**  
**qed**  
**qed**  
**qed**

**lemma** *functor-toUCOp*:  
**shows** *functor*  $Op_0.comp\ UC.Op.comp\ toUCOp$   
**..**

**abbreviation** *toOp<sub>0</sub>*  
**where**  $toOp_0\ f \equiv$  *if*  $UC.Op.arr\ f$   
*then*  $Op_0.MkArr (UC.Cod\ f) (UC.Dom\ f) (UC.Map\ f)$   
*else*  $Op_0.null$

**sublocale** *toOp<sub>0</sub>: functor*  $UC.Op.comp\ Op_0.comp\ toOp_0$

**proof**  
**show**  $\bigwedge f. \neg UC.Op.arr\ f \implies toOp_0\ f = Op_0.null$   
**by** *simp*  
**show**  $1: \bigwedge f. UC.Op.arr\ f \implies Op_0.arr (toOp_0\ f)$   
**using** *UC.arr-char by simp*  
**show**  $\bigwedge f. UC.Op.arr\ f \implies Op_0.dom (toOp_0\ f) = toOp_0 (UC.Op.dom\ f)$   
**using** *1 by auto*  
**show**  $\bigwedge f. UC.Op.arr\ f \implies Op_0.cod (toOp_0\ f) = toOp_0 (UC.Op.cod\ f)$   
**using** *1 by auto*  
**show**  $\bigwedge g\ f. UC.Op.seq\ g\ f \implies$   
 $toOp_0 (UC.Op.comp\ g\ f) = Op_0.comp (toOp_0\ g) (toOp_0\ f)$   
**proof** –  
**fix**  $f\ g$   
**assume**  $fg: UC.Op.seq\ g\ f$   
**show**  $toOp_0 (UC.Op.comp\ g\ f) = Op_0.comp (toOp_0\ g) (toOp_0\ f)$   
**proof** (*intro Op<sub>0</sub>.arr-eqI*)

```

show  $Op_0.arr (toOp_0 (UC.Op.comp g f))$ 
  using fg 1 by blast
show  $2: Op_0.seq (toOp_0 g) (toOp_0 f)$ 
  using fg 1 UC.seq-char UC.arr-char Op_0.seq-char by fastforce
show  $Op_0.Dom (toOp_0 (UC.Op.comp g f)) =$ 
   $Op_0.Dom (Op_0.comp (toOp_0 g) (toOp_0 f))$ 
  using fg 1 2 Op_0.dom-char Op_0.cod-char UC.seq-char Op_0.seq-char
  by auto
show  $Op_0.Cod (toOp_0 (UC.Op.comp g f)) =$ 
   $Op_0.Cod (Op_0.comp (toOp_0 g) (toOp_0 f))$ 
  using fg 1 2 Op_0.dom-char Op_0.cod-char UC.seq-char Op_0.seq-char
  by auto
show  $Op_0.Map (toOp_0 (UC.Op.comp g f)) =$ 
   $Op_0.Map (Op_0.comp (toOp_0 g) (toOp_0 f))$ 
proof –
  have  $Op_0.Map (Op_0.comp (toOp_0 g) (toOp_0 f)) =$ 
   $Op.Comp_{op} (Op_0.Dom (toOp_0 f)) (Op_0.Dom (toOp_0 g))$ 
   $(Op_0.Cod (toOp_0 g)) \cdot$ 
   $(Op_0.Map (toOp_0 g) \otimes Op_0.Map (toOp_0 f)) \cdot inv \iota$ 
  using fg 1 2 UC.seq-char by auto
  also have ... =
   $Comp (Op_0.Dom g) (Op_0.Cod g) (Op_0.Cod f) \cdot$ 
   $(s[Hom (Op_0.Dom g) (Op_0.Cod g),$ 
   $Hom (Op_0.Cod g) (Op_0.Cod f)]) \cdot$ 
   $(Op_0.Map g \otimes Op_0.Map f) \cdot inv \iota$ 
  using fg comp-assoc by auto
  also have ... =
   $Comp (Op_0.Dom g) (Op_0.Cod g) (Op_0.Cod f) \cdot$ 
   $((Op_0.Map f \otimes Op_0.Map g) \cdot s[unity, unity]) \cdot inv \iota$ 
  using fg UC.seq-char UC.arr-char sym-naturality
  by (metis (no-types, lifting) in-homE UC.Op.arr-char
  UC.Op.comp-def mem-Collect-eq)
  also have ... =
   $Comp (Op_0.Dom g) (Op_0.Cod g) (Op_0.Cod f) \cdot$ 
   $(Op_0.Map f \otimes Op_0.Map g) \cdot s[unity, unity] \cdot inv \iota$ 
  using comp-assoc by simp
  also have ... =
   $Comp (Op_0.Dom g) (Op_0.Cod g) (Op_0.Cod f) \cdot$ 
   $(Op_0.Map f \otimes Op_0.Map g) \cdot inv \iota$ 
  using sym-inv-unit ι-def monoidal-category-axioms
  by (simp add: monoidal-category.unitor-coincidence(1))
  also have ... =  $Op_0.Map (toOp_0 (UC.Op.comp g f))$ 
  using fg UC.seq-char by simp
  finally show ?thesis by argo
qed
qed
qed
qed

```

```

lemma functor-toOp0:
shows functor UC.Op.comp Op0.comp toOp0
..

sublocale inverse-functors UC.Op.comp Op0.comp toUCOp toOp0
using Op0.MkArr-Map toUCOp.preserves-reflects-arr Op0.extensionality
       UC.MkArr-Map toOp0.preserves-reflects-arr UC.Op.extensionality
by unfold-locales auto

lemma inverse-functors-toUCOp-toOp0:
shows inverse-functors UC.Op.comp Op0.comp toUCOp toOp0
..

end

```

## 2.4 Enriched Hom Functors

Here we exhibit covariant and contravariant hom functors as enriched functors, as in [1] Section 1.6. We don't bother to exhibit them as partial functors of a single two-argument functor, as to do so would require us to define the tensor product of enriched categories; something that would require more technology for proving coherence conditions than we have developed at present.

### 2.4.1 Covariant Case

```

locale covariant-Hom =
  monoidal-category +

  C: elementary-closed-monoidal-category +
  enriched-category +
fixes x :: 'o
assumes x: x ∈ Obj
begin

  interpretation C: enriched-category C T α ι ‹Collect ide› exp C.Id C.Comp
    using C.is-enriched-in-itself by simp
  interpretation C: self-enriched-category C T α ι exp eval Curry ..

  abbreviation homo
  where homo ≡ Hom x

  abbreviation homa
  where homa ≡ λb c. if b ∈ Obj ∧ c ∈ Obj
        then Curry[Hom b c, Hom x b, Hom x c] (Comp x b c)
        else null

  sublocale enriched-functor C T α ι

```



*Obj Hom Id Comp*  
 ‹Collect ide› exp C.Id C.Comp  
 hom<sub>o</sub> hom<sub>a</sub>

**proof**

**show**  $\bigwedge a b. a \notin \text{Obj} \vee b \notin \text{Obj} \implies \text{hom}_a a b = \text{null}$   
**by** *auto*  
**show**  $\bigwedge y. y \in \text{Obj} \implies \text{hom}_o y \in \text{Collect ide}$   
**using** *x ide-Hom by auto*  
**show** \*:  $\bigwedge a b. \llbracket a \in \text{Obj}; b \in \text{Obj} \rrbracket \implies$   
 $\quad \llbracket \text{hom}_a a b : \text{Hom } a b \rightarrow \text{exp } (\text{hom}_o a) (\text{hom}_o b) \rrbracket$   
**using** *x by auto*  
**show**  $\bigwedge a. a \in \text{Obj} \implies \text{hom}_a a a \cdot \text{Id } a = \text{C.Id } (\text{hom}_o a)$   
**using** *x Comp-Id-Hom Comp-in-hom Id-in-hom C.Id-def C.comp-Curry-arr*  
**apply** *auto[1]*  
**by** (*metis ide-Hom*)  
**show**  $\bigwedge a b c. \llbracket a \in \text{Obj}; b \in \text{Obj}; c \in \text{Obj} \rrbracket \implies$   
 $\quad \text{C.Comp } (\text{hom}_o a) (\text{hom}_o b) (\text{hom}_o c) \cdot$   
 $\quad (\text{hom}_a b c \otimes \text{hom}_a a b) =$   
 $\quad \text{hom}_a a c \cdot \text{Comp } a b c$

**proof** –

**fix** *a b c*  
**assume** *a: a ∈ Obj and b: b ∈ Obj and c: c ∈ Obj*  
**have** *Uncurry[hom<sub>o</sub> a, hom<sub>o</sub> c]*  
 $(\text{C.Comp } (\text{hom}_o a) (\text{hom}_o b) (\text{hom}_o c) \cdot (\text{hom}_a b c \otimes \text{hom}_a a b)) =$   
 $\text{Uncurry}[\text{hom}_o a, \text{hom}_o c] (\text{hom}_a a c \cdot \text{Comp } a b c)$

**proof** –

**have** *Uncurry[hom<sub>o</sub> a, hom<sub>o</sub> c]*  
 $(\text{C.Comp } (\text{hom}_o a) (\text{hom}_o b) (\text{hom}_o c) \cdot (\text{hom}_a b c \otimes \text{hom}_a a b)) =$   
 $\text{Uncurry}[\text{hom}_o a, \text{hom}_o c]$   
 $(\text{Curry}[\text{exp } (\text{hom}_o b) (\text{hom}_o c) \otimes \text{exp } (\text{hom}_o a) (\text{hom}_o b), \text{hom}_o a,$   
 $\quad \text{hom}_o c]$   
 $(\text{eval } (\text{hom}_o b) (\text{hom}_o c) \cdot$   
 $\quad (\text{exp } (\text{hom}_o b) (\text{hom}_o c) \otimes \text{eval } (\text{hom}_o a) (\text{hom}_o b)) \cdot$   
 $\quad \text{a}[\text{exp } (\text{hom}_o b) (\text{hom}_o c), \text{exp } (\text{hom}_o a) (\text{hom}_o b),$   
 $\quad \text{hom}_o a]) \cdot$   
 $(\text{hom}_a b c \otimes \text{hom}_a a b))$

**using** *C.Comp-def by simp*

**also have** ... =

$\text{Uncurry}[\text{hom}_o a, \text{hom}_o c]$   
 $(\text{Curry}[\text{Hom } b c \otimes \text{Hom } a b, \text{hom}_o a, \text{hom}_o c]$   
 $((\text{eval } (\text{hom}_o b) (\text{hom}_o c) \cdot$   
 $(\text{exp } (\text{hom}_o b) (\text{hom}_o c) \otimes \text{eval } (\text{hom}_o a) (\text{hom}_o b)) \cdot$   
 $\text{a}[\text{exp } (\text{hom}_o b) (\text{hom}_o c), \text{exp } (\text{hom}_o a) (\text{hom}_o b),$   
 $\quad \text{hom}_o a]) \cdot$   
 $((\text{hom}_a b c \otimes \text{hom}_a a b) \otimes \text{hom}_o a)))$

**proof** –

**have**  $\llbracket \text{hom}_a b c \otimes \text{hom}_a a b : \text{Hom } b c \otimes \text{Hom } a b \rightarrow$   
 $\text{exp } (\text{hom}_o b) (\text{hom}_o c) \otimes \text{exp } (\text{hom}_o a) (\text{hom}_o b) \rrbracket$

**using**  $x a b c$  \* **by force**  
**moreover have**  $\langle \text{eval } (hom_o b) (hom_o c) \cdot$   
 $(exp (hom_o b) (hom_o c) \otimes eval (hom_o a) (hom_o b)) \cdot$   
 $a[exp (hom_o b) (hom_o c), exp (hom_o a) (hom_o b), hom_o a]$   
 $: (exp (hom_o b) (hom_o c) \otimes exp (hom_o a) (hom_o b))$   
 $\otimes hom_o a$   
 $\rightarrow hom_o c \rangle$

**using**  $x a b c$  **by simp**  
**ultimately show** ?thesis  
**using**  $x a b c$  *C.comp-Curry-arr* **by simp**

**qed**  
**also have** ... =

$$\begin{aligned}
& (eval (hom_o b) (hom_o c) \cdot \\
& (exp (hom_o b) (hom_o c) \otimes eval (hom_o a) (hom_o b)) \cdot \\
& a[exp (hom_o b) (hom_o c), exp (hom_o a) (hom_o b), hom_o a]) \cdot \\
& ((hom_a b c \otimes hom_a a b) \otimes hom_o a)
\end{aligned}$$

**using**  $x a b c$   
*C.Uncurry-Curry*  
[*of Hom b c*  $\otimes$  *Hom a b*  $hom_o a$   $hom_o c$   
 $(eval (hom_o b) (hom_o c) \cdot$   
 $(exp (hom_o b) (hom_o c) \otimes eval (hom_o a) (hom_o b)) \cdot$   
 $a[exp (hom_o b) (hom_o c), exp (hom_o a) (hom_o b), hom_o a]) \cdot$   
 $((Curry[Hom b c, hom_o b, hom_o c] (Comp x b c) \otimes$   
 $Curry[Hom a b, hom_o a, hom_o b] (Comp x a b))$   
 $\otimes hom_o a)]$

**by fastforce**  
**also have** ... =

$$\begin{aligned}
& eval (hom_o b) (hom_o c) \cdot \\
& (exp (hom_o b) (hom_o c) \otimes eval (hom_o a) (hom_o b)) \cdot \\
& a[exp (hom_o b) (hom_o c), exp (hom_o a) (hom_o b), hom_o a] \cdot \\
& ((hom_a b c \otimes hom_a a b) \otimes hom_o a)
\end{aligned}$$

**by (simp add: comp-assoc)**  
**also have** ... =

$$\begin{aligned}
& eval (hom_o b) (hom_o c) \cdot \\
& ((exp (hom_o b) (hom_o c) \otimes eval (hom_o a) (hom_o b)) \cdot \\
& (hom_a b c \otimes hom_a a b \otimes hom_o a)) \cdot \\
& a[Hom b c, Hom a b, hom_o a]
\end{aligned}$$

**using**  $x a b c$  *Comp-in-hom*  
*assoc-naturality*  
[*of Curry[Hom b c, hom\_o b, hom\_o c] (Comp x b c)*  
 $Curry[Hom a b, hom_o a, hom_o b] (Comp x a b)$   
 $hom_o a]$

**using comp-assoc by auto**  
**also have** ... =

$$\begin{aligned}
& eval (hom_o b) (hom_o c) \cdot \\
& (exp (hom_o b) (hom_o c) \cdot \\
& hom_a b c \otimes Uncurry[hom_o a, hom_o b] (hom_a a b)) \cdot \\
& a[Hom b c, Hom a b, hom_o a]
\end{aligned}$$

**using**  $x a b c$  *Comp-in-hom interchange* **by simp**

**also have** ... =  
 $eval (hom_o b) (hom_o c) \cdot$   
 $(exp (hom_o b) (hom_o c) \cdot hom_a b c \otimes Comp x a b) \cdot$   
 $a[Hom b c, Hom a b, hom_o a]$   
**using**  $x a b c C.Uncurry-Curry Comp-in-hom$  **by** *auto*  
**also have** ... =  
 $eval (hom_o b) (hom_o c) \cdot (hom_a b c \otimes Comp x a b) \cdot$   
 $a[Hom b c, Hom a b, hom_o a]$   
**using**  $x a b c$   
**by** (*simp add: Comp-in-hom comp-ide-arr*)  
**also have** ... =  
 $eval (hom_o b) (hom_o c) \cdot$   
 $((hom_a b c \otimes hom_o b) \cdot (Hom b c \otimes Comp x a b)) \cdot$   
 $a[Hom b c, Hom a b, hom_o a]$   
**proof** –  
**have**  $seq (hom_a b c) (Hom b c)$   
**using**  $x a b c Comp-in-hom C.Curry-in-hom ide-Hom$  **by** *simp*  
**moreover have**  $seq (hom_o b) (Comp x a b)$   
**using**  $x a b c Comp-in-hom$  **by** *fastforce*  
**ultimately show** *?thesis*  
**using**  $x a b c Comp-in-hom C.Curry-in-hom comp-arr-ide$   
 $comp-ide-arr ide-Hom interchange$   
**by** *metis*  
**qed**  
**also have** ... =  
 $Uncurry[hom_o b, hom_o c] (hom_a b c) \cdot$   
 $(Hom b c \otimes Comp x a b) \cdot$   
 $a[Hom b c, Hom a b, hom_o a]$   
**using** *comp-assoc* **by** *simp*  
**also have** ... =  $Comp x a c \cdot (Comp a b c \otimes hom_o a)$   
**using**  $x a b c C.Uncurry-Curry Comp-in-hom Comp-assoc$  **by** *auto*  
**also have** ... =  $Uncurry[hom_o a, hom_o c]$   
 $(Curry[Hom b c \otimes Hom a b, hom_o a, hom_o c]$   
 $(Comp x a c \cdot (Comp a b c \otimes hom_o a)))$   
**using**  $x a b c Comp-in-hom comp-assoc$   
 $C.Uncurry-Curry$   
 $[of Hom b c \otimes Hom a b hom_o a hom_o c$   
 $Comp x a c \cdot (Comp a b c \otimes hom_o a)]$   
**by** *fastforce*  
**also have** ... =  $Uncurry[hom_o a, hom_o c] (hom_a a c \cdot Comp a b c)$   
**using**  $x a b c Comp-in-hom$   
 $C.comp-Curry-arr$   
 $[of hom_o a Comp a b c Hom b c \otimes Hom a b$   
 $Hom a c Comp x a c hom_o c]$   
**by** *auto*  
**finally show** *?thesis* **by** *blast*  
**qed**  
**hence**  $Curry[Hom b c \otimes Hom a b, hom_o a, hom_o c]$   
 $(Uncurry[hom_o a, hom_o c]$

$$\begin{aligned}
& (C.Comp (hom_o a) (hom_o b) (hom_o c) \cdot \\
& \quad (hom_a b c \otimes hom_a a b))) = \\
& Curry[Hom b c \otimes Hom a b, hom_o a, hom_o c] \\
& \quad (Uncurry[hom_o a, hom_o c] (hom_a a c \cdot Comp a b c)) \\
\text{by } & \text{simp} \\
\text{thus } & C.Comp (hom_o a) (hom_o b) (hom_o c) \cdot (hom_a b c \otimes hom_a a b) = \\
& \quad hom_a a c \cdot Comp a b c \\
\text{using } & x a b c \text{ Comp-in-hom} \\
& C.Curry-Uncurry \\
& \quad [of Hom b c \otimes Hom a b hom_o a hom_o c hom_a a c \cdot Comp a b c] \\
& C.Curry-Uncurry \\
& \quad [of Hom b c \otimes Hom a b hom_o a hom_o c \\
& \quad \quad C.Comp (hom_o a) (hom_o b) (hom_o c) \cdot (hom_a b c \otimes hom_a a b)] \\
& \text{by } auto \\
& \text{qed} \\
& \text{qed}
\end{aligned}$$

**lemma** *is-enriched-functor*:  
**shows** *enriched-functor*  $C T \alpha \iota$   
*Obj Hom Id Comp*  
*(Collect ide) exp C.Id C.Comp*  
*hom\_o hom\_a*

..

**sublocale**  $C_0$ : *underlying-category*  $C T \alpha \iota \langle \text{Collect ide} \rangle \text{exp } C.Id C.Comp$  ..  
**sublocale**  $UC$ : *underlying-category*  $C T \alpha \iota \text{Obj Hom Id Comp}$  ..  
**sublocale**  $UF$ : *underlying-functor*  $C T \alpha \iota$   
*Obj Hom Id Comp*  
*\langle Collect ide \rangle exp C.Id C.Comp*  
*hom\_o hom\_a*

..

The following is Kelly's formula (1.31), for the result of applying the ordinary functor underlying the covariant hom functor, to an arrow  $g : \mathcal{I} \rightarrow Hom b c$  of  $C_0$ , resulting in an arrow  $Hom^{\rightarrow} x g : Hom x b \rightarrow Hom x c$  of  $C$ . The point of the result is that this can be expressed explicitly as  $Comp x b c \cdot (g \otimes hom_o b) \cdot 1^{-1}[hom_o b]$ . This is all very confusing at first, because Kelly identifies  $C$  with the underlying category  $C_0$  of  $C$  regarded as a self-enriched category, whereas here we cannot ignore the fact that they are merely isomorphic via  $C.frmUC: UC.comp \rightarrow C_0.comp$ . There is also the bother that, for an arrow  $g : \mathcal{I} \rightarrow Hom b c$  of  $C$ , the corresponding arrow of the underlying category  $UC$  has to be formally constructed using  $UC.MkArr$ , i.e. as  $UC.MkArr b c g$ .

