

MonoidalCategories

**Monoidal and monoidal (co)closed
categories**

2024.06-01

9 June 2024

Mohamed Barakat

Sebastian Gutsche

Sebastian Posur

Tom Kuhmichel

Fabian Zickgraf

Mohamed Barakat

Email: mohamed.barakat@uni-siegen.de

Homepage: <http://algebra.mathematik.uni-siegen.de/barakat/>

Address: Walter-Flex-Str. 3
57068 Siegen
Germany

Sebastian Gutsche

Email: gutsche@mathematik.uni-siegen.de

Homepage: <http://algebra.mathematik.uni-siegen.de/gutsche/>

Address: Department Mathematik
Universität Siegen
Walter-Flex-Straße 3
57068 Siegen
Germany

Sebastian Posur

Email: sebastian.posur@uni-siegen.de

Homepage: <http://algebra.mathematik.uni-siegen.de/posur/>

Address: Department Mathematik
Universität Siegen
Walter-Flex-Straße 3
57068 Siegen
Germany

Tom Kuhmichel

Email: tom.kuhmichel@student.uni-siegen.de

Homepage: <https://github.com/TKuh>

Address: Department Mathematik
Universität Siegen
Walter-Flex-Straße 3
57068 Siegen
Germany

Fabian Zickgraf

Email: fabian.zickgraf@uni-siegen.de

Homepage: <https://github.com/zickgraf/>

Address: Walter-Flex-Str. 3
57068 Siegen
Germany

Contents

1	Monoidal Categories	3
1.1	Monoidal Categories	3
1.2	Additive Monoidal Categories	6
1.3	Braided Monoidal Categories	7
1.4	Symmetric Monoidal Categories	8
1.5	Left Closed Monoidal Categories	8
1.6	Closed Monoidal Categories	15
1.7	Left Coclosed Monoidal Categories	25
1.8	Coclosed Monoidal Categories	32
1.9	Symmetric Closed Monoidal Categories	40
1.10	Symmetric Coclosed Monoidal Categories	40
1.11	Rigid Symmetric Closed Monoidal Categories	40
1.12	Rigid Symmetric Coclosed Monoidal Categories	42
1.13	Convenience Methods	45
1.14	Add-methods	46
2	Examples and Tests	90
2.1	Test functions	90
2.2	Basics	97
3	Code Generation for Monoidal Categories	99
3.1	Monoidal Categories	99
3.2	Closed Monoidal Categories	99
3.3	Coclosed Monoidal Categories	100
4	The terminal category with multiple objects	101
4.1	Constructors	101
4.2	GAP Categories	101
5	Legacy Operations and Synonyms	102
5.1	Legacy operations	102
5.2	Synonyms for legacy operations	103
6	MonoidalCategories automatic generated documentation	106
6.1	MonoidalCategories automatic generated documentation of properties	106
	Index	108

Chapter 1

Monoidal Categories

1.1 Monoidal Categories

A 6-tuple $(\mathbf{C}, \otimes, 1, \alpha, \lambda, \rho)$ consisting of

- a category \mathbf{C} ,
- a functor $\otimes : \mathbf{C} \times \mathbf{C} \rightarrow \mathbf{C}$ compatible with the congruence of morphisms,
- an object $1 \in \mathbf{C}$,
- a natural isomorphism $\alpha_{a,b,c} : a \otimes (b \otimes c) \cong (a \otimes b) \otimes c$,
- a natural isomorphism $\lambda_a : 1 \otimes a \cong a$,
- a natural isomorphism $\rho_a : a \otimes 1 \cong a$,

is called a *monoidal category*, if

- for all objects a, b, c, d , the pentagon identity holds:

$$(\alpha_{a,b,c} \otimes \text{id}_d) \circ \alpha_{a,b \otimes c,d} \circ (\text{id}_a \otimes \alpha_{b,c,d}) \sim \alpha_{a \otimes b,c,d} \circ \alpha_{a,b,c \otimes d},$$

- for all objects a, c , the triangle identity holds:

$$(\rho_a \otimes \text{id}_c) \circ \alpha_{a,1,c} \sim \text{id}_a \otimes \lambda_c.$$

The corresponding GAP property is given by `IsMonoidalCategory`.

1.1.1 TensorProductOnMorphisms (for IsCapCategoryMorphism, IsCapCategory-Morphism)

▷ `TensorProductOnMorphisms(alpha, beta)` (operation)

Returns: a morphism in $\text{Hom}(a \otimes b, a' \otimes b')$

The arguments are two morphisms $\alpha : a \rightarrow a', \beta : b \rightarrow b'$. The output is the tensor product $\alpha \otimes \beta$.

1.1.2 TensorProductOnMorphismsWithGivenTensorProducts (for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryObject)

▷ TensorProductOnMorphismsWithGivenTensorProducts(s , α , β , r) (operation)

Returns: a morphism in $\text{Hom}(a \otimes b, a' \otimes b')$

The arguments are an object $s = a \otimes b$, two morphisms $\alpha : a \rightarrow a'$, $\beta : b \rightarrow b'$, and an object $r = a' \otimes b'$. The output is the tensor product $\alpha \otimes \beta$.

1.1.3 AssociatorRightToLeft (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ AssociatorRightToLeft(a , b , c) (operation)

Returns: a morphism in $\text{Hom}(a \otimes (b \otimes c), (a \otimes b) \otimes c)$.

The arguments are three objects a, b, c . The output is the associator $\alpha_{a,(b,c)} : a \otimes (b \otimes c) \rightarrow (a \otimes b) \otimes c$.

1.1.4 AssociatorRightToLeftWithGivenTensorProducts (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ AssociatorRightToLeftWithGivenTensorProducts(s , a , b , c , r) (operation)

Returns: a morphism in $\text{Hom}(a \otimes (b \otimes c), (a \otimes b) \otimes c)$.

The arguments are an object $s = a \otimes (b \otimes c)$, three objects a, b, c , and an object $r = (a \otimes b) \otimes c$. The output is the associator $\alpha_{a,(b,c)} : a \otimes (b \otimes c) \rightarrow (a \otimes b) \otimes c$.

1.1.5 AssociatorLeftToRight (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ AssociatorLeftToRight(a , b , c) (operation)

Returns: a morphism in $\text{Hom}((a \otimes b) \otimes c \rightarrow a \otimes (b \otimes c))$.

The arguments are three objects a, b, c . The output is the associator $\alpha_{(a,b),c} : (a \otimes b) \otimes c \rightarrow a \otimes (b \otimes c)$.

1.1.6 AssociatorLeftToRightWithGivenTensorProducts (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ AssociatorLeftToRightWithGivenTensorProducts(s , a , b , c , r) (operation)

Returns: a morphism in $\text{Hom}((a \otimes b) \otimes c \rightarrow a \otimes (b \otimes c))$.