**lemma** *Kelly-1-31*:  
**assumes**  $b \in Obj$  **and**  $c \in Obj$  **and**  $\langle g : \mathcal{I} \rightarrow Hom b c \rangle$   
**shows**  $C.frmUC (UF.map_0 (UC.MkArr b c g)) =$   
 $Comp x b c \cdot (g \otimes hom_o b) \cdot 1^{-1}[hom_o b]$   
**proof** –

**have**  $C.frmUC (UF.map_0 (UC.MkArr b c g)) =$   
 $(Curry[Hom b c, hom_o b, hom_o c] (Comp x b c) \cdot g) \downarrow[hom_o b, hom_o c]$   
**using** *assms x ide-Hom UF.map\_0-def*  
 $C.UC.arr-MkArr$   
 $[of Hom x b Hom x c$   
 $Curry[Hom b c, Hom x b, Hom x c] (Comp x b c) \cdot g]$   
**by** *fastforce*  
**also have**  $\dots = C.Uncurry (Hom x b) (Hom x c)$   
 $(Curry[\mathcal{I}, Hom x b, Hom x c]$   
 $(Comp x b c \cdot (g \otimes Hom x b))) \cdot 1^{-1}[hom_o b]$   
**using** *assms x C.comp-Curry-arr C.DN-def*  
**by** (*metis Comp-in-hom C.Curry-simps(1-2) in-homE seqI ide-Hom*)  
**also have**  $\dots = (Comp x b c \cdot (g \otimes Hom x b)) \cdot 1^{-1}[hom_o b]$   
**using** *assms x ide-Hom ide-unity*  
 $C.Uncurry-Curry$   
 $[of \mathcal{I} Hom x b Hom x c Comp x b c \cdot (g \otimes Hom x b)]$   
**by** *fastforce*  
**also have**  $\dots = Comp x b c \cdot (g \otimes Hom x b) \cdot 1^{-1}[hom_o b]$   
**using** *comp-assoc by simp*  
**finally show** *?thesis by blast*  
**qed**

**abbreviation**  $map_0$   
**where**  $map_0 b c g \equiv Comp x b c \cdot (g \otimes Hom x b) \cdot 1^{-1}[hom_o b]$

**end**

**context** *elementary-closed-monoidal-category*  
**begin**

**lemma** *cov-Exp-DN*:

**assumes**  $\langle g : \mathcal{I} \rightarrow exp a b \rangle$

**and** *ide a* **and** *ide b* **and** *ide x*

**shows**  $Exp^{\rightarrow} x (g \downarrow[a, b]) =$

$$(Curry[exp a b, exp x a, exp x b] (Comp x a b) \cdot g) \downarrow[exp x a, exp x b]$$

**proof** –

**have**  $(Curry[exp a b, exp x a, exp x b] (Comp x a b) \cdot g) \downarrow[exp x a, exp x b] =$

$$Uncurry[exp x a, exp x b]$$

$$(Curry[\mathcal{I}, exp x a, exp x b] (Comp x a b \cdot (g \otimes exp x a))) \cdot 1^{-1}[exp x a]$$

**using** *assms DN-def*

*comp-Curry-arr*

$$[of exp x a g \mathcal{I} exp a b Comp x a b exp x b]$$

**by** *force*

**also have**  $\dots = (Comp x a b \cdot (g \otimes exp x a)) \cdot 1^{-1}[exp x a]$

**using** *assms Uncurry-Curry by auto*

**also have**  $\dots = Curry[exp a b \otimes exp x a, x, b]$

$$(eval a b \cdot (exp a b \otimes eval x a) \cdot a[exp a b, exp x a, x]) \cdot$$

$(g \otimes \text{exp } x \ a) \cdot \text{l}^{-1}[\text{exp } x \ a]$

**unfolding** *Comp-def*

**using** *assms comp-assoc by auto*

**also have** ... =  $\text{Curry}[\text{exp } x \ a, x, b]$   
 $((\text{eval } a \ b \cdot (\text{exp } a \ b \otimes \text{eval } x \ a) \cdot \text{a}[\text{exp } a \ b, \text{exp } x \ a, x]) \cdot$   
 $((g \otimes \text{exp } x \ a) \cdot \text{l}^{-1}[\text{exp } x \ a] \otimes x))$

**using** *assms comp-Curry-arr by auto*

**also have** ... =  $\text{Curry}[\text{exp } x \ a, x, b]$   
 $(\text{eval } a \ b \cdot (\text{exp } a \ b \otimes \text{eval } x \ a) \cdot$   
 $(\text{a}[\text{exp } a \ b, \text{exp } x \ a, x] \cdot ((g \otimes \text{exp } x \ a) \otimes x)) \cdot$   
 $(\text{l}^{-1}[\text{exp } x \ a] \otimes x))$

**using** *assms comp-arr-dom comp-cod-arr interchange comp-assoc by fastforce*

**also have** ... =  $\text{Curry}[\text{exp } x \ a, x, b]$   
 $(\text{eval } a \ b \cdot (\text{exp } a \ b \otimes \text{eval } x \ a) \cdot$   
 $((g \otimes \text{exp } x \ a \otimes x) \cdot \text{a}[\mathcal{I}, \text{exp } x \ a, x]) \cdot$   
 $(\text{l}^{-1}[\text{exp } x \ a] \otimes x))$

**using** *assms assoc-naturality [of g exp x a x] by auto*

**also have** ... =  $\text{Curry}[\text{exp } x \ a, x, b]$   
 $(\text{eval } a \ b \cdot ((\text{exp } a \ b \otimes \text{eval } x \ a) \cdot (g \otimes \text{exp } x \ a \otimes x)) \cdot$   
 $\text{a}[\mathcal{I}, \text{exp } x \ a, x] \cdot (\text{l}^{-1}[\text{exp } x \ a] \otimes x))$

**using** *assms comp-assoc by simp*

**also have** ... =  $\text{Curry}[\text{exp } x \ a, x, b]$   
 $(\text{eval } a \ b \cdot ((g \otimes a) \cdot (\mathcal{I} \otimes \text{eval } x \ a)) \cdot$   
 $\text{a}[\mathcal{I}, \text{exp } x \ a, x] \cdot (\text{l}^{-1}[\text{exp } x \ a] \otimes x))$

**using** *assms comp-arr-dom comp-cod-arr interchange by auto*

**also have** ... =  $\text{Curry}[\text{exp } x \ a, x, b]$   
 $(\text{Uncurry}[a, b] \ g \cdot (\mathcal{I} \otimes \text{eval } x \ a) \cdot \text{l}^{-1}[\text{exp } x \ a \otimes x])$

**using** *assms lunit-tensor inv-comp comp-assoc by simp*

**also have** ... =  $\text{Exp}^{\rightarrow} x (g \downarrow[a, b])$

**using** *assms lunit'-naturality [of eval x a] comp-assoc DN-def by auto*

**finally show** *?thesis by simp*

**qed**

**end**

## 2.4.2 Contravariant Case

**locale** *contravariant-Hom* =

*symmetric-monoidal-category* +

*C: elementary-closed-symmetric-monoidal-category* +

*enriched-category* +

**fixes**  $y :: 'o$

**assumes**  $y \in \text{Obj}$

**begin**

**interpretation** *C: enriched-category C T α ι <Collect ide> exp C.Id C.Comp*

**using** *C.is-enriched-in-itself by simp*

**interpretation** *C: self-enriched-category C T α ι exp eval Curry ..*

**sublocale** *Op*: *opposite-enriched-category C T α ι σ Obj Hom Id Comp ..*

**abbreviation**  $hom_o$

**where**  $hom_o \equiv \lambda a. Hom\ a\ y$

**abbreviation**  $hom_a$

**where**  $hom_a \equiv \lambda b\ c. \text{if } b \in Obj \wedge c \in Obj$

*then*  $Curry[Hom\ c\ b, Hom\ b\ y, Hom\ c\ y]$  (*Op.Comp<sub>op</sub> y b c*)  
*else null*

**sublocale** *enriched-functor C T α ι*

*Obj Op.Hom<sub>op</sub> Id Op.Comp<sub>op</sub>*

*⟨Collect ide⟩ exp C.Id C.Comp*

*hom\_o hom\_a*

**proof**

**show**  $\bigwedge a\ b. a \notin Obj \vee b \notin Obj \implies hom_a\ a\ b = null$

**by** *auto*

**show**  $\bigwedge x. x \in Obj \implies hom_o\ x \in Collect\ ide$

**using** *y by auto*

**show**  $*$ :  $\bigwedge a\ b. \llbracket a \in Obj; b \in Obj \rrbracket \implies$

$\llbracket hom_a\ a\ b : Hom\ b\ a \rightarrow exp\ (hom_o\ a)\ (hom_o\ b) \rrbracket$

**using** *y C.cnt-Exp-ide C.Curry-in-hom Op.Comp-in-hom [of y] by simp*

**show**  $\bigwedge a. a \in Obj \implies hom_a\ a\ a \cdot Id\ a = C.Id\ (hom_o\ a)$

**using** *y Id-in-hom C.Id-def C.comp-Curry-arr Op.Comp-Id-Hom Op.Comp-in-hom*

**by** *fastforce*

**show**  $\bigwedge a\ b\ c. \llbracket a \in Obj; b \in Obj; c \in Obj \rrbracket \implies$

$C.Comp\ (hom_o\ a)\ (hom_o\ b)\ (hom_o\ c) \cdot$

$(hom_a\ b\ c \otimes hom_a\ a\ b) =$

$hom_a\ a\ c \cdot Op.Comp_{op}\ a\ b\ c$

**proof** –

**fix** *a b c*

**assume** *a*:  $a \in Obj$  **and** *b*:  $b \in Obj$  **and** *c*:  $c \in Obj$

**have**  $C.Comp\ (hom_o\ a)\ (hom_o\ b)\ (hom_o\ c) \cdot (hom_a\ b\ c \otimes hom_a\ a\ b) =$

$Curry[exp\ (hom_o\ b)\ (hom_o\ c) \otimes exp\ (hom_o\ a)\ (hom_o\ b),$

$hom_o\ a, hom_o\ c]$

$(eval\ (hom_o\ b)\ (hom_o\ c) \cdot$

$(exp\ (hom_o\ b)\ (hom_o\ c) \otimes eval\ (hom_o\ a)\ (hom_o\ b)) \cdot$

$a[exp\ (hom_o\ b)\ (hom_o\ c), exp\ (hom_o\ a)\ (hom_o\ b), hom_o\ a]) \cdot$

$(hom_a\ b\ c \otimes hom_a\ a\ b)$

**using** *y a b c comp-assoc C.Comp-def by simp*

**also have**  $\dots = Curry[Hom\ c\ b \otimes Hom\ b\ a, hom_o\ a, hom_o\ c]$

$((eval\ (hom_o\ b)\ (hom_o\ c) \cdot$

$(exp\ (hom_o\ b)\ (hom_o\ c) \otimes eval\ (hom_o\ a)\ (hom_o\ b)) \cdot$

$a[exp\ (hom_o\ b)\ (hom_o\ c), exp\ (hom_o\ a)\ (hom_o\ b),$

$hom_o\ a]) \cdot$

$((hom_a\ b\ c \otimes hom_a\ a\ b) \otimes hom_o\ a))$

**using** *y a b c*

*C.comp-Curry-arr*

*[of Hom a y hom\_a b c \otimes hom\_a a b Hom c b \otimes Hom b a*

$$\begin{aligned} & \text{exp } (hom_o b) (hom_o c) \otimes \text{exp } (hom_o a) (hom_o b) \\ & \text{eval } (hom_o b) (hom_o c) \cdot \\ & \quad (\text{exp } (hom_o b) (hom_o c) \otimes \text{eval } (hom_o a) (hom_o b)) \cdot \\ & \quad \text{a}[\text{exp } (hom_o b) (hom_o c), \text{exp } (hom_o a) (hom_o b), hom_o a] \\ & \quad hom_o c] \end{aligned}$$

by *fastforce*

also have ... =  $\text{Curry}[\text{Hom } c b \otimes \text{Hom } b a, hom_o a, hom_o c]$   
 $(\text{eval } (hom_o b) (hom_o c) \cdot$   
 $(\text{exp } (hom_o b) (hom_o c) \otimes \text{eval } (hom_o a) (hom_o b)) \cdot$   
 $(hom_a b c \otimes hom_a a b \otimes hom_o a) \cdot$   
 $\text{a}[\text{Hom } c b, \text{Hom } b a, hom_o a])$

using *y a b c Op.Comp-in-hom comp-assoc*  
*C.assoc-naturality*  
 $[\text{of } \text{Curry}[\text{Hom } c b, hom_o b, hom_o c] (\text{Op.Comp}_{op} y b c)$   
 $\text{Curry}[\text{Hom } b a, hom_o a, hom_o b] (\text{Op.Comp}_{op} y a b)$   
 $hom_o a]$

by *auto*

also have ... =  $\text{Curry}[\text{Hom } c b \otimes \text{Hom } b a, hom_o a, hom_o c]$   
 $(\text{eval } (hom_o b) (hom_o c) \cdot$   
 $(\text{exp } (hom_o b) (hom_o c) \cdot hom_a b c \otimes$   
 $\text{Uncurry}[hom_o a, hom_o b] (hom_a a b)) \cdot$   
 $\text{a}[\text{Hom } c b, \text{Hom } b a, hom_o a])$

proof –

have  $\text{seq } (\text{exp } (hom_o b) (hom_o c)) (hom_a b c)$   
using *y a b c by force*

moreover have  $\text{seq } (\text{eval } (hom_o a) (hom_o b)) (hom_a a b \otimes hom_o a)$   
using *y a b c by fastforce*

ultimately show *?thesis*  
using *y a b c comp-assoc*  
*C.interchange*  
 $[\text{of } \text{exp } (\text{Hom } b y) (\text{Hom } c y) hom_a b c$   
 $\text{eval } (\text{Hom } a y) (\text{Hom } b y) hom_a a b \otimes hom_o a]$

by *metis*

qed

also have ... =  $\text{Curry}[\text{Hom } c b \otimes \text{Hom } b a, hom_o a, hom_o c]$   
 $(\text{eval } (hom_o b) (hom_o c) \cdot$   
 $(\text{exp } (hom_o b) (hom_o c) \cdot hom_a b c \otimes \text{Op.Comp}_{op} y a b) \cdot$   
 $\text{a}[\text{Hom } c b, \text{Hom } b a, hom_o a])$

using *y a b c C.Uncurry-Curry Op.Comp-in-hom by auto*

also have ... =  $\text{Curry}[\text{Hom } c b \otimes \text{Hom } b a, hom_o a, hom_o c]$   
 $(\text{eval } (hom_o b) (hom_o c) \cdot$   
 $(hom_a b c \otimes \text{Op.Comp}_{op} y a b) \cdot$   
 $\text{a}[\text{Hom } c b, \text{Hom } b a, hom_o a])$

using *y a b c*

by (*simp add: comp-ide-arr Op.Comp-in-hom*)

also have ... =  $\text{Curry}[\text{Hom } c b \otimes \text{Hom } b a, hom_o a, hom_o c]$   
 $(\text{eval } (hom_o b) (hom_o c) \cdot$   
 $(hom_a b c \cdot \text{Hom } c b \otimes hom_o b \cdot \text{Op.Comp}_{op} y a b) \cdot$   
 $\text{a}[\text{Hom } c b, \text{Hom } b a, hom_o a])$



**using**  $y a b c *$   
**by** (*metis Op.Comp-in-hom comp-cod-arr comp-arr-dom in-homE*)  
**also have** ... =  $Curry[Hom\ c\ b \otimes Hom\ b\ a, hom_o\ a, hom_o\ c]$   
 $(eval\ (hom_o\ b)\ (hom_o\ c) \cdot$   
 $((hom_a\ b\ c \otimes hom_o\ b) \cdot (Hom\ c\ b \otimes Op.Comp_{op}\ y\ a\ b)) \cdot$   
 $a[Hom\ c\ b, Hom\ b\ a, hom_o\ a])$   
**using**  $y a b c *$   
 $C.interchange\ [of\ hom_a\ b\ c\ Hom\ c\ b\ hom_o\ b\ Op.Comp_{op}\ y\ a\ b]$   
**by** (*metis Op.Comp-in-hom ide-Hom ide-char in-homE seqI*)  
**also have** ... =  $Curry[Hom\ c\ b \otimes Hom\ b\ a, hom_o\ a, hom_o\ c]$   
 $(Uncurry[hom_o\ b, hom_o\ c]\ (hom_a\ b\ c) \cdot$   
 $(Hom\ c\ b \otimes Op.Comp_{op}\ y\ a\ b) \cdot$   
 $a[Hom\ c\ b, Hom\ b\ a, hom_o\ a])$   
**using** *comp-assoc* **by** *simp*  
**also have** ... =  $Curry[Hom\ c\ b \otimes Hom\ b\ a, hom_o\ a, hom_o\ c]$   
 $(Op.Comp_{op}\ y\ b\ c \cdot$   
 $(Hom\ c\ b \otimes Op.Comp_{op}\ y\ a\ b) \cdot$   
 $a[Hom\ c\ b, Hom\ b\ a, hom_o\ a])$   
**using**  $y a b c\ C.Uncurry-Curry$   
**by** (*simp add: Op.Comp-in-hom*)  
**also have** ... =  $Curry[Hom\ c\ b \otimes Hom\ b\ a, hom_o\ a, hom_o\ c]$   
 $(Op.Comp_{op}\ y\ a\ c \cdot (Op.Comp_{op}\ a\ b\ c \otimes hom_o\ a))$   
**using**  $y a b c\ Op.Comp-assoc\ [of\ y\ a\ b\ c]$  **by** *simp*  
**also have** ... =  $hom_a\ a\ c \cdot Op.Comp_{op}\ a\ b\ c$   
**using**  $y a b c\ C.comp-Curry-arr\ [of\ Hom\ a\ y\ Op.Comp_{op}\ a\ b\ c]$   
 $ide-Hom\ Op.Comp-in-hom$   
**by** *fastforce*  
**finally**  
**show**  $C.Comp\ (hom_o\ a)\ (hom_o\ b)\ (hom_o\ c) \cdot (hom_a\ b\ c \otimes hom_a\ a\ b) =$   
 $hom_a\ a\ c \cdot Op.Comp_{op}\ a\ b\ c$   
**by** *blast*  
**qed**  
**qed**

**lemma** *is-enriched-functor*:

**shows** *enriched-functor*  $C\ T\ \alpha\ \iota$   
 $Obj\ Op.Hom_{op}\ Id\ Op.Comp_{op}$   
 $(Collect\ ide)\ exp\ C.Id\ C.Comp$   
 $hom_o\ hom_a$

..

**sublocale**  $C_0$ : *underlying-category*  $C\ T\ \alpha\ \iota\ \langle Collect\ ide \rangle\ exp\ C.Id\ C.Comp\ ..$

**sublocale**  $Op_0$ : *underlying-category*  $C\ T\ \alpha\ \iota\ Obj\ Op.Hom_{op}\ Id\ Op.Comp_{op}\ ..$

**sublocale**  $UF$ : *underlying-functor*  $C\ T\ \alpha\ \iota$   
 $Obj\ Op.Hom_{op}\ Id\ Op.Comp_{op}$   
 $\langle Collect\ ide \rangle\ exp\ C.Id\ C.Comp$   
 $hom_o\ hom_a$

..

The following is Kelly's formula (1.32) for  $Hom^{\leftarrow} f y : Hom\ b\ y \rightarrow Hom$

$a \ y$ .