The arguments are an object $s = (a \otimes b) \otimes c$, three objects a, b, c , and an object $r = a \otimes (b \otimes c)$. The output is the associator $\alpha_{(a,b),c} : (a \otimes b) \otimes c \rightarrow a \otimes (b \otimes c)$.

1.1.7 LeftUnitor (for IsCapCategoryObject)

▷ LeftUnitor(a) (attribute)

Returns: a morphism in $\text{Hom}(1 \otimes a, a)$

The argument is an object a . The output is the left unitor $\lambda_a : 1 \otimes a \rightarrow a$.

1.1.8 LeftUnitorWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject)

▷ LeftUnitorWithGivenTensorProduct(a, s) (operation)

Returns: a morphism in $\text{Hom}(1 \otimes a, a)$

The arguments are an object a and an object $s = 1 \otimes a$. The output is the left unitor $\lambda_a : 1 \otimes a \rightarrow a$.

1.1.9 LeftUnitorInverse (for IsCapCategoryObject)

▷ LeftUnitorInverse(a) (attribute)

Returns: a morphism in $\text{Hom}(a, 1 \otimes a)$

The argument is an object a . The output is the inverse of the left unitor $\lambda_a^{-1} : a \rightarrow 1 \otimes a$.

1.1.10 LeftUnitorInverseWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject)

▷ LeftUnitorInverseWithGivenTensorProduct(a, r) (operation)

Returns: a morphism in $\text{Hom}(a, 1 \otimes a)$

The argument is an object a and an object $r = 1 \otimes a$. The output is the inverse of the left unitor $\lambda_a^{-1} : a \rightarrow 1 \otimes a$.

1.1.11 RightUnitor (for IsCapCategoryObject)

▷ RightUnitor(a) (attribute)

Returns: a morphism in $\text{Hom}(a \otimes 1, a)$

The argument is an object a . The output is the right unitor $\rho_a : a \otimes 1 \rightarrow a$.

1.1.12 RightUnitorWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject)

▷ RightUnitorWithGivenTensorProduct(a, s) (operation)

Returns: a morphism in $\text{Hom}(a \otimes 1, a)$

The arguments are an object a and an object $s = a \otimes 1$. The output is the right unitor $\rho_a : a \otimes 1 \rightarrow a$.

1.1.13 RightUnitorInverse (for IsCapCategoryObject)

▷ RightUnitorInverse(a) (attribute)

Returns: a morphism in $\text{Hom}(a, a \otimes 1)$

The argument is an object a . The output is the inverse of the right unitor $\rho_a^{-1} : a \rightarrow a \otimes 1$.

1.1.14 RightUnitorInverseWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject)

▷ RightUnitorInverseWithGivenTensorProduct(a, r) (operation)

Returns: a morphism in $\text{Hom}(a, a \otimes 1)$

The arguments are an object a and an object $r = a \otimes 1$. The output is the inverse of the right unitor $\rho_a^{-1} : a \rightarrow a \otimes 1$.

1.1.15 TensorProductOnObjects (for IsCapCategoryObject, IsCapCategoryObject)

▷ `TensorProductOnObjects(a, b)` (operation)

Returns: an object

The arguments are two objects a, b . The output is the tensor product $a \otimes b$.

1.1.16 AddTensorProductOnObjects (for IsCapCategory, IsFunction)

▷ `AddTensorProductOnObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `TensorProductOnObjects`. $F : (a, b) \mapsto a \otimes b$.

1.1.17 TensorUnit (for IsCapCategory)

▷ `TensorUnit(C)` (attribute)

Returns: an object

The argument is a category C . The output is the tensor unit 1 of C .

1.1.18 AddTensorUnit (for IsCapCategory, IsFunction)

▷ `AddTensorUnit(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `TensorUnit`. $F : () \mapsto 1$.

1.2 Additive Monoidal Categories

1.2.1 LeftDistributivityExpanding (for IsCapCategoryObject, IsList)

▷ `LeftDistributivityExpanding(a, L)` (operation)

Returns: a morphism in $\text{Hom}(a \otimes (b_1 \oplus \dots \oplus b_n), (a \otimes b_1) \oplus \dots \oplus (a \otimes b_n))$

The arguments are an object a and a list of objects $L = (b_1, \dots, b_n)$. The output is the left distributivity morphism $a \otimes (b_1 \oplus \dots \oplus b_n) \rightarrow (a \otimes b_1) \oplus \dots \oplus (a \otimes b_n)$.

1.2.2 LeftDistributivityExpandingWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsList, IsCapCategoryObject)

▷ `LeftDistributivityExpandingWithGivenObjects(s, a, L, r)` (operation)

Returns: a morphism in $\text{Hom}(s, r)$

The arguments are an object $s = a \otimes (b_1 \oplus \dots \oplus b_n)$, an object a , a list of objects $L = (b_1, \dots, b_n)$, and an object $r = (a \otimes b_1) \oplus \dots \oplus (a \otimes b_n)$. The output is the left distributivity morphism $s \rightarrow r$.

1.2.3 LeftDistributivityFactoring (for IsCapCategoryObject, IsList)

▷ `LeftDistributivityFactoring(a, L)` (operation)

Returns: a morphism in $\text{Hom}((a \otimes b_1) \oplus \dots \oplus (a \otimes b_n), a \otimes (b_1 \oplus \dots \oplus b_n))$

The arguments are an object a and a list of objects $L = (b_1, \dots, b_n)$. The output is the left distributivity morphism $(a \otimes b_1) \oplus \dots \oplus (a \otimes b_n) \rightarrow a \otimes (b_1 \oplus \dots \oplus b_n)$.

1.2.4 LeftDistributivityFactoringWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsList, IsCapCategoryObject)

▷ LeftDistributivityFactoringWithGivenObjects(s , a , L , r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$

The arguments are an object $s = (a \otimes b_1) \oplus \dots \oplus (a \otimes b_n)$, an object a , a list of objects $L = (b_1, \dots, b_n)$, and an object $r = a \otimes (b_1 \oplus \dots \oplus b_n)$. The output is the left distributivity morphism $s \rightarrow r$.

1.2.5 RightDistributivityExpanding (for IsList, IsCapCategoryObject)

▷ RightDistributivityExpanding(L , a) (operation)

Returns: a morphism in $\text{Hom}((b_1 \oplus \dots \oplus b_n) \otimes a, (b_1 \otimes a) \oplus \dots \oplus (b_n \otimes a))$

The arguments are a list of objects $L = (b_1, \dots, b_n)$ and an object a . The output is the right distributivity morphism $(b_1 \oplus \dots \oplus b_n) \otimes a \rightarrow (b_1 \otimes a) \oplus \dots \oplus (b_n \otimes a)$.

1.2.6 RightDistributivityExpandingWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject, IsCapCategoryObject)

▷ RightDistributivityExpandingWithGivenObjects(s , L , a , r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$

The arguments are an object $s = (b_1 \oplus \dots \oplus b_n) \otimes a$, a list of objects $L = (b_1, \dots, b_n)$, an object a , and an object $r = (b_1 \otimes a) \oplus \dots \oplus (b_n \otimes a)$. The output is the right distributivity morphism $s \rightarrow r$.

1.2.7 RightDistributivityFactoring (for IsList, IsCapCategoryObject)

▷ RightDistributivityFactoring(L , a) (operation)

Returns: a morphism in $\text{Hom}((b_1 \otimes a) \oplus \dots \oplus (b_n \otimes a), (b_1 \oplus \dots \oplus b_n) \otimes a)$

The arguments are a list of objects $L = (b_1, \dots, b_n)$ and an object a . The output is the right distributivity morphism $(b_1 \otimes a) \oplus \dots \oplus (b_n \otimes a) \rightarrow (b_1 \oplus \dots \oplus b_n) \otimes a$.

1.2.8 RightDistributivityFactoringWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject, IsCapCategoryObject)

▷ RightDistributivityFactoringWithGivenObjects(s , L , a , r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$

The arguments are an object $s = (b_1 \otimes a) \oplus \dots \oplus (b_n \otimes a)$, a list of objects $L = (b_1, \dots, b_n)$, an object a , and an object $r = (b_1 \oplus \dots \oplus b_n) \otimes a$. The output is the right distributivity morphism $s \rightarrow r$.

1.3 Braided Monoidal Categories

A monoidal category \mathbf{C} equipped with a natural isomorphism $B_{a,b} : a \otimes b \cong b \otimes a$ is called a *braided monoidal category* if

$$\bullet \lambda_a \circ B_{a,1} \sim \rho_a,$$

- $(B_{c,a} \otimes \text{id}_b) \circ \alpha_{c,a,b} \circ B_{a \otimes b, c} \sim \alpha_{a,c,b} \circ (\text{id}_a \otimes B_{b,c}) \circ \alpha_{a,b,c}^{-1}$,
- $(\text{id}_b \otimes B_{c,a}) \circ \alpha_{b,c,a}^{-1} \circ B_{a,b \otimes c} \sim \alpha_{b,a,c}^{-1} \circ (B_{a,b} \otimes \text{id}_c) \circ \alpha_{a,b,c}$.

The corresponding GAP property is given by `IsBraidedMonoidalCategory`.

1.3.1 Braiding (for `IsCapCategoryObject`, `IsCapCategoryObject`)

▷ `Braiding(a, b)` (operation)

Returns: a morphism in $\text{Hom}(a \otimes b, b \otimes a)$.

The arguments are two objects a, b . The output is the braiding $B_{a,b} : a \otimes b \rightarrow b \otimes a$.

1.3.2 BraidingWithGivenTensorProducts (for `IsCapCategoryObject`, `IsCapCategoryObject`, `IsCapCategoryObject`)

▷ `BraidingWithGivenTensorProducts(s, a, b, r)` (operation)

Returns: a morphism in $\text{Hom}(a \otimes b, b \otimes a)$.

The arguments are an object $s = a \otimes b$, two objects a, b , and an object $r = b \otimes a$. The output is the braiding $B_{a,b} : a \otimes b \rightarrow b \otimes a$.

1.3.3 BraidingInverse (for `IsCapCategoryObject`, `IsCapCategoryObject`)

▷ `BraidingInverse(a, b)` (operation)

Returns: a morphism in $\text{Hom}(b \otimes a, a \otimes b)$.

The arguments are two objects a, b . The output is the inverse braiding $B_{a,b}^{-1} : b \otimes a \rightarrow a \otimes b$.

1.3.4 BraidingInverseWithGivenTensorProducts (for `IsCapCategoryObject`, `IsCapCategoryObject`, `IsCapCategoryObject`)

▷ `BraidingInverseWithGivenTensorProducts(s, a, b, r)` (operation)

Returns: a morphism in $\text{Hom}(b \otimes a, a \otimes b)$.

The arguments are an object $s = b \otimes a$, two objects a, b , and an object $r = a \otimes b$. The output is the inverse braiding $B_{a,b}^{-1} : b \otimes a \rightarrow a \otimes b$.

1.4 Symmetric Monoidal Categories

A braided monoidal category \mathbf{C} is called *symmetric monoidal category* if $B_{a,b}^{-1} \sim B_{b,a}$. The corresponding GAP property is given by `IsSymmetricMonoidalCategory`.

1.5 Left Closed Monoidal Categories

A monoidal category \mathbf{C} which has for each functor $- \otimes b : \mathbf{C} \rightarrow \mathbf{C}$ a right adjoint (denoted by $\underline{\text{Hom}}_\ell(b, -)$) is called a *left closed monoidal category*.

If no operations involving left duals are installed manually, the left dual objects will be derived as $a^\vee := \underline{\text{Hom}}_\ell(a, 1)$.

The corresponding GAP property is called `IsLeftClosedMonoidalCategory`.

1.5.1 LeftInternalHomOnObjects (for IsCapCategoryObject, IsCapCategoryObject)

▷ LeftInternalHomOnObjects(a, b) (operation)

Returns: an object

The arguments are two objects a, b . The output is the internal hom object $\underline{\text{Hom}}_\ell(a, b)$.

1.5.2 LeftInternalHomOnMorphisms (for IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ LeftInternalHomOnMorphisms(α, β) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}_\ell(a', b), \underline{\text{Hom}}_\ell(a, b'))$

The arguments are two morphisms $\alpha : a \rightarrow a', \beta : b \rightarrow b'$. The output is the internal hom morphism $\underline{\text{Hom}}_\ell(\alpha, \beta) : \underline{\text{Hom}}_\ell(a', b) \rightarrow \underline{\text{Hom}}_\ell(a, b')$.

1.5.3 LeftInternalHomOnMorphismsWithGivenLeftInternalHoms (for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryObject)

▷ LeftInternalHomOnMorphismsWithGivenLeftInternalHoms(s, α, β, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$

The arguments are an object $s = \underline{\text{Hom}}_\ell(a', b)$, two morphisms $\alpha : a \rightarrow a', \beta : b \rightarrow b'$, and an object $r = \underline{\text{Hom}}_\ell(a, b')$. The output is the internal hom morphism $\underline{\text{Hom}}_\ell(\alpha, \beta) : \underline{\text{Hom}}_\ell(a', b) \rightarrow \underline{\text{Hom}}_\ell(a, b')$.

1.5.4 LeftClosedMonoidalEvaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ LeftClosedMonoidalEvaluationMorphism(a, b) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}_\ell(a, b) \otimes a, b)$.