**lemma** *Kelly-1-32*:

**assumes**  $a \in \text{Obj}$  **and**  $b \in \text{Obj}$  **and**  $\langle f : \mathcal{I} \rightarrow \text{Hom } a \ b \rangle$

**shows**  $C.\text{frmUC } (UF.\text{map}_0 (Op_0.\text{MkArr } b \ a \ f)) =$   
 $Comp \ a \ b \ y \cdot (\text{Hom } b \ y \otimes f) \cdot r^{-1}[\text{hom}_o \ b]$

**proof** –

**have**  $C.\text{frmUC } (UF.\text{map}_0 (Op_0.\text{MkArr } b \ a \ f)) =$   
 $(Curry[\text{Hom } a \ b, \text{hom}_o \ b, \text{hom}_o \ a] (Op.\text{Comp}_{op} \ y \ b \ a) \cdot f)$   
 $\downarrow[\text{hom}_o \ b, \text{hom}_o \ a]$

**proof** –

**have**  $C.UC.\text{arr } (Op_0.\text{MkArr } (\text{Hom } b \ y) (\text{Hom } a \ y))$   
 $(Curry[\text{Hom } a \ b, \text{Hom } b \ y, \text{Hom } a \ y] (Op.\text{Comp}_{op} \ y \ b \ a) \cdot f)$   
**using** *assms y ide-Hom*  
**apply** *auto[1]*  
**using**  $C.UC.\text{arr-MkArr}$   
 $[of \ \text{Hom } b \ y \ \text{Hom } a \ y$   
 $Curry[\text{Hom } a \ b, \text{hom}_o \ b, \text{hom}_o \ a] (Op.\text{Comp}_{op} \ y \ b \ a) \cdot f]$

**by** *blast*

**thus** *?thesis*

**using** *assms UF.map\_0-def Op\_0.arr-MkArr UF.preserves-arr* **by** *auto*

**qed**

**also have**  $1: \dots = Curry[\mathcal{I}, \text{hom}_o \ b, \text{hom}_o \ a]$   
 $(Op.\text{Comp}_{op} \ y \ b \ a \cdot (f \otimes \text{hom}_o \ b)) \downarrow[\text{hom}_o \ b, \text{hom}_o \ a]$

**proof** –

**have**  $Curry[\text{Hom } a \ b, \text{Hom } b \ y, \text{Hom } a \ y] (Op.\text{Comp}_{op} \ y \ b \ a) \cdot f =$   
 $Curry[\mathcal{I}, \text{Hom } b \ y, \text{Hom } a \ y] (Op.\text{Comp}_{op} \ y \ b \ a \cdot (f \otimes \text{Hom } b \ y))$   
**using** *assms y C.comp-Curry-arr* **by** *blast*

**thus** *?thesis* **by** *simp*

**qed**

**also have**  $\dots = (Op.\text{Comp}_{op} \ y \ b \ a \cdot (f \otimes \text{Hom } b \ y)) \cdot l^{-1}[\text{hom}_o \ b]$

**proof** –

**have**  $\text{arr } (Curry[\mathcal{I}, \text{Hom } b \ y, \text{Hom } a \ y]$   
 $(Op.\text{Comp}_{op} \ y \ b \ a \cdot (f \otimes \text{Hom } b \ y)))$

**using** *assms y ide-Hom C.ide-unity*

**by** (*metis 1 C.Curry-simps(1-3) C.DN-def C.DN-simps(1) cod-comp*  
*dom-comp in-homE not-arr-null seqI Op.Comp-in-hom*)

**thus** *?thesis*

**unfolding**  $C.DN-def$

**using** *assms y ide-Hom C.ide-unity*

$C.Uncurry-Curry$

$[of \ \mathcal{I} \ \text{Hom } b \ y \ \text{Hom } a \ y \ Op.\text{Comp}_{op} \ y \ b \ a \cdot (f \otimes \text{Hom } b \ y)]$

**apply** *auto[1]*

**by** *fastforce*

**qed**

**also have**  $\dots =$

$Comp \ a \ b \ y \cdot (s[\text{Hom } a \ b, \text{Hom } b \ y] \cdot (f \otimes \text{Hom } b \ y)) \cdot l^{-1}[\text{hom}_o \ b]$

**using** *comp-assoc* **by** *simp*

**also have**  $\dots = Comp \ a \ b \ y \cdot ((\text{Hom } b \ y \otimes f) \cdot s[\mathcal{I}, \text{Hom } b \ y]) \cdot l^{-1}[\text{hom}_o \ b]$

**using** *assms y C.sym-naturality [of f Hom b y]* **by** *auto*

```

also have ... = Comp a b y · (Hom b y ⊗ f) · s[ $\mathcal{I}$ , Hom b y] · l-1[homo b]
using comp-assoc by simp
also have ... = Comp a b y · (Hom b y ⊗ f) · r-1[homo b]
proof –
  have r-1[homo b] = inv (l[homo b] · s[Hom b y,  $\mathcal{I}$ ])
    using assms y unitor-coherence by simp
  also have ... = s[ $\mathcal{I}$ , Hom b y] · l-1[homo b]
    using assms y
  by (metis C.ide-unity inv-comp-left(1) inverse-unique
    C.iso-lunit C.iso-runit C.sym-inverse C.unitor-coherence
    Op.ide-Hom)
  finally show ?thesis by simp
qed
finally show ?thesis by blast
qed

abbreviation map0
where map0 a b f ≡ Comp a b y · (Hom b y ⊗ f) · r-1[homo b]

end

context elementary-closed-symmetric-monoidal-category
begin

interpretation enriched-category C T α ι <Collect ide> exp Id Comp
  using is-enriched-in-itself by simp
interpretation self-enriched-category C T α ι exp eval Curry ..

sublocale Op: opposite-enriched-category C T α ι σ <Collect ide> exp Id Comp
  ..

lemma cnt-Exp-DN:
assumes «f :  $\mathcal{I} \rightarrow \text{exp } a \ b$ »
and ide a and ide b and ide y
shows Exp← (f ↓[a, b]) y =
  (Curry[exp a b, exp b y, exp a y] (Op.Compop y b a) · f)
  ↓[exp b y, exp a y]
proof –
  have (Curry[exp a b, exp b y, exp a y] (Op.Compop y b a) · f)
    ↓[exp b y, exp a y] =
    Uncurry[exp b y, exp a y]
    (Curry[ $\mathcal{I}$ , exp b y, exp a y] (Op.Compop y b a · (f ⊗ exp b y))) ·
    l-1[exp b y]
  using assms Op.Comp-in-hom DN-def comp-Curry-arr by force
  also have ... = (Op.Compop y b a · (f ⊗ exp b y)) · l-1[exp b y]
  using assms Uncurry-Curry by auto
  also have ... = Comp a b y · (s[exp a b, exp b y] · (f ⊗ exp b y)) ·
    l-1[exp b y]
  using comp-assoc by simp

```

**also have** ... =  $Comp\ a\ b\ y \cdot ((exp\ b\ y \otimes f) \cdot s[\mathcal{I}, exp\ b\ y]) \cdot l^{-1}[exp\ b\ y]$   
**using** *assms sym-naturality [of f exp b y]* **by** *auto*  
**also have** ... =  $Comp\ a\ b\ y \cdot (exp\ b\ y \otimes f) \cdot s[\mathcal{I}, exp\ b\ y] \cdot l^{-1}[exp\ b\ y]$   
**using** *comp-assoc by simp*  
**also have** ... =  $Comp\ a\ b\ y \cdot (exp\ b\ y \otimes f) \cdot r^{-1}[exp\ b\ y]$   
**proof** –  
**have**  $r^{-1}[exp\ b\ y] = inv\ (l[exp\ b\ y] \cdot s[exp\ b\ y, \mathcal{I}])$   
**using** *assms unitor-coherence by auto*  
**also have** ... =  $inv\ s[exp\ b\ y, \mathcal{I}] \cdot l^{-1}[exp\ b\ y]$   
**using** *assms inv-comp by simp*  
**also have** ... =  $s[\mathcal{I}, exp\ b\ y] \cdot l^{-1}[exp\ b\ y]$   
**using** *assms*  
**by** (*metis ide-exp ide-unity inverse-unique sym-inverse*)  
**finally show** *?thesis by simp*  
**qed**  
**also have** ... =  $Curry[exp\ b\ y \otimes exp\ a\ b, a, y]$   
 $(eval\ b\ y \cdot (exp\ b\ y \otimes eval\ a\ b) \cdot a[exp\ b\ y, exp\ a\ b, a]) \cdot$   
 $(exp\ b\ y \otimes f) \cdot r^{-1}[exp\ b\ y]$   
**unfolding** *Comp-def by simp*  
**also have** ... =  $Curry[exp\ b\ y, a, y]$   
 $((eval\ b\ y \cdot (exp\ b\ y \otimes eval\ a\ b) \cdot a[exp\ b\ y, exp\ a\ b, a]) \cdot$   
 $((exp\ b\ y \otimes f) \cdot r^{-1}[exp\ b\ y] \otimes a))$   
**using** *assms*  
*comp-Curry-arr*  
 $[of\ a\ (exp\ b\ y \otimes f) \cdot r^{-1}[exp\ b\ y]\ exp\ b\ y\ exp\ b\ y \otimes exp\ a\ b$   
 $eval\ b\ y \cdot (exp\ b\ y \otimes eval\ a\ b) \cdot a[exp\ b\ y, exp\ a\ b, a]\ y]$   
**by** *auto*  
**also have** ... =  $Curry[exp\ b\ y, a, y]$   
 $((eval\ b\ y \cdot (exp\ b\ y \otimes eval\ a\ b) \cdot a[exp\ b\ y, exp\ a\ b, a]) \cdot$   
 $((exp\ b\ y \otimes f) \otimes a) \cdot (r^{-1}[exp\ b\ y] \otimes a))$   
**using** *assms comp-arr-dom comp-cod-arr interchange by auto*  
**also have** ... =  $Curry[exp\ b\ y, a, y]$   
 $(eval\ b\ y \cdot (exp\ b\ y \otimes eval\ a\ b) \cdot$   
 $(a[exp\ b\ y, exp\ a\ b, a] \cdot ((exp\ b\ y \otimes f) \otimes a)) \cdot$   
 $(r^{-1}[exp\ b\ y] \otimes a))$   
**using** *comp-assoc by simp*  
**also have** ... =  $Curry[exp\ b\ y, a, y]$   
 $(eval\ b\ y \cdot (exp\ b\ y \otimes eval\ a\ b) \cdot$   
 $((exp\ b\ y \otimes f \otimes a) \cdot a[exp\ b\ y, \mathcal{I}, a]) \cdot$   
 $(r^{-1}[exp\ b\ y] \otimes a))$   
**using** *assms assoc-naturality [of exp b y f a]* **by** *auto*  
**also have** ... =  $Curry[exp\ b\ y, a, y]$   
 $(eval\ b\ y \cdot$   
 $((exp\ b\ y \otimes eval\ a\ b) \cdot (exp\ b\ y \otimes f \otimes a)) \cdot$   
 $a[exp\ b\ y, \mathcal{I}, a] \cdot (r^{-1}[exp\ b\ y] \otimes a))$   
**using** *comp-assoc by simp*  
**also have** ... =  $Curry[exp\ b\ y, a, y]$   
 $(eval\ b\ y \cdot (exp\ b\ y \otimes Uncurry[a, b]\ f) \cdot$   
 $a[exp\ b\ y, \mathcal{I}, a] \cdot (r^{-1}[exp\ b\ y] \otimes a))$

```

using assms comp-arr-dom comp-cod-arr interchange by simp
also have ... = Curry[exp b y, a, y]
                (eval b y · (exp b y ⊗ Uncurry[a, b] f) · (exp b y ⊗ 1-1[a]))
proof –
  have exp b y ⊗ 1-1[a] = inv ((r[exp b y] ⊗ a) · a-1[exp b y, I, a])
    using assms triangle' inv-inv iso-inv-iso
    by (metis ide-exp ide-is-iso inv-ide inv-tensor iso-lunit)
  also have ... = a[exp b y, I, a] · (r-1[exp b y] ⊗ a)
    using assms inv-comp by simp
  finally show ?thesis by simp
qed
also have ... = Curry[exp b y, a, y]
                (eval b y · (exp b y ⊗ Uncurry[a, b] f · 1-1[a]))
    using assms comp-arr-dom comp-cod-arr interchange by fastforce
also have ... = Exp← (f↓[a, b]) y
    using assms DN-def by auto
finally show ?thesis by simp
qed

end

```

## 2.5 Enriched Yoneda Lemma

In this section we prove the (weak) Yoneda lemma for enriched categories, as in Kelly, Section 1.9. The weakness is due to the fact that the lemma asserts only a bijection between sets, rather than an isomorphism of objects of the underlying base category.

### 2.5.1 Preliminaries

The following gives conditions under which  $\tau$  defined as  $\tau x = (\mathcal{T} x)^\dagger$  yields an enriched natural transformation between enriched functors  $F$  and  $G$  to the self-enriched base category.

```

context elementary-closed-monoidal-category
begin

```

```

lemma transformation-lam-UP:

```

```

assumes enriched-functor C T α ι

```

```

                ObjA HomA IdA CompA (Collect ide) exp Id Comp Fo Fa

```

```

assumes enriched-functor C T α ι

```

```

                ObjA HomA IdA CompA (Collect ide) exp Id Comp Go Ga

```

```

and  $\bigwedge x. x \notin \text{Obj}_A \implies \mathcal{T} x = \text{null}$ 

```

```

and  $\bigwedge x. x \in \text{Obj}_A \implies \llbracket \mathcal{T} x : F_o x \rightarrow G_o x \rrbracket$ 

```

```

and  $\bigwedge a b. \llbracket a \in \text{Obj}_A; b \in \text{Obj}_A \rrbracket \implies$ 

```

```

                 $\mathcal{T} b \cdot \text{Uncurry}[F_o a, F_o b] (F_a a b) =$ 

```

```

                 $\text{eval} (G_o a) (G_o b) \cdot (G_a a b \otimes \mathcal{T} a)$ 

```

```

shows enriched-natural-transformation C T α ι

```

$Obj_A Hom_A Id_A Comp_A (Collect\ ide)\ exp\ Id\ Comp$   
 $F_o F_a G_o G_a (\lambda x. (\mathcal{T} x)^\dagger)$

**proof** –

**interpret**  $F$ : *enriched-functor*  $C\ T\ \alpha\ \iota$   
 $Obj_A Hom_A Id_A Comp_A \langle Collect\ ide \rangle\ exp\ Id\ Comp\ F_o\ F_a$   
**using** *assms(1)* **by** *blast*

**interpret**  $G$ : *enriched-functor*  $C\ T\ \alpha\ \iota$   
 $Obj_A Hom_A Id_A Comp_A \langle Collect\ ide \rangle\ exp\ Id\ Comp\ G_o\ G_a$   
**using** *assms(2)* **by** *blast*

**show** *?thesis*

**proof**

**show**  $\bigwedge x. x \notin Obj_A \implies (\mathcal{T} x)^\dagger = null$   
**unfolding** *UP-def*  
**using** *assms(3)* **by** *auto*

**show**  $\bigwedge x. x \in Obj_A \implies \langle (\mathcal{T} x)^\dagger : \mathcal{I} \rightarrow exp(F_o x)(G_o x) \rangle$   
**unfolding** *UP-def*  
**using** *assms(4)*  
**apply** *auto[1]*  
**by** *force*

**fix**  $a\ b$

**assume**  $a \in Obj_A$  **and**  $b \in Obj_A$

**have** 1:  $\langle (\mathcal{T} b)^\dagger \otimes F_a a b \rangle \cdot l^{-1}[Hom_A a b]$   
 $: Hom_A a b \rightarrow exp(F_o b)(G_o b) \otimes exp(F_o a)(F_o b)$   
**using** *assms(4)* [of  $b$ ]  $a\ b$  *UP-DN F.preserves-Hom*  
**apply** (*intro comp-in-homI tensor-in-homI*)  
**apply** *auto[5]*  
**by** *fastforce*

**have** 2:  $\langle (G_a a b \otimes (\mathcal{T} a)^\dagger) \cdot r^{-1}[Hom_A a b]$   
 $: Hom_A a b \rightarrow exp(G_o a)(G_o b) \otimes exp(F_o a)(G_o a) \rangle$   
**using** *assms(4)* [of  $a$ ]  $a\ b$  *UP-DN F.preserves-Obj G.preserves-Hom*  
**apply** (*intro comp-in-homI tensor-in-homI*)  
**apply** *auto[5]*  
**by** *fastforce*

**have** 3:  $\langle Comp(F_o a)(F_o b)(G_o b) \cdot ((\mathcal{T} b)^\dagger \otimes F_a a b) \cdot l^{-1}[Hom_A a b]$   
 $: Hom_A a b \rightarrow exp(F_o a)(G_o b) \rangle$   
**using**  $a\ b$  1 *F.preserves-Obj G.preserves-Obj* **by** *blast*

**have** 4:  $\langle Comp(F_o a)(G_o a)(G_o b) \cdot (G_a a b \otimes (\mathcal{T} a)^\dagger) \cdot r^{-1}[Hom_A a b]$   
 $: Hom_A a b \rightarrow exp(F_o a)(G_o b) \rangle$   
**using**  $a\ b$  2 *F.preserves-Obj G.preserves-Obj* **by** *blast*

**have** *Uncurry*[ $F_o a, G_o b$ ]  
 $(Comp(F_o a)(F_o b)(G_o b) \cdot$   
 $((\mathcal{T} b)^\dagger \otimes F_a a b) \cdot l^{-1}[Hom_A a b]) =$   
*Uncurry*[ $F_o a, G_o b$ ]  
 $(Curry[exp(F_o b)(G_o b) \otimes exp(F_o a)(F_o b), F_o a, G_o b]$   
 $(eval(F_o b)(G_o b) \cdot$   
 $(exp(F_o b)(G_o b) \otimes eval(F_o a)(F_o b)) \cdot$   
 $\alpha[exp(F_o b)(G_o b), exp(F_o a)(F_o b), F_o a]) \cdot$   
 $((\mathcal{T} b)^\dagger \otimes F_a a b) \cdot l^{-1}[Hom_A a b])$

**using**  $a\ b$  *Comp-def comp-assoc by auto*  
**also have** ... =  

$$\begin{aligned} & \text{Uncurry}[F_o\ a,\ G_o\ b] \\ & (\text{Curry}[\text{Hom}_A\ a\ b,\ F_o\ a,\ G_o\ b] \\ & ((\text{eval}\ (F_o\ b)\ (G_o\ b)) \cdot \\ & (\text{exp}\ (F_o\ b)\ (G_o\ b) \otimes \text{eval}\ (F_o\ a)\ (F_o\ b)) \cdot \\ & \text{a}[\text{exp}\ (F_o\ b)\ (G_o\ b), \text{exp}\ (F_o\ a)\ (F_o\ b), F_o\ a]) \cdot \\ & (((\mathcal{T}\ b)^\uparrow \otimes F_a\ a\ b) \cdot 1^{-1}[\text{Hom}_A\ a\ b] \otimes F_o\ a))) \end{aligned}$$
**using**  $a\ b\ 1\ F.\text{preserves-Obj}\ G.\text{preserves-Obj}\ \text{comp-Curry-arr}$  **by auto**  
**also have** ... =  

$$\begin{aligned} & (\text{eval}\ (F_o\ b)\ (G_o\ b)) \cdot \\ & (\text{exp}\ (F_o\ b)\ (G_o\ b) \otimes \text{eval}\ (F_o\ a)\ (F_o\ b)) \cdot \\ & \text{a}[\text{exp}\ (F_o\ b)\ (G_o\ b), \text{exp}\ (F_o\ a)\ (F_o\ b), F_o\ a]) \cdot \\ & (((\mathcal{T}\ b)^\uparrow \otimes F_a\ a\ b) \cdot 1^{-1}[\text{Hom}_A\ a\ b] \otimes F_o\ a) \end{aligned}$$
**using**  $a\ b\ 1\ F.\text{preserves-Obj}\ G.\text{preserves-Obj}\ \text{Uncurry-Curry}$  **by auto**  
**also have** ... =  

$$\begin{aligned} & (\text{eval}\ (F_o\ b)\ (G_o\ b)) \cdot \\ & (\text{exp}\ (F_o\ b)\ (G_o\ b) \otimes \text{eval}\ (F_o\ a)\ (F_o\ b)) \cdot \\ & \text{a}[\text{exp}\ (F_o\ b)\ (G_o\ b), \text{exp}\ (F_o\ a)\ (F_o\ b), F_o\ a]) \cdot \\ & (((\mathcal{T}\ b)^\uparrow \otimes F_a\ a\ b) \otimes F_o\ a) \cdot (1^{-1}[\text{Hom}_A\ a\ b] \otimes F_o\ a) \end{aligned}$$
**proof** –  
**have**  $\text{seq}\ ((\mathcal{T}\ b)^\uparrow \otimes F_a\ a\ b)\ 1^{-1}[\text{Hom}_A\ a\ b]$   
**using**  $\text{assms}(4)\ a\ b\ 1\ F.\text{preserves-Hom}$  [of  $a\ b$ ] *UP-DN*  
**apply** (*intro seqI*)  
**apply** *auto[2]*  
**by** (*metis F.A.ide-Hom arrI cod-inv dom-lunit iso-lunit seqE*)  
**thus** *?thesis*  
**using**  $\text{assms}(3)\ a\ b\ F.\text{preserves-Obj}\ F.\text{preserves-Hom}$  *interchange*  
**by** *simp*  
**qed**  
**also have** ... =  

$$\begin{aligned} & \text{eval}\ (F_o\ b)\ (G_o\ b) \cdot \\ & (\text{exp}\ (F_o\ b)\ (G_o\ b) \otimes \text{eval}\ (F_o\ a)\ (F_o\ b)) \cdot \\ & (\text{a}[\text{exp}\ (F_o\ b)\ (G_o\ b), \text{exp}\ (F_o\ a)\ (F_o\ b), F_o\ a]) \cdot \\ & (((\mathcal{T}\ b)^\uparrow \otimes F_a\ a\ b) \otimes F_o\ a) \cdot (1^{-1}[\text{Hom}_A\ a\ b] \otimes F_o\ a) \end{aligned}$$
**using** *comp-assoc by simp*  
**also have** ... =  

$$\begin{aligned} & \text{eval}\ (F_o\ b)\ (G_o\ b) \cdot \\ & (\text{exp}\ (F_o\ b)\ (G_o\ b) \otimes \text{eval}\ (F_o\ a)\ (F_o\ b)) \cdot \\ & (((\mathcal{T}\ b)^\uparrow \otimes F_a\ a\ b \otimes F_o\ a) \cdot \\ & \text{a}[\mathcal{I}, \text{Hom}_A\ a\ b, F_o\ a]) \cdot \\ & (1^{-1}[\text{Hom}_A\ a\ b] \otimes F_o\ a) \end{aligned}$$
**using**  $\text{assms}(4)\ a\ b\ F.\text{preserves-Obj}\ F.\text{preserves-Hom}$   
*assoc-naturality* [of  $(\mathcal{T}\ b)^\uparrow\ F_a\ a\ b\ F_o\ a$ ]  
**by** *force*  
**also have** ... =  

$$\begin{aligned} & \text{eval}\ (F_o\ b)\ (G_o\ b) \cdot \\ & ((\text{exp}\ (F_o\ b)\ (G_o\ b) \otimes \text{eval}\ (F_o\ a)\ (F_o\ b)) \cdot \\ & ((\mathcal{T}\ b)^\uparrow \otimes F_a\ a\ b \otimes F_o\ a)) \cdot \end{aligned}$$

$\mathfrak{a}[\mathcal{I}, \text{Hom}_A a b, F_o a] \cdot (1^{-1}[\text{Hom}_A a b] \otimes F_o a)$

**using** *comp-assoc* **by** *simp*

**also have** ... =

$\text{eval } (F_o b) (G_o b) \cdot$   
 $((\mathcal{T} b)^\dagger \otimes \text{Uncurry}[F_o a, F_o b] (F_a a b)) \cdot$   
 $\mathfrak{a}[\mathcal{I}, \text{Hom}_A a b, F_o a] \cdot (1^{-1}[\text{Hom}_A a b] \otimes F_o a)$