The arguments are two objects a, b . The output is the evaluation morphism $\text{ev}_{a,b} : \underline{\text{Hom}}_\ell(a, b) \otimes a \rightarrow b$, i.e., the counit of the tensor hom adjunction.

1.5.5 LeftClosedMonoidalEvaluationMorphismWithGivenSource (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftClosedMonoidalEvaluationMorphismWithGivenSource(a, b, s) (operation)

Returns: a morphism in $\text{Hom}(s, b)$.

The arguments are two objects a, b and an object $s = \underline{\text{Hom}}_\ell(a, b) \otimes a$. The output is the evaluation morphism $\text{ev}_{a,b} : \underline{\text{Hom}}_\ell(a, b) \otimes a \rightarrow b$, i.e., the counit of the tensor hom adjunction.

1.5.6 LeftClosedMonoidalCoevaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ LeftClosedMonoidalCoevaluationMorphism(a, b) (operation)

Returns: a morphism in $\text{Hom}(b, \underline{\text{Hom}}_\ell(a, b \otimes a))$.

The arguments are two objects a, b . The output is the coevaluation morphism $\text{coev}_{a,b} : b \rightarrow \underline{\text{Hom}}_\ell(a, b \otimes a)$, i.e., the unit of the tensor hom adjunction.

1.5.7 LeftClosedMonoidalCoevaluationMorphismWithGivenRange (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftClosedMonoidalCoevaluationMorphismWithGivenRange(a, b, r) (operation)

Returns: a morphism in $\text{Hom}(b, r)$.

The arguments are two objects a, b and an object $r = \underline{\text{Hom}}_\ell(a, b \otimes a)$. The output is the coevaluation morphism $\text{coev}_{a,b} : b \rightarrow \underline{\text{Hom}}_\ell(a, b \otimes a)$, i.e., the unit of the tensor hom adjunction.

1.5.8 TensorProductToLeftInternalHomAdjunctMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ TensorProductToLeftInternalHomAdjunctMorphism(a, b, f) (operation)

Returns: a morphism in $\text{Hom}(a, \underline{\text{Hom}}_\ell(b, c))$.

The arguments are two objects a, b and a morphism $f : a \otimes b \rightarrow c$. The output is a morphism $g : a \rightarrow \underline{\text{Hom}}_\ell(b, c)$ corresponding to f under the tensor hom adjunction.

1.5.9 TensorProductToLeftInternalHomAdjunctMorphismWithGivenLeftInternalHom (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ TensorProductToLeftInternalHomAdjunctMorphismWithGivenLeftInternalHom(a, b, f, i) (operation)

Returns: a morphism in $\text{Hom}(a, i)$.

The arguments are two objects a, b , a morphism $f : a \otimes b \rightarrow c$ and an object $i = \underline{\text{Hom}}_\ell(b, c)$. The output is a morphism $g : a \rightarrow \underline{\text{Hom}}_\ell(b, c)$ corresponding to f under the tensor hom adjunction.

1.5.10 LeftInternalHomToTensorProductAdjunctMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ LeftInternalHomToTensorProductAdjunctMorphism(b, c, g) (operation)

Returns: a morphism in $\text{Hom}(a \otimes b, c)$.

The arguments are two objects b, c and a morphism $g : a \rightarrow \underline{\text{Hom}}_\ell(b, c)$. The output is a morphism $f : a \otimes b \rightarrow c$ corresponding to g under the tensor hom adjunction.

1.5.11 LeftInternalHomToTensorProductAdjunctMorphismWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ LeftInternalHomToTensorProductAdjunctMorphismWithGivenTensorProduct(b, c, g, t) (operation)

Returns: a morphism in $\text{Hom}(t, c)$.

The arguments are two objects b, c , a morphism $g : a \rightarrow \underline{\text{Hom}}_\ell(b, c)$ and an object $t = a \otimes b$. The output is a morphism $f : a \otimes b \rightarrow c$ corresponding to g under the tensor hom adjunction.

1.5.12 LeftClosedMonoidalPreComposeMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftClosedMonoidalPreComposeMorphism(a, b, c) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}_\ell(a, b) \otimes \underline{\text{Hom}}_\ell(b, c), \underline{\text{Hom}}_\ell(a, c))$.

The arguments are three objects a, b, c . The output is the precomposition morphism $\text{LeftClosedMonoidalPreComposeMorphism}_{a,b,c} : \underline{\text{Hom}}_\ell(a, b) \otimes \underline{\text{Hom}}_\ell(b, c) \rightarrow \underline{\text{Hom}}_\ell(a, c)$.

1.5.13 LeftClosedMonoidalPreComposeMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftClosedMonoidalPreComposeMorphismWithGivenObjects(s, a, b, c, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = \underline{\text{Hom}}_\ell(a, b) \otimes \underline{\text{Hom}}_\ell(b, c)$, three objects a, b, c , and an object $r = \underline{\text{Hom}}_\ell(a, c)$. The output is the precomposition morphism $\text{LeftClosedMonoidalPreComposeMorphismWithGivenObjects}_{a,b,c} : \underline{\text{Hom}}_\ell(a, b) \otimes \underline{\text{Hom}}_\ell(b, c) \rightarrow \underline{\text{Hom}}_\ell(a, c)$.

1.5.14 LeftClosedMonoidalPostComposeMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftClosedMonoidalPostComposeMorphism(a, b, c) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}_\ell(b, c) \otimes \underline{\text{Hom}}_\ell(a, b), \underline{\text{Hom}}_\ell(a, c))$.

The arguments are three objects a, b, c . The output is the postcomposition morphism $\text{LeftClosedMonoidalPostComposeMorphism}_{a,b,c} : \underline{\text{Hom}}_\ell(b, c) \otimes \underline{\text{Hom}}_\ell(a, b) \rightarrow \underline{\text{Hom}}_\ell(a, c)$.

1.5.15 LeftClosedMonoidalPostComposeMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftClosedMonoidalPostComposeMorphismWithGivenObjects(s, a, b, c, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = \underline{\text{Hom}}_\ell(b, c) \otimes \underline{\text{Hom}}_\ell(a, b)$, three objects a, b, c , and an object $r = \underline{\text{Hom}}_\ell(a, c)$. The output is the postcomposition morphism $\text{LeftClosedMonoidalPostComposeMorphismWithGivenObjects}_{a,b,c} : \underline{\text{Hom}}_\ell(b, c) \otimes \underline{\text{Hom}}_\ell(a, b) \rightarrow \underline{\text{Hom}}_\ell(a, c)$.

1.5.16 LeftDualOnObjects (for IsCapCategoryObject)

▷ LeftDualOnObjects(a) (attribute)

Returns: an object

The argument is an object a . The output is its dual object a^\vee .

1.5.17 LeftDualOnMorphisms (for IsCapCategoryMorphism)

▷ LeftDualOnMorphisms(α) (attribute)

Returns: a morphism in $\text{Hom}(b^\vee, a^\vee)$.