**proof** –

**have** *seq* (*exp* ( $F_o b$ ) ( $G_o b$ )) (*UP* ( $\mathcal{T} b$ ))

**using** *assms*(4) *b F.preserves-Obj G.preserves-Obj* **by** *fastforce*

**moreover have** *seq* (*eval* ( $F_o a$ ) ( $F_o b$ )) ( $F_a a b \otimes F_o a$ )

**using** *a b F.preserves-Obj F.preserves-Hom* **by** *force*

**ultimately show** *?thesis*

**using** *assms*(4) [*of b*] *a b UP-DN(1) comp-cod-arr interchange* **by** *auto*

**qed**

**also have** ... =

$\text{eval } (F_o b) (G_o b) \cdot$   
 $((\mathcal{T} b)^\dagger \otimes F_o b) \cdot (\mathcal{I} \otimes \text{Uncurry}[F_o a, F_o b] (F_a a b)) \cdot$   
 $\mathfrak{a}[\mathcal{I}, \text{Hom}_A a b, F_o a] \cdot (1^{-1}[\text{Hom}_A a b] \otimes F_o a)$

**using** *assms*(4) [*of b*] *a b F.preserves-Obj F.preserves-Hom [of a b]*  
*comp-arr-dom comp-cod-arr [of Uncurry[F\_o a, F\_o b] (F\_a a b)]*  
*interchange [of (\mathcal{T} b)^\dagger \mathcal{I} F\_o b Uncurry[F\_o a, F\_o b] (F\_a a b)]*

**by** *auto*

**also have** ... =

$\text{Uncurry}[F_o b, G_o b] ((\mathcal{T} b)^\dagger) \cdot$   
 $(\mathcal{I} \otimes \text{Uncurry}[F_o a, F_o b] (F_a a b)) \cdot$   
 $\mathfrak{a}[\mathcal{I}, \text{Hom}_A a b, F_o a] \cdot (1^{-1}[\text{Hom}_A a b] \otimes F_o a)$

**using** *comp-assoc* **by** *simp*

**also have** ... = ( $\mathcal{T} b \cdot 1[F_o b]$ ) ·

$(\mathcal{I} \otimes \text{Uncurry}[F_o a, F_o b] (F_a a b)) \cdot$   
 $\mathfrak{a}[\mathcal{I}, \text{Hom}_A a b, F_o a] \cdot (1^{-1}[\text{Hom}_A a b] \otimes F_o a)$

**proof** –

**have**  $\text{Uncurry}[F_o b, G_o b] ((\mathcal{T} b)^\dagger) = \mathcal{T} b \cdot 1[F_o b]$

**unfolding** *UP-def*

**using** *assms*(4) *a b Uncurry-Curry*

**apply** *simp*

**by** (*metis F.preserves-Obj arr-lunit cod-lunit comp-in-homI' dom-lunit*  
*ide-cod ide-unity in-homE mem-Collect-eq*)

**thus** *?thesis* **by** *simp*

**qed**

**also have** ... =  $\mathcal{T} b \cdot (1[F_o b] \cdot (\mathcal{I} \otimes \text{Uncurry}[F_o a, F_o b] (F_a a b))) \cdot$   
 $\mathfrak{a}[\mathcal{I}, \text{Hom}_A a b, F_o a] \cdot (1^{-1}[\text{Hom}_A a b] \otimes F_o a)$

**using** *comp-assoc* **by** *simp*

**also have** ... =  $\mathcal{T} b \cdot (\text{Uncurry}[F_o a, F_o b] (F_a a b) \cdot 1[\text{Hom}_A a b \otimes F_o a])$

$\mathfrak{a}[\mathcal{I}, \text{Hom}_A a b, F_o a] \cdot (1^{-1}[\text{Hom}_A a b] \otimes F_o a)$

**using** *a b lunit-naturality [of Uncurry[F\_o a, F\_o b] (F\_a a b)]*  
*F.preserves-Obj F.preserves-Hom [of a b]*

**by** *auto*

**also have** ... =  $\mathcal{T} b \cdot \text{Uncurry}[F_o a, F_o b] (F_a a b) \cdot$



$l[Hom_A a b \otimes F_o a] \cdot a[\mathcal{I}, Hom_A a b, F_o a] \cdot$   
 $(l^{-1}[Hom_A a b] \otimes F_o a)$

**using** *comp-assoc* **by** *simp*

**also have** ... =  $\mathcal{T} b \cdot Uncurry[F_o a, F_o b] (F_a a b)$

**proof** –

**have**  $l[Hom_A a b \otimes F_o a] \cdot a[\mathcal{I}, Hom_A a b, F_o a] \cdot$   
 $(l^{-1}[Hom_A a b] \otimes F_o a) =$   
 $Hom_A a b \otimes F_o a$

**proof** –

**have**  $l[Hom_A a b \otimes F_o a] \cdot a[\mathcal{I}, Hom_A a b, F_o a] \cdot$   
 $(l^{-1}[Hom_A a b] \otimes F_o a) =$   
 $(l[Hom_A a b] \otimes F_o a) \cdot (l^{-1}[Hom_A a b] \otimes F_o a)$

**using** *a b lunit-tensor'* [of *Hom\_A a b F\_o a*]

**by** (*metis F.A.ide-Hom F.preserves-Obj comp-assoc mem-Collect-eq*)

**also have** ... =  $l[Hom_A a b] \cdot l^{-1}[Hom_A a b] \otimes F_o a \cdot F_o a$

**using** *a b interchange F.preserves-Obj* **by** *force*

**also have** ... =  $Hom_A a b \otimes F_o a$

**using** *a b F.preserves-Obj* **by** *auto*

**finally show** *?thesis* **by** *blast*

**qed**

**thus** *?thesis*

**using** *a b F.preserves-Obj F.preserves-Hom [of a b] comp-arr-dom*

**by** *auto*

**qed**

**finally have** *LHS: Uncurry[F\_o a, G\_o b]*  
 $(Comp (F_o a) (F_o b) (G_o b) \cdot ((\mathcal{T} b)^\dagger \otimes F_a a b) \cdot$   
 $l^{-1}[Hom_A a b]) =$   
 $\mathcal{T} b \cdot Uncurry[F_o a, F_o b] (F_a a b)$

**by** *blast*

**have**  $Uncurry[F_o a, G_o b] (Comp (F_o a) (G_o a) (G_o b) \cdot$   
 $(G_a a b \otimes (\mathcal{T} a)^\dagger) \cdot r^{-1}[Hom_A a b]) =$   
 $Uncurry[F_o a, G_o b]$   
 $(Curry[exp (G_o a) (G_o b) \otimes exp (F_o a) (G_o a), F_o a, G_o b]$   
 $(eval (G_o a) (G_o b) \cdot$   
 $(exp (G_o a) (G_o b) \otimes eval (F_o a) (G_o a)) \cdot$   
 $a[exp (G_o a) (G_o b), exp (F_o a) (G_o a), F_o a]) \cdot$   
 $(G_a a b \otimes (\mathcal{T} a)^\dagger) \cdot r^{-1}[Hom_A a b])$

**using** *a b Comp-def comp-assoc* **by** *auto*

**also have** ... =  
 $Uncurry[F_o a, G_o b]$   
 $(Curry[Hom_A a b, F_o a, G_o b]$   
 $((eval (G_o a) (G_o b) \cdot$   
 $(exp (G_o a) (G_o b) \otimes eval (F_o a) (G_o a)) \cdot$   
 $a[exp (G_o a) (G_o b), exp (F_o a) (G_o a), F_o a]) \cdot$   
 $((G_a a b \otimes (\mathcal{T} a)^\dagger) \cdot r^{-1}[Hom_A a b] \otimes F_o a)))$

**using** *assms(3) a b 2 F.preserves-Obj G.preserves-Obj comp-Curry-arr*

**by** *auto*

**also have** ... =

$(eval (G_o a) (G_o b) \cdot$   
 $(exp (G_o a) (G_o b) \otimes eval (F_o a) (G_o a)) \cdot$   
 $a[exp (G_o a) (G_o b), exp (F_o a) (G_o a), F_o a]) \cdot$   
 $((G_a a b \otimes (\mathcal{T} a)^\uparrow) \cdot r^{-1}[Hom_A a b] \otimes F_o a)$   
**using** *assms(3) a b 2 F.preserves-Obj G.preserves-Obj Uncurry-Curry*  
**by auto**  
**also have ... =**  
 $(eval (G_o a) (G_o b) \cdot$   
 $(exp (G_o a) (G_o b) \otimes eval (F_o a) (G_o a)) \cdot$   
 $a[exp (G_o a) (G_o b), exp (F_o a) (G_o a), F_o a]) \cdot$   
 $((G_a a b \otimes (\mathcal{T} a)^\uparrow) \otimes F_o a) \cdot (r^{-1}[Hom_A a b] \otimes F_o a))$   
**using** *assms(4) a b F.preserves-Obj G.preserves-Hom*  
*interchange [of G\_a a b \otimes (\mathcal{T} a)^\uparrow r^{-1}[Hom\_A a b] F\_o a F\_o a]*  
**by fastforce**  
**also have ... =**  
 $eval (G_o a) (G_o b) \cdot$   
 $(exp (G_o a) (G_o b) \otimes eval (F_o a) (G_o a)) \cdot$   
 $(a[exp (G_o a) (G_o b), exp (F_o a) (G_o a), F_o a]) \cdot$   
 $((G_a a b \otimes (\mathcal{T} a)^\uparrow) \otimes F_o a) \cdot (r^{-1}[Hom_A a b] \otimes F_o a)$   
**using** *comp-assoc by simp*  
**also have ... =**  
 $eval (G_o a) (G_o b) \cdot$   
 $(exp (G_o a) (G_o b) \otimes eval (F_o a) (G_o a)) \cdot$   
 $((G_a a b \otimes (\mathcal{T} a)^\uparrow \otimes F_o a) \cdot$   
 $a[Hom_A a b, \mathcal{I}, F_o a]) \cdot$   
 $(r^{-1}[Hom_A a b] \otimes F_o a)$   
**using** *assms(4) a b F.preserves-Obj G.preserves-Hom*  
*assoc-naturality [of G\_a a b (\mathcal{T} a)^\uparrow F\_o a]*  
**by fastforce**  
**also have ... =**  
 $eval (G_o a) (G_o b) \cdot$   
 $((exp (G_o a) (G_o b) \otimes eval (F_o a) (G_o a)) \cdot$   
 $(G_a a b \otimes (\mathcal{T} a)^\uparrow \otimes F_o a)) \cdot$   
 $a[Hom_A a b, \mathcal{I}, F_o a] \cdot (r^{-1}[Hom_A a b] \otimes F_o a)$   
**using** *comp-assoc by simp*  
**also have ... =**  
 $eval (G_o a) (G_o b) \cdot$   
 $(G_a a b \otimes Uncurry[F_o a, G_o a] ((\mathcal{T} a)^\uparrow)) \cdot$   
 $a[Hom_A a b, \mathcal{I}, F_o a] \cdot (r^{-1}[Hom_A a b] \otimes F_o a)$   
**using** *assms(4) a b F.preserves-Obj G.preserves-Obj*  
*G.preserves-Hom [of a b] comp-cod-arr interchange*  
**by fastforce**  
**also have ... =**  
 $eval (G_o a) (G_o b) \cdot$   
 $((G_a a b \otimes G_o a) \cdot (Hom_A a b \otimes Uncurry[F_o a, G_o a] ((\mathcal{T} a)^\uparrow))) \cdot$   
 $a[Hom_A a b, \mathcal{I}, F_o a] \cdot (r^{-1}[Hom_A a b] \otimes F_o a)$   
**proof –**  
**have** *seq (G\_o a) (Uncurry[F\_o a, G\_o a] ((\mathcal{T} a)^\uparrow))*  
**using** *assms(4) [of a] a b F.preserves-Obj G.preserves-Obj by auto*

**moreover have**  $G_o a \cdot \text{Uncurry}[F_o a, G_o a] ((\mathcal{T} a)^\dagger) =$   
 $\text{Uncurry}[F_o a, G_o a] ((\mathcal{T} a)^\dagger)$   
**using**  $a b F.\text{preserves-Obj } G.\text{preserves-Obj } \text{calculation}(1)$   
 $\text{comp-ide-arr}$   
**by** *blast*  
**ultimately show** *?thesis*  
**using**  $\text{assms}(3) a b G.\text{preserves-Hom [of } a b \text{] interchange}$   
 $\text{comp-arr-dom}$   
**by** *auto*  
**qed**  
**also have**  $\dots =$   
 $\text{Uncurry}[G_o a, G_o b] (G_a a b) \cdot$   
 $(\text{Hom}_A a b \otimes \text{Uncurry}[F_o a, G_o a] ((\mathcal{T} a)^\dagger)) \cdot$   
 $\mathfrak{a}[\text{Hom}_A a b, \mathcal{I}, F_o a] \cdot (\mathfrak{r}^{-1}[\text{Hom}_A a b] \otimes F_o a)$   
**using**  $\text{comp-assoc}$  **by** *simp*  
**also have**  $\dots =$   
 $\text{Uncurry}[G_o a, G_o b] (G_a a b) \cdot$   
 $(\text{Hom}_A a b \otimes \mathcal{T} a \cdot \mathfrak{l}[F_o a]) \cdot$   
 $\mathfrak{a}[\text{Hom}_A a b, \mathcal{I}, F_o a] \cdot (\mathfrak{r}^{-1}[\text{Hom}_A a b] \otimes F_o a)$   
**using**  $\text{assms}(4) [\text{of } a] a b F.\text{preserves-Obj } G.\text{preserves-Obj } \text{UP-def}$   
 $\text{Uncurry-Curry}$   
**by** *auto*  
**also have**  $\dots =$   
 $\text{Uncurry}[G_o a, G_o b] (G_a a b) \cdot$   
 $((\text{Hom}_A a b \otimes \mathcal{T} a) \cdot (\text{Hom}_A a b \otimes \mathfrak{l}[F_o a])) \cdot$   
 $\mathfrak{a}[\text{Hom}_A a b, \mathcal{I}, F_o a] \cdot (\mathfrak{r}^{-1}[\text{Hom}_A a b] \otimes F_o a)$   
**using**  $\text{assms}(4) [\text{of } a] a b F.\text{preserves-Obj } G.\text{preserves-Obj } \text{interchange}$   
**by** *auto*  
**also have**  $\dots =$   
 $\text{Uncurry}[G_o a, G_o b] (G_a a b) \cdot (\text{Hom}_A a b \otimes \mathcal{T} a) \cdot$   
 $(\text{Hom}_A a b \otimes \mathfrak{l}[F_o a]) \cdot \mathfrak{a}[\text{Hom}_A a b, \mathcal{I}, F_o a] \cdot$   
 $(\mathfrak{r}^{-1}[\text{Hom}_A a b] \otimes F_o a)$   
**using**  $\text{comp-assoc}$  **by** *simp*  
**also have**  $\dots = \text{Uncurry}[G_o a, G_o b] (G_a a b) \cdot (\text{Hom}_A a b \otimes \mathcal{T} a)$   
**proof** –  
**have**  $(\text{Hom}_A a b \otimes \mathfrak{l}[F_o a]) \cdot \mathfrak{a}[\text{Hom}_A a b, \mathcal{I}, F_o a] \cdot$   
 $(\mathfrak{r}^{-1}[\text{Hom}_A a b] \otimes F_o a) =$   
 $\text{Hom}_A a b \otimes F_o a$   
**proof** –  
**have**  $(\text{Hom}_A a b \otimes \mathfrak{l}[F_o a]) \cdot \mathfrak{a}[\text{Hom}_A a b, \mathcal{I}, F_o a] \cdot$   
 $(\mathfrak{r}^{-1}[\text{Hom}_A a b] \otimes F_o a) =$   
 $(\mathfrak{r}[\text{Hom}_A a b] \otimes F_o a) \cdot (\mathfrak{r}^{-1}[\text{Hom}_A a b] \otimes F_o a)$   
**using**  $a b \text{ triangle [of } \text{Hom}_A a b F_o a \text{]}$   
**by**  $(\text{metis } F.A.\text{ide-Hom } F.\text{preserves-Obj } \text{comp-assoc mem-Collect-eq})$   
**also have**  $\dots = \mathfrak{r}[\text{Hom}_A a b] \cdot \mathfrak{r}^{-1}[\text{Hom}_A a b] \otimes F_o a \cdot F_o a$   
**using**  $a b \text{ interchange } F.\text{preserves-Obj}$  **by** *force*  
**also have**  $\dots = \text{Hom}_A a b \otimes F_o a$   
**using**  $a b F.\text{preserves-Obj}$  **by** *auto*  
**finally show** *?thesis* **by** *blast*

```

qed
thus ?thesis
  using assms(4) [of a] a b comp-arr-dom by auto
qed
also have ... = eval (Go a) (Go b) · (Ga a b ⊗ Go a) · (HomA a b ⊗ T a)
  using comp-assoc by auto
also have ... = eval (Go a) (Go b) · (Ga a b ⊗ T a)
  using assms(4) a b G.preserves-Hom comp-arr-dom comp-cod-arr
    interchange
  by (metis in-homE)
finally have RHS: Uncurry[Fo a, Go b]
  (Comp (Fo a) (Go a) (Go b) · (Ga a b ⊗ (T a)↑) ·
    r-1[HomA a b]) =
  eval (Go a) (Go b) · (Ga a b ⊗ T a)
by blast

have Uncurry[Fo a, Go b]
  (Comp (Fo a) (Fo b) (Go b) · ((T b)↑ ⊗ Fa a b) · l-1[HomA a b]) =
  Uncurry[Fo a, Go b]
  (Comp (Fo a) (Go a) (Go b) · (Ga a b ⊗ (T a)↑) · r-1[HomA a b])
  using a b assms(5) LHS RHS by simp
moreover have «Comp (Fo a) (Fo b) (Go b) ·
  ((T b)↑ ⊗ Fa a b) · l-1[HomA a b]
  : HomA a b → exp (Fo a) (Go b)»
  using assms(4) a b 1 F.preserves-Obj G.preserves-Obj
    F.preserves-Hom G.preserves-Hom
  apply (intro comp-in-homI' seqI)
  apply auto[1]
  by fastforce+
moreover have «Comp (Fo a) (Go a) (Go b) ·
  (Ga a b ⊗ (T a)↑) · r-1[HomA a b]
  : HomA a b → exp (Fo a) (Go b)»
  using assms(4) a b 2 UP-DN(1) F.preserves-Obj G.preserves-Obj
    F.preserves-Hom G.preserves-Hom [of a b]
  apply (intro comp-in-homI' seqI)
  apply auto[7]
  by fastforce
ultimately show Comp (Fo a) (Fo b) (Go b) ·
  ((T b)↑ ⊗ Fa a b) · l-1[HomA a b] =
  Comp (Fo a) (Go a) (Go b) ·
  (Ga a b ⊗ (T a)↑) · r-1[HomA a b]
  using a b Curry-Uncurry F.A.ide-Hom F.preserves-Obj
    G.preserves-Obj mem-Collect-eq
by metis
qed
qed
end

```

Kelly (1.39) expresses enriched naturality in an alternate form, using

the underlying functors of the covariant and contravariant enriched hom functors.

**locale** *Kelly-1-39* =  
*symmetric-monoidal-category* +  
*elementary-closed-monoidal-category* +  
*enriched-natural-transformation*  
**for**  $a :: 'a$   
**and**  $b :: 'a +$   
**assumes**  $a: a \in \text{Obj}_A$   
**and**  $b: b \in \text{Obj}_A$   
**begin**

**interpretation** *enriched-category*  $C\ T\ \alpha\ \iota\ \langle \text{Collect ide} \rangle\ \text{exp}\ \text{Id}\ \text{Comp}$   
**using** *is-enriched-in-itself* **by** *blast*  
**interpretation** *self-enriched-category*  $C\ T\ \alpha\ \iota\ \text{exp}\ \text{eval}\ \text{Curry}$   
..

**sublocale** *cov-Hom: covariant-Hom*  $C\ T\ \alpha\ \iota$   
*exp eval Curry*  $\text{Obj}_B\ \text{Hom}_B\ \text{Id}_B\ \text{Comp}_B\ \langle F_o\ a \rangle$   
**using**  $a\ F.\text{preserves-Obj}$  **by** *unfold-locale*  
**sublocale** *cnt-Hom: contravariant-Hom*  $C\ T\ \alpha\ \iota\ \sigma$   
*exp eval Curry*  $\text{Obj}_B\ \text{Hom}_B\ \text{Id}_B\ \text{Comp}_B\ \langle G_o\ b \rangle$   
**using**  $b\ G.\text{preserves-Obj}$  **by** *unfold-locale*

**lemma** *Kelly-1-39*:

**shows**  $\text{cov-Hom.map}_0\ (F_o\ b)\ (G_o\ b)\ (\tau\ b) \cdot F_a\ a\ b =$   
 $\text{cnt-Hom.map}_0\ (F_o\ a)\ (G_o\ a)\ (\tau\ a) \cdot G_a\ a\ b$

**proof** –

**have**  $\text{cov-Hom.map}_0\ (F_o\ b)\ (G_o\ b)\ (\tau\ b) \cdot F_a\ a\ b =$   
 $\text{Comp}_B\ (F_o\ a)\ (F_o\ b)\ (G_o\ b) \cdot (\tau\ b \otimes F_a\ a\ b) \cdot 1^{-1}[\text{Hom}_A\ a\ b]$

**proof** –

**have**  $\text{cov-Hom.map}_0\ (F_o\ b)\ (G_o\ b)\ (\tau\ b) \cdot F_a\ a\ b =$   
 $\text{Comp}_B\ (F_o\ a)\ (F_o\ b)\ (G_o\ b) \cdot$   
 $(\tau\ b \otimes \text{Hom}_B\ (F_o\ a)\ (F_o\ b)) \cdot 1^{-1}[\text{Hom}_B\ (F_o\ a)\ (F_o\ b)] \cdot F_a\ a\ b$

**using** *comp-assoc* **by** *simp*

**also have**  $\dots = \text{Comp}_B\ (F_o\ a)\ (F_o\ b)\ (G_o\ b) \cdot$   
 $(\tau\ b \otimes \text{Hom}_B\ (F_o\ a)\ (F_o\ b)) \cdot (\mathcal{I} \otimes F_a\ a\ b) \cdot 1^{-1}[\text{Hom}_A\ a\ b]$

**using**  $a\ b\ \text{lunit}'\text{-naturality}\ F.\text{preserves-Hom}\ [\text{of}\ a\ b]$  **by** *fastforce*

**also have**  $\dots = \text{Comp}_B\ (F_o\ a)\ (F_o\ b)\ (G_o\ b) \cdot$   
 $((\tau\ b \otimes \text{Hom}_B\ (F_o\ a)\ (F_o\ b)) \cdot (\mathcal{I} \otimes F_a\ a\ b)) \cdot$   
 $1^{-1}[\text{Hom}_A\ a\ b]$

**using** *comp-assoc* **by** *simp*

**also have**  $\dots = \text{Comp}_B\ (F_o\ a)\ (F_o\ b)\ (G_o\ b) \cdot (\tau\ b \otimes F_a\ a\ b) \cdot$   
 $1^{-1}[\text{Hom}_A\ a\ b]$

**using**  $a\ b\ \text{component-in-hom}\ [\text{of}\ b]\ F.\text{preserves-Hom}\ [\text{of}\ a\ b]$   
 $\text{comp-arr-dom}\ \text{comp-cod-arr}\ [\text{of}\ F_a\ a\ b\ \text{Hom}_B\ (F_o\ a)\ (F_o\ b)]$   
*interchange*

**by** *fastforce*

**finally show** *?thesis* **by** *blast*

qed  
**moreover have**  $\text{cnt-Hom.map}_0 (F_o a) (G_o a) (\tau a) \cdot G_a a b =$   
 $\text{Comp}_B (F_o a) (G_o a) (G_o b) \cdot (G_a a b \otimes \tau a) \cdot \text{r}^{-1}[\text{Hom}_A a b]$   
**proof** –  
**have**  $\text{cnt-Hom.map}_0 (F_o a) (G_o a) (\tau a) \cdot G_a a b =$   
 $\text{Comp}_B (F_o a) (G_o a) (G_o b) \cdot (\text{Hom}_B (G_o a) (G_o b) \otimes \tau a) \cdot$   
 $\text{r}^{-1}[\text{Hom}_B (G_o a) (G_o b)] \cdot G_a a b$   
**using** *comp-assoc* **by** *simp*  
**also have**  $\dots = \text{Comp}_B (F_o a) (G_o a) (G_o b) \cdot$   
 $(\text{Hom}_B (G_o a) (G_o b) \otimes \tau a) \cdot (G_a a b \otimes \mathcal{I}) \cdot$   
 $\text{r}^{-1}[\text{Hom}_A a b]$   
**using** *a b runit'-naturality* *G.preserves-Hom [of a b]* **by** *fastforce*  
**also have**  $\dots = \text{Comp}_B (F_o a) (G_o a) (G_o b) \cdot$   
 $((\text{Hom}_B (G_o a) (G_o b) \otimes \tau a) \cdot (G_a a b \otimes \mathcal{I})) \cdot$   
 $\text{r}^{-1}[\text{Hom}_A a b]$   
**using** *comp-assoc* **by** *simp*  
**also have**  $\dots = \text{Comp}_B (F_o a) (G_o a) (G_o b) \cdot (G_a a b \otimes \tau a) \cdot$   
 $\text{r}^{-1}[\text{Hom}_A a b]$   
**using** *a b interchange component-in-hom [of a]* *G.preserves-Hom [of a b]*  
*comp-arr-dom comp-cod-arr [of G\_a a b Hom\_B (G\_o a) (G\_o b)]*  
**by** *fastforce*  
**finally show** *?thesis* **by** *blast*  
qed  
**ultimately show** *?thesis*  
**using** *a b naturality* **by** *simp*  
qed  
end

## 2.5.2 Covariant Case

**locale** *covariant-yoneda-lemma* =  
*symmetric-monoidal-category* +  
*C: closed-symmetric-monoidal-category* +  
*covariant-Hom* +  
*F: enriched-functor C T α ι Obj Hom Id Comp ‹Collect ide› exp C.Id C.Comp*  
**begin**

**interpretation** *C: elementary-closed-symmetric-monoidal-category C T α ι σ*  
*exp eval Curry ..*

**interpretation** *C: self-enriched-category C T α ι exp eval Curry ..*

Every element  $e : \mathcal{I} \rightarrow F_o x$  of  $F_o x$  determines an enriched natural transformation  $\tau_e : \text{hom } x \rightarrow F$ . The formula here is Kelly (1.47):  $\tau_e y : \text{hom } x y \rightarrow F y$  is obtained as the composite:

$$\text{hom } x y \xrightarrow{F_a x y} \text{exp } (F x) (F y) \xrightarrow{\text{Exp}^{\leftarrow e} (F y)} \text{exp } \mathcal{I} (F y) \longrightarrow F y$$

where the third component is a canonical isomorphism. This basically amounts to evaluating  $F_a x y$  on element  $e$  of  $F_o x$  to obtain an element of

$F_o y$ .

Note that the above composite gives an arrow  $\tau_e y: \text{hom } x y \rightarrow F y$ , whereas the definition of enriched natural transformation formally requires  $\tau_e y: \mathcal{I} \rightarrow \text{exp } (\text{hom } x y) (F y)$ . So we need to transform the composite to achieve that.

**abbreviation** *generated-transformation*

**where** *generated-transformation*  $e \equiv$

$$\lambda y. (\text{eval } \mathcal{I} (F_o y) \cdot r^{-1}[\text{exp } \mathcal{I} (F_o y)]) \cdot \text{Exp}^{\leftarrow} e (F_o y) \cdot F_a x y)^{\dagger}$$

**lemma** *enriched-natural-transformation-generated-transformation:*

**assumes**  $\langle e : \mathcal{I} \rightarrow F_o x \rangle$

**shows** *enriched-natural-transformation*  $C T \alpha \iota$

$$\text{Obj Hom Id Comp } (\text{Collect ide}) \text{ exp C.Id C.Comp}$$

$$\text{hom}_o \text{ hom}_a F_o F_a \text{ (generated-transformation } e)$$

**proof** (*intro C.transformation-lam-UP*)

**show**  $\bigwedge y. y \notin \text{Obj} \implies$

$$\text{eval } \mathcal{I} (F_o y) \cdot r^{-1}[\text{exp } \mathcal{I} (F_o y)] \cdot \text{Exp}^{\leftarrow} e (F_o y) \cdot F_a x y = \text{null}$$

**by** (*simp add: F.extensionality*)

**show** *enriched-functor*  $(\cdot) T \alpha \iota \text{ Obj Hom Id Comp}$

$$(\text{Collect ide}) \text{ exp C.Id C.Comp hom}_o \text{ hom}_a$$

..

**show** *enriched-functor*  $(\cdot) T \alpha \iota \text{ Obj Hom Id Comp}$

$$(\text{Collect ide}) \text{ exp C.Id C.Comp } F_o F_a$$

..

**show**  $*$ :  $\bigwedge y. y \in \text{Obj} \implies$

$$\begin{aligned} &\langle \text{eval } \mathcal{I} (F_o y) \cdot r^{-1}[\text{exp } \mathcal{I} (F_o y)] \cdot \text{Exp}^{\leftarrow} e (F_o y) \cdot F_a x y \\ &\quad : \text{hom}_o y \rightarrow F_o y \rangle \end{aligned}$$

**using** *assms x F.preserves-Obj F.preserves-Hom*

**apply** (*intro in-homI seqI*)

**apply** *auto[6]*

**by** *fastforce+*

**show**  $\bigwedge a b. \llbracket a \in \text{Obj}; b \in \text{Obj} \rrbracket \implies$

$$\begin{aligned} &(\text{eval } \mathcal{I} (F_o b) \cdot r^{-1}[\text{exp } \mathcal{I} (F_o b)]) \cdot \\ &\quad \text{Exp}^{\leftarrow} e (F_o b) \cdot F_a x b) \cdot \\ &\quad \text{Uncurry}[\text{hom}_o a, \text{hom}_o b] (\text{hom}_a a b) = \\ &\text{eval } (F_o a) (F_o b) \cdot \\ &\quad (F_a a b \otimes \text{eval } \mathcal{I} (F_o a) \cdot r^{-1}[\text{exp } \mathcal{I} (F_o a)]) \cdot \\ &\quad \text{Exp}^{\leftarrow} e (F_o a) \cdot F_a x a \end{aligned}$$

**proof** –

**fix**  $a b$

**assume**  $a: a \in \text{Obj}$  **and**  $b: b \in \text{Obj}$

**have**  $(\text{eval } \mathcal{I} (F_o b) \cdot r^{-1}[\text{exp } \mathcal{I} (F_o b)]) \cdot$

$$\text{Exp}^{\leftarrow} e (F_o b) \cdot F_a x b) \cdot \text{Uncurry}[\text{hom}_o a, \text{hom}_o b] (\text{hom}_a a b) =$$

$$\text{eval } (F_o x) (F_o b) \cdot (F_a x b \cdot \text{Comp } x a b \otimes e) \cdot$$

$$r^{-1}[\text{Hom } a b \otimes \text{hom}_o a]$$

**proof** –

**have**  $(\text{eval } \mathcal{I} (F_o b) \cdot r^{-1}[\text{exp } \mathcal{I} (F_o b)]) \cdot$

$$\text{Exp}^{\leftarrow} e (F_o b) \cdot F_a x b) \cdot \text{Uncurry}[\text{hom}_o a, \text{hom}_o b] (\text{hom}_a a b) =$$

$eval \mathcal{I} (F_o b) \cdot$   
 $(r^{-1}[exp \mathcal{I} (F_o b)] \cdot Exp^{\leftarrow} e (F_o b) \cdot F_a x b) \cdot$   
 $Uncurry[hom_o a, hom_o b] (hom_a a b)$   
**using comp-assoc by simp**  
**also have** ... =  $eval \mathcal{I} (F_o b) \cdot$   
 $(r^{-1}[exp \mathcal{I} (F_o b)] \cdot Exp^{\leftarrow} e (F_o b) \cdot F_a x b) \cdot$   
 $Comp x a b$   
**using a b x C.Uncurry-Curry [of - hom\_o a hom\_o b] Comp-in-hom**  
**by auto**  
**also have** ... =  $eval \mathcal{I} (F_o b) \cdot$   
 $((Exp^{\leftarrow} e (F_o b) \cdot F_a x b \otimes \mathcal{I}) \cdot$   
 $r^{-1}[hom_o b]) \cdot Comp x a b$   
**proof –**  
**have** « $Exp^{\leftarrow} e (F_o b) \cdot F_a x b : hom_o b \rightarrow exp \mathcal{I} (F_o b)$ »  
**using** *assms a b x F.preserves-Obj F.preserves-Hom [of x b]* **by force**  
**thus ?thesis**  
**using** *a b F.preserves-Obj F.preserves-Hom*  
*runit'-naturality [of Exp^{\leftarrow} e (F\_o b) \cdot F\_a x b]*  
**by auto**  
**qed**  
**also have** ... =  $eval \mathcal{I} (F_o b) \cdot$   
 $((Exp^{\leftarrow} e (F_o b) \otimes \mathcal{I}) \cdot (F_a x b \otimes \mathcal{I})) \cdot$   
 $r^{-1}[hom_o b]) \cdot$   
 $Comp x a b$   
**using** *assms a b x F.preserves-Obj F.preserves-Hom [of x b]*  
*interchange [of Exp^{\leftarrow} e (F\_o b) F\_a x b \mathcal{I} \mathcal{I}]*  
**by fastforce**  
**also have** ... =  $Uncurry[\mathcal{I}, F_o b] (Exp^{\leftarrow} e (F_o b)) \cdot (F_a x b \otimes \mathcal{I}) \cdot$   
 $r^{-1}[hom_o b] \cdot Comp x a b$   
**using comp-assoc by simp**  
**also have** ... =  $(eval (F_o x) (F_o b) \cdot (exp (F_o x) (F_o b) \otimes e)) \cdot$   
 $(F_a x b \otimes \mathcal{I}) \cdot r^{-1}[hom_o b] \cdot Comp x a b$   
**using** *assms a b x F.preserves-Obj C.Uncurry-Curry* **by auto**  
**also have** ... =  $eval (F_o x) (F_o b) \cdot$   
 $((exp (F_o x) (F_o b) \otimes e) \cdot (F_a x b \otimes \mathcal{I})) \cdot$   
 $r^{-1}[hom_o b] \cdot Comp x a b$   
**using comp-assoc by simp**  
**also have** ... =  $eval (F_o x) (F_o b) \cdot (F_a x b \otimes e) \cdot r^{-1}[hom_o b] \cdot$   
 $Comp x a b$   
**using** *assms a b x F.preserves-Hom [of x b]*  
*comp-cod-arr [of F\_a x b exp (F\_o x) (F\_o b)] comp-arr-dom*  
*interchange*  
**by fastforce**  
**also have** ... =  $eval (F_o x) (F_o b) \cdot (F_a x b \otimes e) \cdot$   
 $(Comp x a b \otimes \mathcal{I}) \cdot r^{-1}[Hom a b \otimes hom_o a]$   
**using** *assms a b x runit'-naturality [of Comp x a b]*  
*Comp-in-hom [of x a b]*  
**by auto**  
**also have** ... =  $eval (F_o x) (F_o b) \cdot ((F_a x b \otimes e) \cdot (Comp x a b \otimes \mathcal{I})) \cdot$



$r^{-1}[\text{Hom } a \ b \ \otimes \ \text{hom}_o \ a]$

**using** *comp-assoc by simp*

**also have**  $\dots = \text{eval } (F_o \ x) \ (F_o \ b) \cdot (F_a \ x \ b \cdot \text{Comp } x \ a \ b \ \otimes \ e) \cdot$   
 $r^{-1}[\text{Hom } a \ b \ \otimes \ \text{hom}_o \ a]$

**using** *assms a b x F.preserves-Hom [of x b] Comp-in-hom comp-arr-dom*  
*interchange [of F\_a x b Comp x a b e I]*

**by** *fastforce*

**finally show** *?thesis by blast*

**qed**

**also have**  $\dots = \text{eval } (F_o \ a) \ (F_o \ b) \cdot$   
 $(F_a \ a \ b \ \otimes \ \text{eval } \mathcal{I} \ (F_o \ a) \cdot r^{-1}[\text{exp } \mathcal{I} \ (F_o \ a)]) \cdot$   
 $\text{Exp}^{\leftarrow} \ e \ (F_o \ a) \cdot F_a \ x \ a)$

**proof** –

**have**  $\text{eval } (F_o \ a) \ (F_o \ b) \cdot$   
 $(F_a \ a \ b \ \otimes \ \text{eval } \mathcal{I} \ (F_o \ a) \cdot r^{-1}[\text{exp } \mathcal{I} \ (F_o \ a)]) \cdot$   
 $\text{Exp}^{\leftarrow} \ e \ (F_o \ a) \cdot F_a \ x \ a) =$   
 $\text{eval } (F_o \ a) \ (F_o \ b) \cdot$   
 $(F_a \ a \ b \ \otimes \ \text{eval } \mathcal{I} \ (F_o \ a) \cdot (r^{-1}[\text{exp } \mathcal{I} \ (F_o \ a)] \cdot$   
 $\text{Exp}^{\leftarrow} \ e \ (F_o \ a)) \cdot F_a \ x \ a)$

**using** *comp-assoc by simp*

**also have**  $\dots =$   
 $\text{eval } (F_o \ a) \ (F_o \ b) \cdot$   
 $(F_a \ a \ b \ \otimes \ \text{eval } \mathcal{I} \ (F_o \ a) \cdot$   
 $((\text{Exp}^{\leftarrow} \ e \ (F_o \ a) \ \otimes \ \mathcal{I}) \cdot r^{-1}[\text{exp } (F_o \ x) \ (F_o \ a)]) \cdot$   
 $F_a \ x \ a)$

**using** *assms a b x F.preserves-Obj F.preserves-Hom*  
*runit'-naturality [of Exp<sup>←</sup> e (F\_o a)]*

**by** *auto*

**also have**  $\dots =$   
 $\text{eval } (F_o \ a) \ (F_o \ b) \cdot$   
 $(F_a \ a \ b \ \otimes$   
 $\text{Uncurry}[\mathcal{I}, F_o \ a] \ (\text{Exp}^{\leftarrow} \ e \ (F_o \ a)) \cdot r^{-1}[\text{exp } (F_o \ x) \ (F_o \ a)]) \cdot$   
 $F_a \ x \ a)$

**using** *comp-assoc by simp*

**also have**  $\dots =$   
 $\text{eval } (F_o \ a) \ (F_o \ b) \cdot$   
 $(F_a \ a \ b \ \otimes$   
 $(\text{eval } (F_o \ x) \ (F_o \ a) \cdot (\text{exp } (F_o \ x) \ (F_o \ a) \ \otimes \ e)) \cdot$   
 $r^{-1}[\text{exp } (F_o \ x) \ (F_o \ a)] \cdot F_a \ x \ a)$

**using** *assms a b x F.preserves-Obj C.Uncurry-Curry by auto*

**also have**  $\dots =$   
 $\text{eval } (F_o \ a) \ (F_o \ b) \cdot$   
 $(F_a \ a \ b \ \otimes$   
 $(\text{eval } (F_o \ x) \ (F_o \ a) \cdot (\text{exp } (F_o \ x) \ (F_o \ a) \ \otimes \ e)) \cdot$   
 $(F_a \ x \ a \ \otimes \ \mathcal{I}) \cdot r^{-1}[\text{hom}_o \ a])$

**using** *a b x F.preserves-Hom [of x a] runit'-naturality by fastforce*

**also have**  $\dots =$   
 $\text{eval } (F_o \ a) \ (F_o \ b) \cdot$   
 $(F_a \ a \ b \ \otimes$

$$\text{eval } (F_o x) (F_o a) \cdot (\text{exp } (F_o x) (F_o a) \otimes e) \cdot$$

$$(F_a x a \otimes \mathcal{I}) \cdot r^{-1}[\text{hom}_o a]$$

**using** *comp-assoc* **by** *simp*

**also have** ... =

$$\text{eval } (F_o a) (F_o b) \cdot$$

$$(F_a a b \otimes \text{eval } (F_o x) (F_o a) \cdot (F_a x a \otimes e) \cdot r^{-1}[\text{hom}_o a])$$

**using** *assms a b x F.preserves-Obj F.preserves-Hom F.preserves-Hom*  
*comp-arr-dom* [of *e*  $\mathcal{I}$ ]  
*comp-cod-arr* [of  $F_a x a \text{ exp } (F_o x) (F_o a)$ ]  
*interchange* [of  $\text{exp } (F_o x) (F_o a) F_a x a e \mathcal{I}$ ] *comp-assoc*

**by** (*metis in-homE*)

**also have** ... =

$$\text{eval } (F_o a) (F_o b) \cdot$$

$$(\text{exp } (F_o a) (F_o b) \otimes \text{eval } (F_o x) (F_o a)) \cdot$$

$$(F_a a b \otimes (F_a x a \otimes e) \cdot r^{-1}[\text{hom}_o a])$$

**using** *assms a b x F.preserves-Obj F.preserves-Hom* [of *x a*]  
*F.preserves-Hom* [of *a b*]  
*comp-cod-arr* [of  $F_a a b \text{ exp } (F_o a) (F_o b)$ ]  
*interchange*  
[of  $\text{exp } (F_o a) (F_o b) F_a a b$   
 $\text{eval } (F_o x) (F_o a) (F_a x a \otimes e) \cdot r^{-1}[\text{hom}_o a]$ ]

**by** *fastforce*

**also have** ... =  $(\text{eval } (F_o a) (F_o b) \cdot$   
 $(\text{exp } (F_o a) (F_o b) \otimes \text{eval } (F_o x) (F_o a))) \cdot$   
 $(F_a a b \otimes (F_a x a \otimes e) \cdot r^{-1}[\text{hom}_o a])$

**using** *comp-assoc* **by** *simp*

**also have** ... =  $(\text{eval } (F_o a) (F_o b) \cdot$   
 $(\text{exp } (F_o a) (F_o b) \otimes \text{eval } (F_o x) (F_o a))) \cdot$   
 $(F_a a b \otimes (F_a x a \otimes e)) \cdot (\text{Hom } a b \otimes r^{-1}[\text{hom}_o a])$

**using** *assms a b x F.preserves-Obj F.preserves-Hom* [of *a b*]  
*F.preserves-Hom* [of *x a*] *comp-arr-dom* [of  $F_a a b \text{ Hom } a b$ ]  
*interchange* [of  $F_a a b \text{ Hom } a b F_a x a \otimes e r^{-1}[\text{hom}_o a]$ ]

**by** *fastforce*

**also have** ... =  $(\text{eval } (F_o a) (F_o b) \cdot$   
 $(\text{exp } (F_o a) (F_o b) \otimes \text{eval } (F_o x) (F_o a))) \cdot$   
 $(\text{exp } (F_o a) (F_o b) \otimes \text{exp } (F_o x) (F_o a) \otimes F_o x) \cdot$   
 $(F_a a b \otimes F_a x a \otimes e) \cdot (\text{Hom } a b \otimes r^{-1}[\text{hom}_o a])$