The argument is a morphism $\alpha : a \rightarrow b$. The output is its dual morphism $\alpha^\vee : b^\vee \rightarrow a^\vee$.

1.5.18 LeftDualOnMorphismsWithGivenLeftDuals (for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ LeftDualOnMorphismsWithGivenLeftDuals(s , α , r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The argument is an object $s = b^\vee$, a morphism $\alpha : a \rightarrow b$, and an object $r = a^\vee$. The output is the dual morphism $\alpha^\vee : b^\vee \rightarrow a^\vee$.

1.5.19 LeftClosedMonoidalEvaluationForLeftDual (for IsCapCategoryObject)

▷ LeftClosedMonoidalEvaluationForLeftDual(a) (attribute)

Returns: a morphism in $\text{Hom}(a^\vee \otimes a, 1)$.

The argument is an object a . The output is the evaluation morphism $\text{ev}_a : a^\vee \otimes a \rightarrow 1$.

1.5.20 LeftClosedMonoidalEvaluationForLeftDualWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftClosedMonoidalEvaluationForLeftDualWithGivenTensorProduct(s , a , r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = a^\vee \otimes a$, an object a , and an object $r = 1$. The output is the evaluation morphism $\text{ev}_a : a^\vee \otimes a \rightarrow 1$.

1.5.21 MorphismToLeftBidual (for IsCapCategoryObject)

▷ MorphismToLeftBidual(a) (attribute)

Returns: a morphism in $\text{Hom}(a, (a^\vee)^\vee)$.

The argument is an object a . The output is the morphism to the bidual $a \rightarrow (a^\vee)^\vee$.

1.5.22 MorphismToLeftBidualWithGivenLeftBidual (for IsCapCategoryObject, IsCapCategoryObject)

▷ MorphismToLeftBidualWithGivenLeftBidual(a , r) (operation)

Returns: a morphism in $\text{Hom}(a, r)$.

The arguments are an object a , and an object $r = (a^\vee)^\vee$. The output is the morphism to the bidual $a \rightarrow (a^\vee)^\vee$.

1.5.23 TensorProductLeftInternalHomCompatibilityMorphism (for IsList)

▷ TensorProductLeftInternalHomCompatibilityMorphism($list$) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}_\ell(a, a') \otimes \underline{\text{Hom}}_\ell(b, b'), \underline{\text{Hom}}_\ell(a \otimes b, a' \otimes b'))$.

The argument is a list of four objects $[a, a', b, b']$. The output is the natural morphism $\text{TensorProductLeftInternalHomCompatibilityMorphism}_{a, a', b, b'} : \underline{\text{Hom}}_\ell(a, a') \otimes \underline{\text{Hom}}_\ell(b, b') \rightarrow \underline{\text{Hom}}_\ell(a \otimes b, a' \otimes b')$.

1.5.24 TensorProductLeftInternalHomCompatibilityMorphismWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject)

▷ TensorProductLeftInternalHomCompatibilityMorphismWithGivenObjects(s , $list$, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are a list of four objects $[a, a', b, b']$, and two objects $s = \underline{\text{Hom}}_\ell(a, a') \otimes \underline{\text{Hom}}_\ell(b, b')$ and $r = \underline{\text{Hom}}_\ell(a \otimes b, a' \otimes b')$. The output is the natural morphism $\text{TensorProductLeftInternalHomCompatibilityMorphismWithGivenObjects}_{a, a', b, b'} : \underline{\text{Hom}}_\ell(a, a') \otimes \underline{\text{Hom}}_\ell(b, b') \rightarrow \underline{\text{Hom}}_\ell(a \otimes b, a' \otimes b')$.

1.5.25 TensorProductLeftDualityCompatibilityMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ TensorProductLeftDualityCompatibilityMorphism(a , b) (operation)

Returns: a morphism in $\text{Hom}(a^\vee \otimes b^\vee, (a \otimes b)^\vee)$.

The arguments are two objects a, b . The output is the natural morphism $\text{TensorProductLeftDualityCompatibilityMorphism} : a^\vee \otimes b^\vee \rightarrow (a \otimes b)^\vee$.

1.5.26 TensorProductLeftDualityCompatibilityMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ TensorProductLeftDualityCompatibilityMorphismWithGivenObjects(s , a , b , r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = a^\vee \otimes b^\vee$, two objects a, b , and an object $r = (a \otimes b)^\vee$. The output is the natural morphism $\text{TensorProductLeftDualityCompatibilityMorphismWithGivenObjects}_{a, b} : a^\vee \otimes b^\vee \rightarrow (a \otimes b)^\vee$.

1.5.27 MorphismFromTensorProductToLeftInternalHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ MorphismFromTensorProductToLeftInternalHom(a , b) (operation)

Returns: a morphism in $\text{Hom}(a^\vee \otimes b, \underline{\text{Hom}}_\ell(a, b))$.

The arguments are two objects a, b . The output is the natural morphism $\text{MorphismFromTensorProductToLeftInternalHom}_{a, b} : a^\vee \otimes b \rightarrow \underline{\text{Hom}}_\ell(a, b)$.

1.5.28 MorphismFromTensorProductToLeftInternalHomWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ MorphismFromTensorProductToLeftInternalHomWithGivenObjects(s , a , b , r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = a^\vee \otimes b$, two objects a, b , and an object $r = \underline{\text{Hom}}_\ell(a, b)$. The output is the natural morphism $\text{MorphismFromTensorProductToLeftInternalHomWithGivenObjects}_{a, b} : a^\vee \otimes b \rightarrow \underline{\text{Hom}}_\ell(a, b)$.

1.5.29 IsomorphismFromLeftDualObjectToLeftInternalHomIntoTensorUnit (for IsCapCategoryObject)

▷ `IsomorphismFromLeftDualObjectToLeftInternalHomIntoTensorUnit(a)` (attribute)

Returns: a morphism in $\text{Hom}(a^\vee, \underline{\text{Hom}}_\ell(a, 1))$.

The argument is an object a . The output is the isomorphism $\text{IsomorphismFromLeftDualObjectToLeftInternalHomIntoTensorUnit}_a : a^\vee \rightarrow \underline{\text{Hom}}_\ell(a, 1)$.

1.5.30 IsomorphismFromLeftInternalHomIntoTensorUnitToLeftDualObject (for IsCapCategoryObject)

▷ `IsomorphismFromLeftInternalHomIntoTensorUnitToLeftDualObject(a)` (attribute)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}_\ell(a, 1), a^\vee)$.

The argument is an object a . The output is the isomorphism $\text{IsomorphismFromLeftInternalHomIntoTensorUnitToLeftDualObject}_a : \underline{\text{Hom}}_\ell(a, 1) \rightarrow a^\vee$.