**using** *assms a b x F.preserves-Obj F.preserves-Hom* [of *a b*]  
*F.preserves-Hom* [of *x a*]  
*comp-cod-arr* [of  $(F_a a b \otimes F_a x a \otimes e) \cdot (\text{Hom } a b \otimes r^{-1}[\text{hom}_o a])$ ]  
 $\text{exp } (F_o a) (F_o b) \otimes \text{exp } (F_o x) (F_o a) \otimes F_o x]$

**by** *fastforce*

**also have** ... =  $(\text{eval } (F_o a) (F_o b) \cdot$   
 $(\text{exp } (F_o a) (F_o b) \otimes \text{eval } (F_o x) (F_o a))) \cdot$   
 $(\text{a}[\text{exp } (F_o a) (F_o b), \text{exp } (F_o x) (F_o a), F_o x] \cdot$   
 $\text{a}^{-1}[\text{exp } (F_o a) (F_o b), \text{exp } (F_o x) (F_o a), F_o x]) \cdot$   
 $(F_a a b \otimes (F_a x a \otimes e)) \cdot (\text{Hom } a b \otimes r^{-1}[\text{hom}_o a])$

**using** *assms a b x F.preserves-Obj comp-assoc-assoc'* **by** *simp*

**also have** ... =  $(\text{eval } (F_o a) (F_o b) \cdot$

$(\text{exp } (F_o a) (F_o b) \otimes \text{eval } (F_o x) (F_o a)) \cdot$   
 $\text{a}[\text{exp } (F_o a) (F_o b), \text{exp } (F_o x) (F_o a), F_o x] \cdot$   
 $(\text{a}^{-1}[\text{exp } (F_o a) (F_o b), \text{exp } (F_o x) (F_o a), F_o x] \cdot$   
 $(F_a a b \otimes F_a x a \otimes e)) \cdot (\text{Hom } a b \otimes \text{r}^{-1}[\text{hom}_o a])$   
**using comp-assoc by simp**  
**also have** ... =  $\text{Uncurry}[F_o x, F_o b] (C.\text{Comp } (F_o x) (F_o a) (F_o b)) \cdot$   
 $(\text{a}^{-1}[\text{exp } (F_o a) (F_o b), \text{exp } (F_o x) (F_o a), F_o x] \cdot$   
 $(F_a a b \otimes F_a x a \otimes e)) \cdot (\text{Hom } a b \otimes \text{r}^{-1}[\text{hom}_o a])$   
**using assms a b x F.preserves-Obj C.Uncurry-Curry C.Comp-def**  
**by auto**  
**also have** ... =  $\text{Uncurry}[F_o x, F_o b] (C.\text{Comp } (F_o x) (F_o a) (F_o b)) \cdot$   
 $((F_a a b \otimes F_a x a \otimes e) \cdot \text{a}^{-1}[\text{Hom } a b, \text{hom}_o a, \mathcal{I}]) \cdot$   
 $(\text{Hom } a b \otimes \text{r}^{-1}[\text{hom}_o a])$   
**using assms a b x F.preserves-Hom [of a b] F.preserves-Hom [of x a]**  
**assoc'-naturality [of F\_a a b F\_a x a e]**  
**by fastforce**  
**also have** ... =  $\text{eval } (F_o x) (F_o b) \cdot$   
 $((C.\text{Comp } (F_o x) (F_o a) (F_o b) \otimes F_o x) \cdot$   
 $((F_a a b \otimes F_a x a \otimes e)) \cdot$   
 $\text{a}^{-1}[\text{Hom } a b, \text{hom}_o a, \mathcal{I}] \cdot (\text{Hom } a b \otimes \text{r}^{-1}[\text{hom}_o a])$   
**using comp-assoc by simp**  
**also have** ... =  $\text{eval } (F_o x) (F_o b) \cdot$   
 $(C.\text{Comp } (F_o x) (F_o a) (F_o b) \cdot (F_a a b \otimes F_a x a \otimes e) \cdot$   
 $\text{a}^{-1}[\text{Hom } a b, \text{hom}_o a, \mathcal{I}] \cdot (\text{Hom } a b \otimes \text{r}^{-1}[\text{hom}_o a])$   
**using assms a b x F.preserves-Obj F.preserves-Hom [of a b]**  
**F.preserves-Hom [of x a] comp-cod-arr [of e F\_o x]**  
**interchange**  
 $[of C.\text{Comp } (F_o x) (F_o a) (F_o b) F_a a b \otimes F_a x a F_o x e]$   
**by fastforce**  
**also have** ... =  $\text{eval } (F_o x) (F_o b) \cdot (F_a x b \cdot \text{Comp } x a b \otimes e) \cdot$   
 $\text{a}^{-1}[\text{Hom } a b, \text{hom}_o a, \mathcal{I}] \cdot (\text{Hom } a b \otimes \text{r}^{-1}[\text{hom}_o a])$   
**using assms a b x F.preserves-Obj F.preserves-Hom F.preserves-Comp**  
**by simp**  
**also have** ... =  $\text{eval } (F_o x) (F_o b) \cdot (F_a x b \cdot \text{Comp } x a b \otimes e) \cdot$   
 $\text{r}^{-1}[\text{Hom } a b \otimes \text{hom}_o a]$   
**proof –**  
**have**  $\text{a}^{-1}[\text{Hom } a b, \text{hom}_o a, \mathcal{I}] \cdot (\text{Hom } a b \otimes \text{r}^{-1}[\text{hom}_o a]) =$   
 $\text{r}^{-1}[\text{Hom } a b \otimes \text{hom}_o a]$   
**proof –**  
**have**  $\text{a}^{-1}[\text{Hom } a b, \text{hom}_o a, \mathcal{I}] \cdot (\text{Hom } a b \otimes \text{r}^{-1}[\text{hom}_o a]) =$   
 $\text{inv } ((\text{Hom } a b \otimes \text{r}[\text{hom}_o a]) \cdot \text{a}[\text{Hom } a b, \text{hom}_o a, \mathcal{I}])$   
**using assms a b x inv-comp by auto**  
**also have** ... =  $\text{r}^{-1}[\text{Hom } a b \otimes \text{hom}_o a]$   
**using assms a b x runit-tensor by auto**  
**finally show ?thesis by blast**  
**qed**  
**thus ?thesis by simp**  
**qed**  
**finally show ?thesis by simp**

**qed**  
**finally show**  $(eval \mathcal{I} (F_o b) \cdot r^{-1}[exp \mathcal{I} (F_o b)] \cdot Exp^{\leftarrow} e (F_o b) \cdot F_a x b) \cdot$   
 $Uncurry[hom_o a, hom_o b] (hom_a a b) =$   
 $eval (F_o a) (F_o b) \cdot (F_a a b \otimes eval \mathcal{I} (F_o a) \cdot r^{-1}[exp \mathcal{I} (F_o a)] \cdot$   
 $Exp^{\leftarrow} e (F_o a) \cdot F_a x a)$   
**by argo**  
**qed**  
**qed**

If  $\tau: hom\ x \rightarrow F$  is an enriched natural transformation, then there exists an element  $e_\tau: \mathcal{I} \rightarrow F\ x$  that generates  $\tau$  via the preceding formula. The idea (Kelly 1.46) is to take:

$$e_\tau = \mathcal{I} \xrightarrow{Id\ x} hom_o\ x \xrightarrow{\tau\ x} F\ x$$

This amounts to the “evaluation of  $\tau\ x$  at the identity on  $x$ ”.

However, note once again that, according to the formal definition of enriched natural transformation, we have  $\tau\ x: \mathcal{I} \rightarrow exp(hom_o\ x)(F_o\ x)$ , so it is necessary to transform this to an arrow:  $(\tau\ x) \downarrow [hom_o\ x, F_o\ x]: hom_o\ x \rightarrow F\ x$ .

**abbreviation** *generating-elem*  
**where** *generating-elem*  $\tau \equiv (\tau\ x) \downarrow [hom_o\ x, F_o\ x] \cdot Id\ x$

**lemma** *generating-elem-in-hom*:

**assumes** *enriched-natural-transformation*  $C\ T\ \alpha\ \iota$   
 $Obj\ Hom\ Id\ Comp\ (Collect\ ide)\ exp\ C.Id\ C.Comp$   
 $hom_o\ hom_a\ F_o\ F_a\ \tau$

**shows** «*generating-elem*  $\tau: \mathcal{I} \rightarrow F_o\ x$ »

**proof** –

**interpret**  $\tau$ : *enriched-natural-transformation*  $C\ T\ \alpha\ \iota$   
 $Obj\ Hom\ Id\ Comp\ \langle Collect\ ide \rangle\ exp\ C.Id\ C.Comp$   
 $hom_o\ hom_a\ F_o\ F_a\ \tau$

**using** *assms* **by** *blast*

**show** «*generating-elem*  $\tau: \mathcal{I} \rightarrow F_o\ x$ »

**using**  $x\ Id-in-hom\ \tau.component-in-hom\ [of\ x]\ F.preserves-Obj\ C.DN-def$

**by** *auto fastforce*

**qed**

Now we have to verify the elements of the diagram after Kelly (1.47):

$$\begin{array}{ccccccc}
& & & \text{hom}_o a & & & \\
& & & \curvearrowright & & & \\
\text{hom}_o a & \xrightarrow{\text{hom}_a x a} & [\text{hom}_o x, \text{hom}_o a] & \xrightarrow{[Id x, \text{hom}_o a]} & [\mathcal{I}, \text{hom}_o a] & \xrightarrow{\text{iso}} & \text{hom}_o a \\
\downarrow F_a a & & \downarrow [\text{hom}_o x, \tau a] & & \downarrow [\mathcal{I}, \tau a] & & \downarrow \tau a \\
[F_o x, F_o a] & \xrightarrow{[\tau_e x, F_o a]} & [\text{hom}_o x, F_o a] & \xrightarrow{[Id x, F_o a]} & [\mathcal{I}, F_o a] & \xrightarrow{\text{iso}} & F_o a \\
& & & \curvearrowleft & & & \\
& & & [\tau_e x \cdot Id x, F_o a] & & & 
\end{array}$$

The left square is enriched naturality of  $\tau$  (Kelly (1.39)). The middle square commutes trivially. The right square commutes by the naturality of the canonical isomorphism from  $[\mathcal{I}, \text{hom}_o a]$  to  $\text{hom}_o a$ . The top edge composes to  $\text{hom}_o a$  (an identity). The commutativity of the entire diagram shows that  $\tau a$  is recovered from  $e_\tau$ . Note that where  $\tau a$  appears, what is actually meant formally is  $(\tau a) \downarrow [\text{hom}_o a, F_o a]$ .

**lemma center-square:**

**assumes** *enriched-natural-transformation*  $C T \alpha \iota$   
*Obj Hom Id Comp (Collect ide) exp C.Id C.Comp*  
 $\text{hom}_o \text{hom}_a F_o F_a \tau$

**and**  $a \in \text{Obj}$

**shows**  $C.Exp \mathcal{I} (\tau a \downarrow [\text{hom}_o a, F_o a]) \cdot C.Exp (Id x) (\text{hom}_o a) =$   
 $C.Exp (Id x) (F_o a) \cdot C.Exp (\text{hom}_o x) (\tau a \downarrow [\text{hom}_o a, F_o a])$

**proof** –

**interpret**  $\tau$ : *enriched-natural-transformation*  $C T \alpha \iota$   
*Obj Hom Id Comp (Collect ide) exp C.Id C.Comp*  
 $\text{hom}_o \text{hom}_a F_o F_a \tau$

**using** *assms by blast*

**let**  $? \tau_a = \tau a \downarrow [\text{hom}_o a, F_o a]$

**show**  $C.Exp \mathcal{I} ? \tau_a \cdot C.Exp (Id x) (\text{hom}_o a) =$   
 $C.Exp (Id x) (F_o a) \cdot C.Exp (\text{hom}_o x) ? \tau_a$

**by** (*metis assms(2) x C.Exp-comp F.preserves-Obj Id-in-hom*  
*C.DN-simps(1–3) comp-arr-dom comp-cod-arr in-homE \tau.component-in-hom*  
*ide-Hom mem-Collect-eq*)

**qed**

**lemma right-square:**

**assumes** *enriched-natural-transformation*  $C T \alpha \iota$   
*Obj Hom Id Comp (Collect ide) exp C.Id C.Comp*

$hom_o hom_a F_o F_a \tau$

**and**  $a \in Obj$

**shows**  $\tau a \downarrow [hom_o a, F_o a] \cdot C.Dn (hom_o a) =$   
 $C.Dn (F_o a) \cdot C.Exp \mathcal{I} (\tau a \downarrow [hom_o a, F_o a])$

**proof** –

**interpret**  $\tau$ : *enriched-natural-transformation*  $C T \alpha \iota$   
 $Obj Hom Id Comp \langle Collect\ ide \rangle exp C.Id C.Comp$   
 $hom_o hom_a F_o F_a \tau$

**using** *assms* **by** *blast*

**show** *?thesis*

**using** *assms(2)* *C.Up-Dn-naturality* *C.DN-simps*  $\tau.component-in-hom$

**apply** *auto[1]*

**by** (*metis* *C.Exp-ide-y* *C.UP-DN(2)* *F.preserves-Obj ide-Hom ide-unity*  
*in-homE mem-Collect-eq x*)

**qed**

**lemma** *top-path*:

**assumes**  $a \in Obj$

**shows**  $eval \mathcal{I} (hom_o a) \cdot r^{-1}[exp \mathcal{I} (hom_o a)] \cdot C.Exp (Id x) (hom_o a) \cdot$   
 $hom_a x a =$   
 $hom_o a$

**proof** –

**have**  $eval \mathcal{I} (hom_o a) \cdot r^{-1}[exp \mathcal{I} (hom_o a)] \cdot C.Exp (Id x) (hom_o a) \cdot$   
 $hom_a x a =$   
 $eval \mathcal{I} (hom_o a) \cdot r^{-1}[exp \mathcal{I} (hom_o a)] \cdot$   
 $(Curry[exp \mathcal{I} (hom_o a), \mathcal{I}, hom_o a] (hom_o a \cdot eval \mathcal{I} (hom_o a)) \cdot$   
 $Curry[exp (hom_o x) (hom_o a), \mathcal{I}, hom_o a]$   
 $(eval (hom_o x) (hom_o a) \cdot (exp (hom_o x) (hom_o a) \otimes Id x))) \cdot$   
 $hom_a x a$

**using** *assms*  $x$  *C.Exp-def Id-in-hom* [of  $x$ ] **by** *auto*

**also have** ... =

$eval \mathcal{I} (hom_o a) \cdot r^{-1}[exp \mathcal{I} (hom_o a)] \cdot$   
 $Curry[exp \mathcal{I} (hom_o a), \mathcal{I}, hom_o a] (hom_o a \cdot eval \mathcal{I} (hom_o a)) \cdot$   
 $Curry[exp (hom_o x) (hom_o a), \mathcal{I}, hom_o a]$   
 $(eval (hom_o x) (hom_o a) \cdot (exp (hom_o x) (hom_o a) \otimes Id x)) \cdot$   
 $hom_a x a$

**using** *comp-assoc* **by** *simp*

**also have** ... =

$eval \mathcal{I} (hom_o a) \cdot r^{-1}[exp \mathcal{I} (hom_o a)] \cdot$   
 $Curry[exp \mathcal{I} (hom_o a), \mathcal{I}, hom_o a] (hom_o a \cdot eval \mathcal{I} (hom_o a)) \cdot$   
 $Curry[hom_o a, \mathcal{I}, hom_o a]$   
 $((eval (hom_o x) (hom_o a) \cdot (exp (hom_o x) (hom_o a) \otimes Id x)) \cdot$   
 $(hom_a x a \otimes \mathcal{I}))$

**proof** –

**have**  $\ll eval (hom_o x) (hom_o a) \cdot (exp (hom_o x) (hom_o a) \otimes Id x)$   
 $: exp (hom_o x) (hom_o a) \otimes \mathcal{I} \rightarrow hom_o a \gg$

**using** *assms*  $x$

**by** (*meson* *Id-in-hom comp-in-homI* *C.eval-in-hom-ax* *C.ide-exp*  
*ide-in-hom tensor-in-hom ide-Hom*)

**thus** *?thesis*  
**using** *assms x preserves-Hom [of x a] C.comp-Curry-arr by simp*  
**qed**  
**also have** ... =  

$$\begin{aligned} & \text{eval } \mathcal{I} (\text{hom}_o a) \cdot \mathbf{r}^{-1}[\text{exp } \mathcal{I} (\text{hom}_o a)] \cdot \\ & \text{Curry}[\text{exp } \mathcal{I} (\text{hom}_o a), \mathcal{I}, \text{hom}_o a] (\text{hom}_o a \cdot \text{eval } \mathcal{I} (\text{hom}_o a)) \cdot \\ & \text{Curry}[\text{hom}_o a, \mathcal{I}, \text{hom}_o a] \\ & (\text{eval } (\text{hom}_o x) (\text{hom}_o a) \cdot \\ & (\text{exp } (\text{hom}_o x) (\text{hom}_o a) \otimes \text{Id } x) \cdot (\text{hom}_a x a \otimes \mathcal{I})) \end{aligned}$$
**using** *comp-assoc by simp*  
**also have** ... =  

$$\begin{aligned} & \text{eval } \mathcal{I} (\text{hom}_o a) \cdot \mathbf{r}^{-1}[\text{exp } \mathcal{I} (\text{hom}_o a)] \cdot \\ & \text{Curry}[\text{exp } \mathcal{I} (\text{hom}_o a), \mathcal{I}, \text{hom}_o a] (\text{hom}_o a \cdot \text{eval } \mathcal{I} (\text{hom}_o a)) \cdot \\ & \text{Curry}[\text{hom}_o a, \mathcal{I}, \text{hom}_o a] \\ & (\text{eval } (\text{hom}_o x) (\text{hom}_o a) \cdot (\text{hom}_a x a \otimes \text{Id } x)) \end{aligned}$$
**proof** –  
**have** *seq (Id x) \mathcal{I} \wedge seq (hom\_o x) (Id x)*  
**using** *x Id-in-hom ide-in-hom ide-unity by blast*  
**thus** *?thesis*  
**using** *assms x preserves-Hom comp-arr-dom [of Id x \mathcal{I}]*  
*interchange [of exp (hom\_o x) (hom\_o a) hom\_a x a Id x \mathcal{I}]*  
**by** *(metis comp-cod-arr comp-ide-arr dom-eqI ide-unity*  
*in-homE ide-Hom)*  
**qed**  
**also have** ... =  

$$\begin{aligned} & \text{eval } \mathcal{I} (\text{hom}_o a) \cdot \mathbf{r}^{-1}[\text{exp } \mathcal{I} (\text{hom}_o a)] \cdot \\ & \text{Curry}[\text{exp } \mathcal{I} (\text{hom}_o a), \mathcal{I}, \text{hom}_o a] (\text{eval } \mathcal{I} (\text{hom}_o a)) \cdot \\ & \text{Curry}[\text{hom}_o a, \mathcal{I}, \text{hom}_o a] \\ & (\text{Uncurry}[\text{hom}_o x, \text{hom}_o a] (\text{hom}_a x a) \cdot (\text{hom}_o a \otimes \text{Id } x)) \end{aligned}$$
**proof** –  
**have** *eval (hom\_o x) (hom\_o a) \cdot (hom\_a x a \otimes Id x) =*  

$$\text{eval } (\text{hom}_o x) (\text{hom}_o a) \cdot (\text{hom}_a x a \cdot \text{hom}_o a \otimes \text{hom}_o x \cdot \text{Id } x)$$
**using** *assms x Id-in-hom comp-cod-arr comp-arr-dom Comp-in-hom*  
**by** *(metis in-homE preserves-Hom)*  
**also have** ... = *eval (hom\_o x) (hom\_o a) \cdot (hom\_a x a \otimes hom\_o x) \cdot*  

$$(\text{hom}_o a \otimes \text{Id } x)$$
**using** *assms x Id-in-hom Comp-in-hom*  
*interchange [of hom\_a x a hom\_o a hom\_o x Id x]*  
**by** *(metis comp-arr-dom comp-cod-arr in-homE preserves-Hom)*  
**also have** ... = *Uncurry[hom\_o x, hom\_o a] (hom\_a x a) \cdot (hom\_o a \otimes Id x)*  
**using** *comp-assoc by simp*  
**finally show** *?thesis*  
**using** *assms x comp-cod-arr ide-Hom ide-unity C.eval-simps(1,3) by metis*  
**qed**  
**also have** ... =  

$$\begin{aligned} & \text{eval } \mathcal{I} (\text{hom}_o a) \cdot \mathbf{r}^{-1}[\text{exp } \mathcal{I} (\text{hom}_o a)] \cdot \\ & \text{Curry}[\text{hom}_o a, \mathcal{I}, \text{hom}_o a] \\ & (\text{Uncurry}[\mathcal{I}, \text{hom}_o a] \\ & (\text{Curry}[\text{hom}_o a, \mathcal{I}, \text{hom}_o a] \end{aligned}$$

$(\text{Uncurry}[\text{hom}_o x, \text{hom}_o a] (\text{hom}_a x a) \cdot (\text{hom}_o a \otimes \text{Id } x))$

**using** *assms x*  
*C.comp-Curry-arr*  
[*of I*  
*Curry*[\text{hom}\_o a, *I*, \text{hom}\_o a]  
 $(\text{Uncurry}[\text{hom}_o x, \text{hom}_o a] (\text{hom}_a x a) \cdot (\text{hom}_o a \otimes \text{Id } x))$   
*hom}\_o a \text{ exp } I (\text{hom}\_o a)*  
*eval I (hom}\_o a) \text{ hom}\_o a]*

**apply** *auto[1]*  
**by** (*metis Comp-Hom-Id Comp-in-hom C.Uncurry-Curry C.eval-in-hom-ax*  
*ide-unity C.isomorphic-exp-unity(1) ide-Hom*)

**also have** ... =  
*eval I (hom}\_o a) \cdot r^{-1}[\text{exp } I (\text{hom}\_o a)] \cdot*  
*Curry*[\text{hom}\_o a, *I*, \text{hom}\_o a]  
 $(\text{Uncurry}[\text{hom}_o x, \text{hom}_o a] (\text{hom}_a x a) \cdot (\text{hom}_o a \otimes \text{Id } x))$

**using** *assms x C.Uncurry-Curry*  
**by** (*simp add: Comp-Hom-Id Comp-in-hom C.Curry-Uncurry*  
*C.isomorphic-exp-unity(1)*)

**also have** ... =  
*eval I (hom}\_o a) \cdot r^{-1}[\text{exp } I (\text{hom}\_o a)] \cdot*  
*Curry*[\text{hom}\_o a, *I*, \text{hom}\_o a]  
 $(\text{eval } (\text{hom}_o x) (\text{hom}_o a) \cdot (\text{hom}_a x a \otimes \text{hom}_o x) \cdot (\text{hom}_o a \otimes \text{Id } x))$

**using** *comp-assoc by simp*

**also have** ... =  
*eval I (hom}\_o a) \cdot r^{-1}[\text{exp } I (\text{hom}\_o a)] \cdot*  
*Curry*[\text{hom}\_o a, *I*, \text{hom}\_o a]  
 $(\text{eval } (\text{hom}_o x) (\text{hom}_o a) \cdot (\text{hom}_a x a \otimes \text{Id } x))$

**using** *assms x comp-cod-arr [of Id x hom}\_o x] comp-arr-dom*  
*interchange [of hom}\_a x a hom}\_o a hom}\_o x Id x]*  
*preserves-Hom [of x a] Id-in-hom*

**apply** *auto[1]*  
**by** *fastforce*

**also have** ... =  
*eval I (hom}\_o a) \cdot*  
*(Curry*[\text{hom}\_o a, *I*, \text{hom}\_o a]  
 $(\text{eval } (\text{hom}_o x) (\text{hom}_o a) \cdot (\text{hom}_a x a \otimes \text{Id } x)) \otimes I) \cdot$   
 $r^{-1}[\text{hom}_o a]$

**proof** –  
**have** «*Curry*[\text{hom}\_o a, *I*, \text{hom}\_o a]  
 $(\text{eval } (\text{hom}_o x) (\text{hom}_o a) \cdot (\text{hom}_a x a \otimes \text{Id } x))$   
 $: \text{hom}_o a \rightarrow \text{exp } I (\text{hom}_o a)$ »  
**using** *assms x preserves-Hom [of x a] Id-in-hom [of x] by force*  
**thus** *?thesis*  
**using** *assms x runit'-naturality by fastforce*

**qed**

**also have** ... =  
*Uncurry*[*I*, \text{hom}\_o a]  
*(Curry*[\text{hom}\_o a, *I*, \text{hom}\_o a]  
 $(\text{eval } (\text{hom}_o x) (\text{hom}_o a) \cdot (\text{hom}_a x a \otimes \text{Id } x)) \cdot r^{-1}[\text{hom}_o a]$



**using** *comp-assoc by simp*  
**also have** ... = (eval (hom<sub>o</sub> x) (hom<sub>o</sub> a) ·  
(Curry[hom<sub>o</sub> a, hom<sub>o</sub> x, hom<sub>o</sub> a] (Comp x x a) ⊗ Id x)) ·  
r<sup>-1</sup>[hom<sub>o</sub> a]  
**using** *assms x C.Uncurry-Curry preserves-Hom [of x a] Id-in-hom [of x]*  
**by** *fastforce*  
**also have** ... = (eval (hom<sub>o</sub> x) (hom<sub>o</sub> a) ·  
((Curry[hom<sub>o</sub> a, hom<sub>o</sub> x, hom<sub>o</sub> a] (Comp x x a) ⊗ hom<sub>o</sub> x) ·  
(hom<sub>o</sub> a ⊗ Id x))) · r<sup>-1</sup>[hom<sub>o</sub> a]  
**using** *assms x Id-in-hom [of x] Comp-in-hom comp-arr-dom comp-cod-arr*  
*interchange*  
**by** *auto*  
**also have** ... = Uncurry[hom<sub>o</sub> x, hom<sub>o</sub> a]  
(Curry[hom<sub>o</sub> a, hom<sub>o</sub> x, hom<sub>o</sub> a] (Comp x x a)) ·  
(hom<sub>o</sub> a ⊗ Id x) · r<sup>-1</sup>[hom<sub>o</sub> a]  
**using** *comp-assoc by simp*  
**also have** ... = Comp x x a · (hom<sub>o</sub> a ⊗ Id x) · r<sup>-1</sup>[hom<sub>o</sub> a]  
**using** *assms x C.Uncurry-Curry Comp-in-hom by simp*  
**also have** ... = (Comp x x a · (hom<sub>o</sub> a ⊗ Id x)) · r<sup>-1</sup>[hom<sub>o</sub> a]  
**using** *comp-assoc by simp*  
**also have** ... = r[hom<sub>o</sub> a] · r<sup>-1</sup>[hom<sub>o</sub> a]  
**using** *assms x Comp-Hom-Id by auto*  
**also have** ... = hom<sub>o</sub> a  
**using** *assms x comp-runit-runit' by blast*  
**finally show** *?thesis by blast*  
**qed**

The left square is an instance of Kelly (1.39), so we can get that by instantiating that result. The confusing business is that the target enriched category is the base category C.

**lemma** *left-square:*

**assumes** *enriched-natural-transformation C T α ι*  
*Obj Hom Id Comp (Collect ide) exp C.Id C.Comp*  
*hom<sub>o</sub> hom<sub>a</sub> F<sub>o</sub> F<sub>a</sub> τ*  
**and** *a ∈ Obj*  
**shows** *Exp<sup>→</sup> (hom<sub>o</sub> x) ((τ a) ↓[hom<sub>o</sub> a, F<sub>o</sub> a]) · hom<sub>a</sub> x a =*  
*Exp<sup>←</sup> ((τ x) ↓[hom<sub>o</sub> x, F<sub>o</sub> x]) (F<sub>o</sub> a) · F<sub>a</sub> x a*  
**proof** –  
**interpret** *τ: enriched-natural-transformation C T α ι*  
*Obj Hom Id Comp ⟨Collect ide⟩ exp C.Id C.Comp*  
*hom<sub>o</sub> hom<sub>a</sub> F<sub>o</sub> F<sub>a</sub> τ*  
**using** *assms(1) by blast*  
  
**interpret** *cov-Hom: covariant-Hom C T α ι exp eval Curry*  
*⟨Collect ide⟩ exp C.Id C.Comp ⟨hom<sub>o</sub> x⟩*  
**using** *x by unfold-locales auto*  
**interpret** *cnt-Hom: contravariant-Hom C T α ι σ exp eval Curry*  
*⟨Collect ide⟩ exp C.Id C.Comp ⟨F<sub>o</sub> a⟩*  
**using** *assms(2) F.preserves-Obj by unfold-locales*

**interpret** *Kelly*: *Kelly-1-39 C T  $\alpha$   $\iota$   $\sigma$  exp eval Curry*  
*Obj Hom Id Comp <Collect ide> exp C.Id C.Comp*  
*hom<sub>o</sub> hom<sub>a</sub> F<sub>o</sub> F<sub>a</sub>  $\tau$  x a*  
**using** *assms(2) x*  
**by** *unfold-locales*

The following is the enriched naturality of  $\tau$ , expressed in the alternate form involving the underlying ordinary functors of the enriched hom functors.

**have** *1: cov-Hom.map<sub>0</sub> (hom<sub>o</sub> a) (F<sub>o</sub> a) ( $\tau$  a) · hom<sub>a</sub> x a =*  
*cnt-Hom.map<sub>0</sub> (hom<sub>o</sub> x) (F<sub>o</sub> x) ( $\tau$  x) · F<sub>a</sub> x a*  
**using** *Kelly.Kelly-1-39 by simp*

Here we have the underlying ordinary functor of the enriched covariant hom, expressed in terms of the covariant endofunctor  $Exp^{\rightarrow} (hom_o x)$  on the base category.

**have** *2: cov-Hom.map<sub>0</sub> (hom<sub>o</sub> a) (F<sub>o</sub> a) ( $\tau$  a) =*  
*Exp<sup>→</sup> (hom<sub>o</sub> x) (( $\tau$  a)  $\downarrow$ [hom<sub>o</sub> a, F<sub>o</sub> a])*

**proof** –

**have** *cov-Hom.map<sub>0</sub> (hom<sub>o</sub> a) (F<sub>o</sub> a) ( $\tau$  a) =*  
*(Curry[cnt-Hom.hom<sub>o</sub> (hom<sub>o</sub> a), cov-Hom.hom<sub>o</sub> (hom<sub>o</sub> a),*  
*cnt-Hom.hom<sub>o</sub> (hom<sub>o</sub> x)]*  
*(C.Comp (hom<sub>o</sub> x) (hom<sub>o</sub> a) (F<sub>o</sub> a)) ·  $\tau$  a)*  
 *$\downarrow$ [cov-Hom.hom<sub>o</sub> (hom<sub>o</sub> a), cnt-Hom.hom<sub>o</sub> (hom<sub>o</sub> x)]*

**proof** –

**have** *cov-Hom.map<sub>0</sub> (hom<sub>o</sub> a) (F<sub>o</sub> a) ( $\tau$  a) =*  
*cnt-Hom.Op<sub>0</sub>.Map*  
*(cov-Hom.UF.map<sub>0</sub> (cnt-Hom.Op<sub>0</sub>.MkArr (hom<sub>o</sub> a) (F<sub>o</sub> a) ( $\tau$  a)))*  
 *$\downarrow$ [cnt-Hom.Op<sub>0</sub>.Dom*  
*(cov-Hom.UF.map<sub>0</sub>*  
*(cnt-Hom.Op<sub>0</sub>.MkArr (hom<sub>o</sub> a) (F<sub>o</sub> a) ( $\tau$  a))),*  
*cnt-Hom.Op<sub>0</sub>.Cod*  
*(cov-Hom.UF.map<sub>0</sub>*  
*(cnt-Hom.Op<sub>0</sub>.MkArr (hom<sub>o</sub> a) (F<sub>o</sub> a) ( $\tau$  a)))]*

**using** *assms x preserves-Obj F.preserves-Obj  $\tau$ .component-in-hom*  
*cov-Hom.Kelly-1-31 cov-Hom.UF.preserves-arr*

**by** *force*

**moreover**

**have** *cnt-Hom.Op<sub>0</sub>.Dom*  
*(cov-Hom.UF.map<sub>0</sub>*  
*(cnt-Hom.Op<sub>0</sub>.MkArr (hom<sub>o</sub> a) (F<sub>o</sub> a) ( $\tau$  a))) =*  
*exp (hom<sub>o</sub> x) (hom<sub>o</sub> a)*

**using** *assms x cov-Hom.UF.map<sub>0</sub>-def*

**apply** *auto[1]*

**using** *cnt-Hom.y  $\tau$ .component-in-hom by force*

**moreover**

**have** *cnt-Hom.Op<sub>0</sub>.Cod*  
*(cov-Hom.UF.map<sub>0</sub>*  
*(cnt-Hom.Op<sub>0</sub>.MkArr (hom<sub>o</sub> a) (F<sub>o</sub> a) ( $\tau$  a))) =*

```

      exp (homo x) (Fo a)
    using assms x cov-Hom.UF.map0-def
    apply auto[1]
    using cnt-Hom.y τ.component-in-hom by fastforce
  moreover
  have cnt-Hom.Op0.Map
    (cov-Hom.UF.map0
     (cnt-Hom.Op0.MkArr (homo a) (Fo a) (τ a))) =
    cov-Hom.homa (homo a) (Fo a) · τ a
  using assms x cov-Hom.UF.map0-def
  apply auto[1]
  using cnt-Hom.y τ.component-in-hom by auto
  ultimately show ?thesis
  using assms x ide-Hom F.preserves-Obj by simp
qed
also have ... = Exp→ (homo x) ((τ a) ↓[homo a, Fo a])
  using assms(2) x C.cov-Exp-DN τ.component-in-hom F.preserves-Obj
  by simp
  finally show ?thesis by blast
qed

```

Here we have the underlying ordinary functor of the enriched contravariant hom, expressed in terms of the contravariant endofunctor  $\lambda f. \text{Exp}^{\leftarrow} f$  ( $F_o a$ ) on the base category.

```

  have 3: cnt-Hom.map0 (homo x) (Fo x) (τ x) =
    Exp← (τ x ↓[homo x, Fo x]) (Fo a)
  proof -
  have cnt-Hom.map0 (homo x) (Fo x) (τ x) =
    Uncurry[exp (Fo x) (Fo a), exp (homo x) (Fo a)]
    (cnt-Hom.homa (Fo x) (homo x) · τ x) ·
    1-1[exp (Fo x) (Fo a)]
  proof -
  have cnt-Hom.map0 (homo x) (Fo x) (τ x) =
    Uncurry[cnt-Hom.Op0.Dom
      (cnt-Hom.UF.map0
        (cnt-Hom.Op0.MkArr (Fo x) (homo x) (τ x))),
      cnt-Hom.Op0.Cod
        (cnt-Hom.UF.map0
          (cnt-Hom.Op0.MkArr (Fo x) (homo x) (τ x)))]
    (cnt-Hom.Op0.Map
      (cnt-Hom.UF.map0
        (cnt-Hom.Op0.MkArr (Fo x) (Hom x x) (τ x)))) ·
    1-1[cnt-Hom.Op0.Dom
      (cnt-Hom.UF.map0
        (cnt-Hom.Op0.MkArr (Fo x) (homo x) (τ x)))]
  using assms x 1 2 cnt-Hom.Kelly-1-32 [of homo x Fo x τ x]
    C.Curry-simps(1-3) C.DN-def C.UP-DN(2) C.eval-simps(1-3)
    C.ide-exp Comp-in-hom F.preserves-Obj comp-in-homI'
    not-arr-null preserves-Obj τ.component-in-hom in-homE
  
```

$mem\text{-}Collect\text{-}eq\ seqE$   
**by** (*smt* (*verit*))  
**moreover have**  $cnt\text{-}Hom.Op_0.Dom$   
 $(cnt\text{-}Hom.UF.map_0$   
 $(cnt\text{-}Hom.Op_0.MkArr (F_o x) (hom_o x) (\tau x))) =$   
 $exp (F_o x) (F_o a)$   
**using**  $assms\ x\ cnt\text{-}Hom.UF.map_0\text{-}def$   
**apply**  $auto[1]$   
**using**  $F.preserves\text{-}Obj\ cnt\text{-}Hom.Op_0.arr\text{-}MkArr\ \tau.component\text{-}in\text{-}hom$   
**by**  $blast$   
**moreover have**  $cnt\text{-}Hom.Op_0.Cod$   
 $(cnt\text{-}Hom.UF.map_0$   
 $(cnt\text{-}Hom.Op_0.MkArr (F_o x) (hom_o x) (\tau x))) =$   
 $exp (hom_o x) (F_o a)$   
**using**  $assms\ x\ cnt\text{-}Hom.UF.map_0\text{-}def$   
**apply**  $auto[1]$   
**using**  $F.preserves\text{-}Obj\ cnt\text{-}Hom.Op_0.arr\text{-}MkArr\ \tau.component\text{-}in\text{-}hom$   
**by**  $blast$   
**moreover have**  $cnt\text{-}Hom.Op_0.Map$   
 $(cnt\text{-}Hom.UF.map_0$   
 $(cnt\text{-}Hom.Op_0.MkArr (F_o x) (hom_o x) (\tau x))) =$   
 $cnt\text{-}Hom.hom_a (F_o x) (hom_o x) \cdot \tau x$   
**using**  $assms\ x\ cnt\text{-}Hom.UF.map_0\text{-}def\ F.preserves\text{-}Obj$   
**by** (*simp*  $add: \tau.component\text{-}in\text{-}hom$ )  
**ultimately show**  $?thesis$  **by**  $argo$   
**qed**  
**also have**  $\dots = Exp^{\leftarrow} (\tau x \downarrow [hom_o x, F_o x]) (F_o a)$   
**using**  $assms(2)\ x\ \tau.component\text{-}in\text{-}hom\ [of\ x]\ F.preserves\text{-}Obj$   
 $C.DN\text{-}def\ C.cnt\text{-}Exp\text{-}DN$   
**by**  $fastforce$   
**finally show**  $?thesis$  **by**  $simp$   
**qed**  
**show**  $?thesis$   
**using**  $1\ 2\ 3$  **by**  $auto$   
**qed**

**lemma**  $transformation\text{-}generated\text{-}by\text{-}element$ :

**assumes**  $enriched\text{-}natural\text{-}transformation\ C\ T\ \alpha\ \iota$

$Obj\ Hom\ Id\ Comp\ (Collect\ ide)\ exp\ C.Id\ C.Comp$

$hom_o\ hom_a\ F_o\ F_a\ \tau$

**and**  $a \in Obj$

**shows**  $\tau a = generated\text{-}transformation\ (generating\text{-}elem\ \tau)\ a$

**proof** –

**interpret**  $\tau: enriched\text{-}natural\text{-}transformation\ C\ T\ \alpha\ \iota$

$Obj\ Hom\ Id\ Comp\ \langle Collect\ ide \rangle\ exp\ C.Id\ C.Comp$

$hom_o\ hom_a\ F_o\ F_a\ \tau$

**using**  $assms(1)$  **by**  $blast$

**have**  $\tau a \downarrow [hom_o a, F_o a] =$

$\tau a \downarrow [hom_o a, F_o a] \cdot$

$eval \mathcal{I} (hom_o a) \cdot r^{-1}[exp \mathcal{I} (hom_o a)] \cdot C.Exp (Id x) (hom_o a) \cdot$   
 $Curry[hom_o a, hom_o x, hom_o a] (Comp x x a)$   
**using** *assms(2) x top-path  $\tau$ .component-in-hom [of a] F.preserves-Obj*  
*comp-arr-dom C.UP-DN(2)*  
**by auto**  
**also have ... =**  
 $(\tau a \downarrow [hom_o a, F_o a] \cdot eval \mathcal{I} (hom_o a) \cdot r^{-1}[exp \mathcal{I} (hom_o a)]) \cdot$   
 $C.Exp (Id x) (hom_o a) \cdot$   
 $Curry[hom_o a, hom_o x, hom_o a] (Comp x x a)$   
**using comp-assoc by simp**  
**also have ... =**  
 $(eval \mathcal{I} (F_o a) \cdot r^{-1}[exp \mathcal{I} (F_o a)] \cdot$   
 $C.Exp \mathcal{I} (Uncurry[hom_o a, F_o a] (\tau a) \cdot l^{-1}[hom_o a])) \cdot$   
 $C.Exp (Id x) (hom_o a) \cdot$   
 $Curry[hom_o a, hom_o x, hom_o a] (Comp x x a)$   
**using assms right-square C.DN-def  $\tau$ .component-in-hom comp-assoc**  
**by auto blast**  
**also have ... =**  
 $eval \mathcal{I} (F_o a) \cdot r^{-1}[exp \mathcal{I} (F_o a)] \cdot$   
 $(C.Exp \mathcal{I} (Uncurry[hom_o a, F_o a] (\tau a) \cdot l^{-1}[hom_o a]) \cdot$   
 $C.Exp (Id x) (hom_o a)) \cdot$   
 $Curry[hom_o a, hom_o x, hom_o a] (Comp x x a)$   
**using comp-assoc by simp**  
**also have ... =**  
 $eval \mathcal{I} (F_o a) \cdot r^{-1}[exp \mathcal{I} (F_o a)] \cdot$   
 $(C.Exp (Id x) (F_o a) \cdot$   
 $C.Exp (hom_o x) (Uncurry[hom_o a, F_o a] (\tau a) \cdot l^{-1}[hom_o a])) \cdot$   
 $Curry[hom_o a, hom_o x, hom_o a] (Comp x x a)$   
**using assms center-square C.DN-def**  
*enriched-natural-transformation.component-in-hom*  
**by fastforce**  
**also have ... =**  
 $eval \mathcal{I} (F_o a) \cdot r^{-1}[exp \mathcal{I} (F_o a)] \cdot C.Exp (Id x) (F_o a) \cdot$   
 $C.Exp (hom_o x) (Uncurry[hom_o a, F_o a] (\tau a) \cdot l^{-1}[hom_o a]) \cdot$   
 $Curry[hom_o a, hom_o x, hom_o a] (Comp x x a)$   
**using comp-assoc by simp**  
**also have ... =**  
 $eval \mathcal{I} (F_o a) \cdot r^{-1}[exp \mathcal{I} (F_o a)] \cdot C.Exp (Id x) (F_o a) \cdot$   
 $Exp^{\leftarrow} (Uncurry[hom_o x, F_o x] (\tau x) \cdot l^{-1}[hom_o x]) (F_o a) \cdot F_a x a$   
**proof –**  
**have**  $eval \mathcal{I} (F_o a) \cdot r^{-1}[exp \mathcal{I} (F_o a)] \cdot C.Exp (Id x) (F_o a) \cdot$   
 $C.Exp (hom_o x) (Uncurry[hom_o a, F_o a] (\tau a) \cdot l^{-1}[hom_o a]) \cdot$   
 $Curry[hom_o a, hom_o x, hom_o a] (Comp x x a) =$   
 $eval \mathcal{I} (F_o a) \cdot r^{-1}[exp \mathcal{I} (F_o a)] \cdot C.Exp (Id x) (F_o a) \cdot$   
 $C.Exp (hom_o x) (Uncurry[hom_o a, F_o a] (\tau a) \cdot l^{-1}[hom_o a]) \cdot$   
 $hom_a x a$   
**using assms(2) x by force**  
**also have ... =**  
 $eval \mathcal{I} (F_o a) \cdot r^{-1}[exp \mathcal{I} (F_o a)] \cdot C.Exp (Id x) (F_o a) \cdot$

$$\text{Exp}^\rightarrow (\text{hom}_o x) (\text{Uncurry}[\text{hom}_o a, F_o a] (\tau a) \cdot 1^{-1}[\text{hom}_o a]) \cdot \text{hom}_a x a$$
**using** *assms*  $x$  *C.Exp-def C.cnt-Exp-ide comp-arr-dom* **by** *auto*