1.5.31 UniversalPropertyOfLeftDual (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ `UniversalPropertyOfLeftDual(t, a, alpha)` (operation)

Returns: a morphism in $\text{Hom}(t, a^\vee)$.

The arguments are two objects t, a , and a morphism $\alpha : t \otimes a \rightarrow 1$. The output is the morphism $t \rightarrow a^\vee$ given by the universal property of a^\vee .

1.5.32 LeftClosedMonoidalLambdaIntroduction (for IsCapCategoryMorphism)

▷ `LeftClosedMonoidalLambdaIntroduction(alpha)` (attribute)

Returns: a morphism in $\text{Hom}(1, \underline{\text{Hom}}_\ell(a, b))$.

The argument is a morphism $\alpha : a \rightarrow b$. The output is the corresponding morphism $1 \rightarrow \underline{\text{Hom}}_\ell(a, b)$ under the tensor hom adjunction.

1.5.33 LeftClosedMonoidalLambdaElimination (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ `LeftClosedMonoidalLambdaElimination(a, b, alpha)` (operation)

Returns: a morphism in $\text{Hom}(a, b)$.

The arguments are two objects a, b , and a morphism $\alpha : 1 \rightarrow \underline{\text{Hom}}_\ell(a, b)$. The output is a morphism $a \rightarrow b$ corresponding to α under the tensor hom adjunction.

1.5.34 IsomorphismFromObjectToLeftInternalHom (for IsCapCategoryObject)

▷ `IsomorphismFromObjectToLeftInternalHom(a)` (attribute)

Returns: a morphism in $\text{Hom}(a, \underline{\text{Hom}}_\ell(1, a))$.

The argument is an object a . The output is the natural isomorphism $a \rightarrow \underline{\text{Hom}}_\ell(1, a)$.

1.5.35 IsomorphismFromObjectToLeftInternalHomWithGivenLeftInternalHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ `IsomorphismFromObjectToLeftInternalHomWithGivenLeftInternalHom(a, r)` (operation)

Returns: a morphism in $\text{Hom}(a, r)$.

The argument is an object a , and an object $r = \underline{\text{Hom}}_\ell(1, a)$. The output is the natural isomorphism $a \rightarrow \underline{\text{Hom}}_\ell(1, a)$.

1.5.36 IsomorphismFromLeftInternalHomToObject (for IsCapCategoryObject)

▷ `IsomorphismFromLeftInternalHomToObject(a)` (attribute)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}_\ell(1, a), a)$.

The argument is an object a . The output is the natural isomorphism $\underline{\text{Hom}}_\ell(1, a) \rightarrow a$.

1.5.37 IsomorphismFromLeftInternalHomToObjectWithGivenLeftInternalHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ `IsomorphismFromLeftInternalHomToObjectWithGivenLeftInternalHom(a, s)` (operation)

Returns: a morphism in $\text{Hom}(s, a)$.

The argument is an object a , and an object $s = \underline{\text{Hom}}_\ell(1, a)$. The output is the natural isomorphism $\underline{\text{Hom}}_\ell(1, a) \rightarrow a$.

1.6 Closed Monoidal Categories

A monoidal category \mathbf{C} which has for each functor $- \otimes b : \mathbf{C} \rightarrow \mathbf{C}$ a right adjoint (denoted by $\underline{\text{Hom}}_\ell(b, -)$) is called a *closed monoidal category*.

If no operations involving duals are installed manually, the dual objects will be derived as $a^\vee := \underline{\text{Hom}}_\ell(a, 1)$.

The corresponding GAP property is called `IsClosedMonoidalCategory`.

1.6.1 InternalHomOnObjects (for IsCapCategoryObject, IsCapCategoryObject)

▷ `InternalHomOnObjects(a, b)` (operation)

Returns: an object

The arguments are two objects a, b . The output is the internal hom object $\underline{\text{Hom}}(a, b)$.

1.6.2 InternalHomOnMorphisms (for IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `InternalHomOnMorphisms(alpha, beta)` (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(a', b), \underline{\text{Hom}}(a, b'))$

The arguments are two morphisms $\alpha : a \rightarrow a', \beta : b \rightarrow b'$. The output is the internal hom morphism $\underline{\text{Hom}}(\alpha, \beta) : \underline{\text{Hom}}(a', b) \rightarrow \underline{\text{Hom}}(a, b')$.

1.6.3 InternalHomOnMorphismsWithGivenInternalHoms (for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryObject)

▷ InternalHomOnMorphismsWithGivenInternalHoms(s , α , β , r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$

The arguments are an object $s = \underline{\text{Hom}}(a', b)$, two morphisms $\alpha : a \rightarrow a'$, $\beta : b \rightarrow b'$, and an object $r = \underline{\text{Hom}}(a, b')$. The output is the internal hom morphism $\underline{\text{Hom}}(\alpha, \beta) : \underline{\text{Hom}}(a', b) \rightarrow \underline{\text{Hom}}(a, b')$.

1.6.4 ClosedMonoidalRightEvaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ ClosedMonoidalRightEvaluationMorphism(a , b) (operation)

Returns: a morphism in $\text{Hom}(a \otimes \underline{\text{Hom}}(a, b), b)$.

The arguments are two objects a, b . The output is the right evaluation morphism $\text{ev}_{a,b} : a \otimes \underline{\text{Hom}}(a, b) \rightarrow b$, i.e., the counit of the tensor hom adjunction.

1.6.5 ClosedMonoidalRightEvaluationMorphismWithGivenSource (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ ClosedMonoidalRightEvaluationMorphismWithGivenSource(a , b , s) (operation)

Returns: a morphism in $\text{Hom}(s, b)$.

The arguments are two objects a, b and an object $s = a \otimes \underline{\text{Hom}}(a, b)$. The output is the right evaluation morphism $\text{ev}_{a,b} : a \otimes \underline{\text{Hom}}(a, b) \rightarrow b$, i.e., the counit of the tensor hom adjunction.

1.6.6 ClosedMonoidalRightCoevaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ ClosedMonoidalRightCoevaluationMorphism(a , b) (operation)

Returns: a morphism in $\text{Hom}(b, \underline{\text{Hom}}(a, a \otimes b))$.

The arguments are two objects a, b . The output is the right coevaluation morphism $\text{coev}_{a,b} : b \rightarrow \underline{\text{Hom}}(a, a \otimes b)$, i.e., the unit of the tensor hom adjunction.

1.6.7 ClosedMonoidalRightCoevaluationMorphismWithGivenRange (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ ClosedMonoidalRightCoevaluationMorphismWithGivenRange(a , b , r) (operation)

Returns: a morphism in $\text{Hom}(b, r)$.

The arguments are two objects a, b and an object $r = \underline{\text{Hom}}(a, a \otimes b)$. The output is the right coevaluation morphism $\text{coev}_{a,b} : b \rightarrow \underline{\text{Hom}}(a, a \otimes b)$, i.e., the unit of the tensor hom adjunction.

1.6.8 TensorProductToInternalHomRightAdjunctMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ TensorProductToInternalHomRightAdjunctMorphism(a , b , f) (operation)

Returns: a morphism in $\text{Hom}(b, \underline{\text{Hom}}(a, c))$.

The arguments are two objects a, b and a morphism $f : a \otimes b \rightarrow c$. The output is a morphism $g : b \rightarrow \underline{\text{Hom}}(a, c)$ corresponding to f under the tensor hom adjunction.

1.6.9 TensorProductToInternalHomRightAdjunctMorphismWithGivenInternalHom (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ `TensorProductToInternalHomRightAdjunctMorphismWithGivenInternalHom(a, b, f, i)` (operation)

Returns: a morphism in $\text{Hom}(b, i)$.

The arguments are two objects a, b , a morphism $f : a \otimes b \rightarrow c$ and an object $i = \underline{\text{Hom}}(a, c)$. The output is a morphism $g : b \rightarrow i$ corresponding to f under the tensor hom adjunction.

1.6.10 TensorProductToInternalHomRightAdjunctionIsomorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ `TensorProductToInternalHomRightAdjunctionIsomorphism(a, b, c)` (operation)

Returns: a morphism in $\text{Hom}(H(a \otimes b, c), H(b, \underline{\text{Hom}}(a, c)))$.

The arguments are three objects a, b, c . The output is the tri-natural isomorphism $H(a \otimes b, c) \rightarrow H(b, \underline{\text{Hom}}(a, c))$ in the range category of the homomorphism structure H .

1.6.11 TensorProductToInternalHomRightAdjunctionIsomorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ `TensorProductToInternalHomRightAdjunctionIsomorphismWithGivenObjects(s, a, b, c, r)` (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are five objects s, a, b, c, r where $s = H(a \otimes b, c)$ and $r = H(b, \underline{\text{Hom}}(a, c))$. The output is the tri-natural isomorphism $s \rightarrow r$ in the range category of the homomorphism structure H .

1.6.12 InternalHomToTensorProductRightAdjunctMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ `InternalHomToTensorProductRightAdjunctMorphism(a, c, g)` (operation)

Returns: a morphism in $\text{Hom}(a \otimes b, c)$.

The arguments are two objects a, c and a morphism $g : b \rightarrow \underline{\text{Hom}}(a, c)$. The output is a morphism $f : a \otimes b \rightarrow c$ corresponding to g under the tensor hom adjunction.

1.6.13 InternalHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ `InternalHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct(a, c, g, s)` (operation)

Returns: a morphism in $\text{Hom}(s, c)$.

The arguments are two objects a, c , a morphism $g : b \rightarrow \underline{\text{Hom}}(a, c)$ and an object $s = a \otimes b$. The output is a morphism $f : s \rightarrow c$ corresponding to g under the tensor hom adjunction.

1.6.14 InternalHomToTensorProductRightAdjunctionIsomorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ InternalHomToTensorProductRightAdjunctionIsomorphism(a, b, c) (operation)

Returns: a morphism in $\text{Hom}(H(b, \underline{\text{Hom}}(a, c)), H(a \otimes b, c))$.

The arguments are three objects a, b, c . The output is the tri-natural isomorphism $H(b, \underline{\text{Hom}}(a, c)) \rightarrow H(a \otimes b, c)$ in the range category of the homomorphism structure H .

1.6.15 InternalHomToTensorProductRightAdjunctionIsomorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ InternalHomToTensorProductRightAdjunctionIsomorphismWithGivenObjects(s, a, b, c, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are five objects s, a, b, c, r where $s = H(b, \underline{\text{Hom}}(a, c))$ and $r = H(a \otimes b, c)$. The output is the tri-natural isomorphism $s \rightarrow r$ in the range category of the homomorphism structure H .

1.6.16 ClosedMonoidalLeftEvaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ ClosedMonoidalLeftEvaluationMorphism(a, b) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(a, b) \otimes a, b)$.

The arguments are two objects a, b . The output is the left evaluation morphism $\text{ev}_{a,b} : \underline{\text{Hom}}(a, b) \otimes a \rightarrow b$, i.e., the counit of the tensor hom adjunction.

1.6.17 ClosedMonoidalLeftEvaluationMorphismWithGivenSource (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ ClosedMonoidalLeftEvaluationMorphismWithGivenSource(a, b, s) (operation)

Returns: a morphism in $\text{Hom}(s, b)$.

The arguments are two objects a, b and an object $s = \underline{\text{Hom}}(a, b) \otimes a$. The output is the left evaluation morphism $\text{ev}_{a,b} : \underline{\text{Hom}}(a, b) \otimes a \rightarrow b$, i.e., the counit of the tensor hom adjunction.

1.6.18 ClosedMonoidalLeftCoevaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ ClosedMonoidalLeftCoevaluationMorphism(a, b) (operation)

Returns: a morphism in $\text{Hom}(b, \underline{\text{Hom}}(a, b \otimes a))$.

The arguments are two objects a, b . The output is the left coevaluation morphism $\text{coev}_{a,b} : b \rightarrow \underline{\text{Hom}}(a, b \otimes a)$, i.e., the unit of the tensor hom adjunction.

1.6.19 ClosedMonoidalLeftCoevaluationMorphismWithGivenRange (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ ClosedMonoidalLeftCoevaluationMorphismWithGivenRange(a, b, r) (operation)

Returns: a morphism in $\text{Hom}(b, r)$.

The arguments are two objects a, b and an object $r = \underline{\text{Hom}}(a, b \otimes a)$. The output is the left coevaluation morphism $\text{coev}_{a,b} : b \rightarrow \underline{\text{Hom}}(a, b \otimes a)$, i.e., the unit of the tensor hom adjunction.

1.6.20 **TensorProductToInternalHomLeftAdjunctMorphism** (for **IsCapCategoryObject**, **IsCapCategoryObject**, **IsCapCategoryMorphism**)

▷ **TensorProductToInternalHomLeftAdjunctMorphism**(a, b, f) (operation)

Returns: a morphism in $\text{Hom}(a, \underline{\text{Hom}}(b, c))$.

The arguments are two objects a, b and a morphism $f : a \otimes b \rightarrow c$. The output is a morphism $g : a \rightarrow \underline{\text{Hom}}(b, c)$ corresponding to f under the tensor hom adjunction.

1.6.21 **TensorProductToInternalHomLeftAdjunctMorphismWithGivenInternalHom** (for **IsCapCategoryObject**, **IsCapCategoryObject**, **IsCapCategoryMorphism**, **IsCapCategoryObject**)

▷ **TensorProductToInternalHomLeftAdjunctMorphismWithGivenInternalHom**(a, b, f, i) (operation)

Returns: a morphism in $\text{Hom}(a, i)$.