**also have** ... =
$$\text{eval } \mathcal{I} (F_o a) \cdot \text{r}^{-1}[\text{exp } \mathcal{I} (F_o a)] \cdot \text{C.Exp } (\text{Id } x) (F_o a) \cdot \text{Exp}^\rightarrow (\text{hom}_o x) (\tau a^\dagger[\text{hom}_o a, F_o a]) \cdot \text{hom}_a x a$$
**using** *assms*  $x$  *C.DN-def* **by** *fastforce*

**also have** ... =
$$\text{eval } \mathcal{I} (F_o a) \cdot \text{r}^{-1}[\text{exp } \mathcal{I} (F_o a)] \cdot \text{C.Exp } (\text{Id } x) (F_o a) \cdot \text{Exp}^\leftarrow (\tau x^\dagger[\text{hom}_o x, F_o x]) (F_o a) \cdot F_a x a$$
**using** *assms*(2) *left-square*  $\tau$ .*enriched-natural-transformation-axioms* **by** *fastforce*

**also have** ... =
$$\text{eval } \mathcal{I} (F_o a) \cdot \text{r}^{-1}[\text{exp } \mathcal{I} (F_o a)] \cdot \text{C.Exp } (\text{Id } x) (F_o a) \cdot \text{Exp}^\leftarrow (\text{Uncurry}[\text{hom}_o x, F_o x] (\tau x) \cdot 1^{-1}[\text{hom}_o x]) (F_o a) \cdot F_a x a$$
**using** *C.DN-def* **by** *fastforce*

**finally show** *?thesis* **by** *blast*

**qed**

**also have** ... =
$$\text{eval } \mathcal{I} (F_o a) \cdot \text{r}^{-1}[\text{exp } \mathcal{I} (F_o a)] \cdot (\text{C.Exp } (\text{Id } x) (F_o a) \cdot \text{Exp}^\leftarrow (\text{Uncurry}[\text{hom}_o x, F_o x] (\tau x) \cdot 1^{-1}[\text{hom}_o x]) (F_o a)) \cdot F_a x a$$
**using** *comp-assoc* **by** *simp*

**also have** ... =
$$\text{eval } \mathcal{I} (F_o a) \cdot \text{r}^{-1}[\text{exp } \mathcal{I} (F_o a)] \cdot (\text{Exp}^\leftarrow (\text{Id } x) (F_o a) \cdot \text{Exp}^\leftarrow (\text{Uncurry}[\text{hom}_o x, F_o x] (\tau x) \cdot 1^{-1}[\text{hom}_o x]) (F_o a)) \cdot F_a x a$$
**using** *assms*  $x$  *F.preserves-Obj C.Exp-def C.cov-Exp-ide comp-cod-arr* [of  $\text{Exp}^\leftarrow (\text{Id } x) (\text{dom } (F_o a))$ ] **by** *auto*

**also have** ... =
$$\text{eval } \mathcal{I} (F_o a) \cdot \text{r}^{-1}[\text{exp } \mathcal{I} (F_o a)] \cdot \text{Exp}^\leftarrow ((\text{Uncurry}[\text{hom}_o x, F_o x] (\tau x) \cdot 1^{-1}[\text{hom}_o x]) \cdot \text{Id } x) (F_o a) \cdot F_a x a$$

**proof** –

**have** *seq*  $(\text{Uncurry}[\text{hom}_o x, F_o x] (\tau x) \cdot 1^{-1}[\text{hom}_o x]) (\text{Id } x)$ 
**using** *assms*  $x$  *F.preserves-Obj Id-in-hom*  $\tau$ .*component-in-hom*

**apply** (*intro seqI*)

**apply** *auto*[1]

**by** *force+*

**thus** *?thesis*

**using** *assms*  $x$  *F.preserves-Obj C.cnt-Exp-comp* **by** *simp*

**qed**

**also have** ... =  $\text{eval } \mathcal{I} (F_o a) \cdot \text{r}^{-1}[\text{exp } \mathcal{I} (F_o a)] \cdot \text{Exp}^\leftarrow (\text{generating-elem } \tau) (F_o a) \cdot F_a x a$

**using**  $x$  *C.DN-def comp-assoc*  $\tau$ .*component-in-hom* **by** *fastforce*

**also have** 1: ... =

$(\text{generated-transformation } (\text{generating-elem } \tau) a) \downarrow[\text{hom}_o a, F_o a]$   
**using** *assms*  $x F.\text{preserves-Obj } C.UP-DN(4) \tau.\text{component-in-hom calculation}$   
*ide-Hom*  
**by** *(metis (no-types, lifting) mem-Collect-eq)*  
**finally have**  $*$ :  $(\tau a) \downarrow[\text{hom}_o a, F_o a] =$   
 $(\text{generated-transformation } (\text{generating-elem } \tau) a)$   
 $\downarrow[\text{hom}_o a, F_o a]$   
**by** *blast*  
**have**  $\tau a = ((\tau a) \downarrow[\text{hom}_o a, F_o a])^\uparrow$   
**using** *assms*  $x \tau.\text{component-in-hom ide-Hom } F.\text{preserves-Obj}$  **by** *auto*  
**also have**  $\dots = ((\text{generated-transformation } (\text{generating-elem } \tau) a)$   
 $\downarrow[\text{hom}_o a, F_o a])^\uparrow$   
**using**  $*$  **by** *argo*  
**also have**  $\dots = \text{generated-transformation } (\text{generating-elem } \tau) a$   
**using** *assms*  $x 1 \text{ide-Hom}$  **by** *presburger*  
**finally show**  $\tau a = \text{generated-transformation } (\text{generating-elem } \tau) a$  **by** *blast*  
**qed**

**lemma** *element-of-generated-transformation*:

**assumes**  $e \in \text{hom } \mathcal{I} (F_o x)$

**shows** *generating-elem* (*generated-transformation*  $e$ ) =  $e$

**proof** –

**have** *generating-elem* (*generated-transformation*  $e$ ) =  
 $\text{Uncurry}[\text{hom}_o x, F_o x]$   
 $((\text{eval } \mathcal{I} (F_o x) \cdot \text{r}^{-1}[\text{exp } \mathcal{I} (F_o x)]) \cdot$   
 $\text{Curry}[\text{exp } (F_o x) (F_o x), \mathcal{I}, F_o x]$   
 $(\text{eval } (F_o x) (F_o x) \cdot (\text{exp } (F_o x) (F_o x) \otimes e)) \cdot F_a x x)^\uparrow) \cdot$   
 $\text{l}^{-1}[\text{hom}_o x] \cdot \text{Id } x$

**proof** –

**have** *arr*  $((\text{eval } \mathcal{I} (F_o x) \cdot$   
 $\text{r}^{-1}[\text{exp } \mathcal{I} (F_o x)] \cdot$   
 $\text{Curry}[\text{exp } (F_o x) (F_o x), \mathcal{I}, F_o x]$   
 $(\text{eval } (F_o x) (F_o x) \cdot (\text{exp } (F_o x) (F_o x) \otimes e)) \cdot F_a x x)^\uparrow)$

**using** *assms*  $x F.\text{preserves-Hom } F.\text{preserves-Obj}$

**apply** *(intro C.UP-simps seqI)*

**apply** *auto[1]*

**by** *fastforce+*

**thus** *?thesis*

**using** *assms*  $x C.DN-def \text{comp-assoc}$  **by** *auto*

**qed**

**also have**  $\dots =$

$\text{Uncurry}[\text{hom}_o x, F_o x]$   
 $((\text{eval } \mathcal{I} (F_o x) \cdot$   
 $(\text{r}^{-1}[\text{exp } \mathcal{I} (F_o x)] \cdot$   
 $\text{Curry}[\text{exp } (F_o x) (F_o x), \mathcal{I}, F_o x]$   
 $(\text{eval } (F_o x) (F_o x) \cdot (\text{exp } (F_o x) (F_o x) \otimes e))) \cdot F_a x x)^\uparrow) \cdot$   
 $\text{l}^{-1}[\text{hom}_o x] \cdot \text{Id } x$

**using** *comp-assoc* **by** *simp*

**also have**  $\dots =$

$$\begin{aligned} & \text{Uncurry}[hom_o x, F_o x] \\ & ((\text{eval } \mathcal{I} (F_o x) \cdot \\ & \quad ((\text{Curry}[\text{exp } (F_o x) (F_o x), \mathcal{I}, F_o x] \\ & \quad (\text{eval } (F_o x) (F_o x) \cdot (\text{exp } (F_o x) (F_o x) \otimes e)) \otimes \mathcal{I}) \cdot \\ & \quad r^{-1}[\text{exp } (F_o x) (F_o x)]) \cdot \\ & \quad F_a x x)^\dagger) \cdot \\ & l^{-1}[hom_o x] \cdot Id x \end{aligned}$$

**proof** –

**have** « $\text{Curry}[\text{exp } (F_o x) (F_o x), \mathcal{I}, F_o x]$   
 $(\text{eval } (F_o x) (F_o x) \cdot (\text{exp } (F_o x) (F_o x) \otimes e))$   
 $: \text{exp } (F_o x) (F_o x) \rightarrow \text{exp } \mathcal{I} (F_o x)$ »

**using** *assms x F.preserves-Obj C.ide-exp*

**by** (*intro C.Curry-in-hom*) *auto*

**thus** *?thesis*

**using** *assms*

*runit'-naturality*

$[of \text{Curry}[\text{exp } (F_o x) (F_o x), \mathcal{I}, F_o x]$   
 $(\text{eval } (F_o x) (F_o x) \cdot (\text{exp } (F_o x) (F_o x) \otimes e))]$

**by force**

**qed**

**also have** ... =

$$\begin{aligned} & \text{Uncurry}[hom_o x, F_o x] \\ & ((\text{Uncurry}[\mathcal{I}, F_o x] \\ & \quad (\text{Curry}[\text{exp } (F_o x) (F_o x), \mathcal{I}, F_o x] \\ & \quad (\text{eval } (F_o x) (F_o x) \cdot (\text{exp } (F_o x) (F_o x) \otimes e))) \cdot \\ & \quad r^{-1}[\text{exp } (F_o x) (F_o x)] \cdot F_a x x)^\dagger) \cdot \\ & l^{-1}[hom_o x] \cdot Id x \end{aligned}$$

**using** *comp-assoc by simp*

**also have** ... =

$$\begin{aligned} & \text{Uncurry}[hom_o x, F_o x] \\ & (((\text{eval } (F_o x) (F_o x) \cdot (\text{exp } (F_o x) (F_o x) \otimes e)) \cdot \\ & \quad r^{-1}[\text{exp } (F_o x) (F_o x)] \cdot F_a x x)^\dagger) \cdot \\ & l^{-1}[hom_o x] \cdot Id x \end{aligned}$$

**using** *assms x F.preserves-Obj C.Uncurry-Curry by auto*

**also have** ... =

$$\begin{aligned} & \text{Uncurry}[hom_o x, F_o x] \\ & (((\text{eval } (F_o x) (F_o x) \cdot (\text{exp } (F_o x) (F_o x) \otimes e)) \cdot \\ & \quad (F_a x x \otimes \mathcal{I}) \cdot r^{-1}[hom_o x])^\dagger) \cdot \\ & l^{-1}[hom_o x] \cdot Id x \end{aligned}$$

**using** *assms x runit'-naturality F.preserves-Hom [of x x] by fastforce*

**also have** ... =

$$\begin{aligned} & \text{Uncurry}[hom_o x, F_o x] \\ & (((\text{eval } (F_o x) (F_o x) \cdot (\text{exp } (F_o x) (F_o x) \otimes e) \cdot (F_a x x \otimes \mathcal{I})) \cdot \\ & \quad r^{-1}[hom_o x])^\dagger) \cdot \\ & l^{-1}[hom_o x] \cdot Id x \end{aligned}$$

**using** *comp-assoc by simp*

**also have** ... =

$$\begin{aligned} & \text{Uncurry}[hom_o x, F_o x] \\ & (((\text{eval } (F_o x) (F_o x) \cdot (F_a x x \otimes e)) \cdot r^{-1}[hom_o x])^\dagger) \cdot \\ & l^{-1}[hom_o x] \cdot Id x \end{aligned}$$



```

    l-1[homo x] · Id x
using assms x F.preserves-Hom [of x x] comp-arr-dom [of e I] comp-cod-arr
    interchange
by fastforce
also have ... =
    Uncurry[homo x, Fo x]
    (Curry[I, homo x, Fo x]
    (((eval (Fo x) (Fo x) · (Fa x x ⊗ e)) · r-1[homo x]) · l[homo x])) ·
    l-1[homo x] · Id x
proof –
    have seq (eval (Fo x) (Fo x) · (Fa x x ⊗ e)) r-1[Hom x x]
    using assms x F.preserves-Obj F.preserves-Hom by blast
    thus ?thesis
    using assms x C.UP-def F.preserves-Obj by auto
qed
also have ... =
    (((eval (Fo x) (Fo x) · (Fa x x ⊗ e)) · r-1[Hom x x]) · l[Hom x x]) ·
    l-1[Hom x x] · Id x
    using assms x C.Uncurry-Curry F.preserves-Obj F.preserves-Hom by force
also have ... =
    eval (Fo x) (Fo x) · (Fa x x ⊗ e) · r-1[Hom x x] ·
    (l[Hom x x] · l-1[Hom x x]) · Id x
    using comp-assoc by simp
also have ... = eval (Fo x) (Fo x) · (Fa x x ⊗ e) · r-1[Hom x x] · Id x
    using assms x ide-Hom Id-in-hom comp-lunit-lunit'(1) comp-cod-arr
    by fastforce
also have ... = eval (Fo x) (Fo x) · (Fa x x ⊗ e) · (Id x ⊗ I) · r-1[I]
    using x Id-in-hom runit'-naturality by fastforce
also have ... = eval (Fo x) (Fo x) · ((Fa x x ⊗ e) · (Id x ⊗ I)) · r-1[I]
    using comp-assoc by simp
also have ... = eval (Fo x) (Fo x) · (Fa x x · Id x ⊗ e) · r-1[I]
    using assms x interchange [of Fa x x Id x e I] F.preserves-Hom
    comp-arr-dom Id-in-hom
    by fastforce
also have ... = eval (Fo x) (Fo x) · (C.Id (Fo x) ⊗ e) · r-1[I]
    using x F.preserves-Id by auto
also have ... =
    eval (Fo x) (Fo x) · ((C.Id (Fo x) ⊗ Fo x) · (I ⊗ e)) · r-1[I]
    using assms x interchange C.Id-in-hom F.preserves-Obj comp-arr-dom
    comp-cod-arr
    by (metis in-homE mem-Collect-eq)
also have ... = Uncurry[Fo x, Fo x] (C.Id (Fo x)) · (I ⊗ e) · r-1[I]
    using comp-assoc by simp
also have ... = l[Fo x] · (I ⊗ e) · r-1[I]
    using x F.preserves-Obj C.Id-def C.Uncurry-Curry by fastforce
also have ... = l[Fo x] · (I ⊗ e) · l-1[I]
    using unitor-coincidence by simp
also have ... = l[Fo x] · l-1[Fo x] · e
    using assms lunit'-naturality by fastforce

```

```

also have ... = (l[Fo x] · l-1[Fo x]) · e
using comp-assoc by simp
also have ... = e
using assms x comp-lunit-lunit' F.preserves-Obj comp-cod-arr by auto
finally show generating-elem (generated-transformation e) = e
by blast
qed

```

We can now state and prove the (weak) covariant Yoneda lemma (Kelly, Section 1.9) for enriched categories.

```

theorem covariant-yoneda:
shows bij-betw generated-transformation
  (hom  $\mathcal{I}$  (Fo x))
  (Collect (enriched-natural-transformation C T  $\alpha$   $\iota$ 
    Obj Hom Id Comp (Collect ide) exp C.Id C.Comp
    homo homa Fo Fa))
proof (intro bij-betwI)
show generated-transformation  $\in$ 
  hom  $\mathcal{I}$  (Fo x)  $\rightarrow$  Collect
    (enriched-natural-transformation C T  $\alpha$   $\iota$ 
    Obj Hom Id Comp (Collect ide) exp C.Id C.Comp
    homo homa Fo Fa)
using enriched-natural-transformation-generated-transformation by blast
show generating-elem  $\in$ 
  Collect (enriched-natural-transformation C T  $\alpha$   $\iota$ 
    Obj Hom Id Comp (Collect ide) exp C.Id C.Comp
    homo homa Fo Fa)
   $\rightarrow$  hom  $\mathcal{I}$  (Fo x)
using generating-elem-in-hom by blast
show  $\bigwedge e. e \in \text{hom } \mathcal{I} (F_o x) \implies$ 
  generating-elem (generated-transformation e) = e
using element-of-generated-transformation by blast
show  $\bigwedge \tau. \tau \in \text{Collect (enriched-natural-transformation C T } \alpha \ \iota$ 
  Obj Hom Id Comp (Collect ide) exp C.Id C.Comp
  homo homa Fo Fa)
   $\implies$  generated-transformation (generating-elem  $\tau$ ) =  $\tau$ 
proof –
fix  $\tau$ 
assume  $\tau: \tau \in \text{Collect (enriched-natural-transformation C T } \alpha \ \iota$ 
  Obj Hom Id Comp (Collect ide) exp C.Id C.Comp
  homo homa Fo Fa)
interpret  $\tau: \text{enriched-natural-transformation C T } \alpha \ \iota$ 
  Obj Hom Id Comp (Collect ide) exp C.Id C.Comp
  homo homa Fo Fa  $\tau$ 
using  $\tau$  by blast
show generated-transformation (generating-elem  $\tau$ ) =  $\tau$ 
proof
fix a
show generated-transformation (generating-elem  $\tau$ ) a =  $\tau$  a

```

```

    using  $\tau$  transformation-generated-by-element  $\tau$ .extensionality
      F.extensionality C.UP-def not-arr-null null-is-zero(2)
    by (cases a  $\in$  Obj) auto
  qed
qed
qed
end

```

### 2.5.3 Contravariant Case

The (weak) contravariant Yoneda lemma is obtained by just replacing the enriched category by its opposite in the covariant version.

```

locale contravariant-yoneda-lemma =
  opposite-enriched-category C T  $\alpha$   $\iota$   $\sigma$  Obj Hom Id Comp +
  covariant-yoneda-lemma C T  $\alpha$   $\iota$   $\sigma$  exp eval Curry Obj Homop Id Compop y Fo
Fa
for C :: 'a  $\Rightarrow$  'a  $\Rightarrow$  'a (infixr <> 55)
and T :: 'a  $\times$  'a  $\Rightarrow$  'a
and  $\alpha$  :: 'a  $\times$  'a  $\times$  'a  $\Rightarrow$  'a
and  $\iota$  :: 'a
and  $\sigma$  :: 'a  $\times$  'a  $\Rightarrow$  'a
and exp :: 'a  $\Rightarrow$  'a  $\Rightarrow$  'a
and eval :: 'a  $\Rightarrow$  'a  $\Rightarrow$  'a
and Curry :: 'a  $\Rightarrow$  'a  $\Rightarrow$  'a  $\Rightarrow$  'a  $\Rightarrow$  'a
and Obj :: 'b set
and Hom :: 'b  $\Rightarrow$  'b  $\Rightarrow$  'a
and Id :: 'b  $\Rightarrow$  'a
and Comp :: 'b  $\Rightarrow$  'b  $\Rightarrow$  'b  $\Rightarrow$  'a
and y :: 'b
and Fo :: 'b  $\Rightarrow$  'a
and Fa :: 'b  $\Rightarrow$  'b  $\Rightarrow$  'a
begin

  corollary contravariant-yoneda:
  shows bij-betw generated-transformation
    (hom  $\mathcal{I}$  (Fo y))
    (Collect
      (enriched-natural-transformation
        C T  $\alpha$   $\iota$  Obj Homop Id Compop (Collect ide) exp C.Id C.Comp
        homo homa Fo Fa))
  using covariant-yoneda by blast

end

end

```

# Bibliography

- [1] G. M. Kelly. Basic concepts of enriched category theory. *Reprints in Theory and Applications of Categories*, 10, 2005. <http://www.tac.mta.ca/tac/reprints/articles/10/tr10.pdf>.
- [2] nLab. internal hom. *nLab (various contributors)*, 2009 – 2024. <https://ncatlab.org/nlab/show/internal+hom>, [Online; accessed 22-May-2024].
- [3] E. W. Stark. Monoidal categories. *Archive of Formal Proofs*, May 2017. <https://isa-afp.org/entries/MonoidalCategory.html>, Formal proof development.
- [4] E. W. Stark. Residuated transition systems II: Categorical properties. *Archive of Formal Proofs*, June 2024. (submitted for publication).