The arguments are two objects a, b , a morphism $f : a \otimes b \rightarrow c$ and an object $i = \underline{\text{Hom}}(b, c)$. The output is a morphism $g : a \rightarrow i$ corresponding to f under the tensor hom adjunction.

1.6.22 **TensorProductToInternalHomLeftAdjunctionIsomorphism** (for **IsCapCategoryObject**, **IsCapCategoryObject**, **IsCapCategoryObject**)

▷ **TensorProductToInternalHomLeftAdjunctionIsomorphism**(a, b, c) (operation)

Returns: a morphism in $\text{Hom}(H(a \otimes b, c), H(a, \underline{\text{Hom}}(b, c)))$.

The arguments are three objects a, b, c . The output is the tri-natural isomorphism $H(a \otimes b, c) \rightarrow H(a, \underline{\text{Hom}}(b, c))$ in the range category of the homomorphism structure H .

1.6.23 **TensorProductToInternalHomLeftAdjunctionIsomorphismWithGivenObjects** (for **IsCapCategoryObject**, **IsCapCategoryObject**, **IsCapCategoryObject**, **IsCapCategoryObject**, **IsCapCategoryObject**)

▷ **TensorProductToInternalHomLeftAdjunctionIsomorphismWithGivenObjects**(s, a, b, c, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are five objects s, a, b, c, r where $s = H(a \otimes b, c)$ and $r = H(a, \underline{\text{Hom}}(b, c))$. The output is the tri-natural isomorphism $s \rightarrow r$ in the range category of the homomorphism structure H .

1.6.24 **InternalHomToTensorProductLeftAdjunctMorphism** (for **IsCapCategoryObject**, **IsCapCategoryObject**, **IsCapCategoryMorphism**)

▷ **InternalHomToTensorProductLeftAdjunctMorphism**(b, c, g) (operation)

Returns: a morphism in $\text{Hom}(a \otimes b, c)$.

The arguments are two objects b, c and a morphism $g : a \rightarrow \underline{\text{Hom}}(b, c)$. The output is a morphism $f : a \otimes b \rightarrow c$ corresponding to g under the tensor hom adjunction.

1.6.25 InternalHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ InternalHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct(b, c, g, s) (operation)

Returns: a morphism in $\text{Hom}(s, c)$.

The arguments are two objects b, c , a morphism $g : a \rightarrow \underline{\text{Hom}}(b, c)$ and an object $s = a \otimes b$. The output is a morphism $f : s \rightarrow c$ corresponding to g under the tensor hom adjunction.

1.6.26 InternalHomToTensorProductLeftAdjunctionIsomorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ InternalHomToTensorProductLeftAdjunctionIsomorphism(a, b, c) (operation)

Returns: a morphism in $\text{Hom}(H(a, \underline{\text{Hom}}(b, c)), H(a \otimes b, c))$.

The arguments are three objects a, b, c . The output is the tri-natural isomorphism $H(a, \underline{\text{Hom}}(b, c)) \rightarrow H(a \otimes b, c)$ in the range category of the homomorphism structure H .

1.6.27 InternalHomToTensorProductLeftAdjunctionIsomorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ InternalHomToTensorProductLeftAdjunctionIsomorphismWithGivenObjects(s, a, b, c, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are five objects s, a, b, c, r where $s = H(a, \underline{\text{Hom}}(b, c))$ and $r = H(a \otimes b, c)$. The output is the tri-natural isomorphism $s \rightarrow r$ in the range category of the homomorphism structure H .

1.6.28 MonoidalPreComposeMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ MonoidalPreComposeMorphism(a, b, c) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(a, b) \otimes \underline{\text{Hom}}(b, c), \underline{\text{Hom}}(a, c))$.

The arguments are three objects a, b, c . The output is the precomposition morphism $\text{MonoidalPreComposeMorphism}_{a,b,c} : \underline{\text{Hom}}(a, b) \otimes \underline{\text{Hom}}(b, c) \rightarrow \underline{\text{Hom}}(a, c)$.

1.6.29 MonoidalPreComposeMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ MonoidalPreComposeMorphismWithGivenObjects(s, a, b, c, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = \underline{\text{Hom}}(a, b) \otimes \underline{\text{Hom}}(b, c)$, three objects a, b, c , and an object $r = \underline{\text{Hom}}(a, c)$. The output is the precomposition morphism $\text{MonoidalPreComposeMorphismWithGivenObjects}_{a,b,c} : \underline{\text{Hom}}(a, b) \otimes \underline{\text{Hom}}(b, c) \rightarrow \underline{\text{Hom}}(a, c)$.

1.6.30 MonoidalPostComposeMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ MonoidalPostComposeMorphism(a, b, c) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(b, c) \otimes \underline{\text{Hom}}(a, b), \underline{\text{Hom}}(a, c))$.

The arguments are three objects a, b, c . The output is the postcomposition morphism $\text{MonoidalPostComposeMorphism}_{a,b,c} : \underline{\text{Hom}}(b, c) \otimes \underline{\text{Hom}}(a, b) \rightarrow \underline{\text{Hom}}(a, c)$.

1.6.31 MonoidalPostComposeMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ MonoidalPostComposeMorphismWithGivenObjects(s, a, b, c, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = \underline{\text{Hom}}(b, c) \otimes \underline{\text{Hom}}(a, b)$, three objects a, b, c , and an object $r = \underline{\text{Hom}}(a, c)$. The output is the postcomposition morphism $\text{MonoidalPostComposeMorphismWithGivenObjects}_{a,b,c} : \underline{\text{Hom}}(b, c) \otimes \underline{\text{Hom}}(a, b) \rightarrow \underline{\text{Hom}}(a, c)$.

1.6.32 DualOnObjects (for IsCapCategoryObject)

▷ DualOnObjects(a) (attribute)

Returns: an object

The argument is an object a . The output is its dual object a^\vee .

1.6.33 DualOnMorphisms (for IsCapCategoryMorphism)

▷ DualOnMorphisms(alpha) (attribute)

Returns: a morphism in $\text{Hom}(b^\vee, a^\vee)$.

The argument is a morphism $\alpha : a \rightarrow b$. The output is its dual morphism $\alpha^\vee : b^\vee \rightarrow a^\vee$.

1.6.34 DualOnMorphismsWithGivenDuals (for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ DualOnMorphismsWithGivenDuals(s, alpha, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The argument is an object $s = b^\vee$, a morphism $\alpha : a \rightarrow b$, and an object $r = a^\vee$. The output is the dual morphism $\alpha^\vee : b^\vee \rightarrow a^\vee$.

1.6.35 EvaluationForDual (for IsCapCategoryObject)

▷ EvaluationForDual(a) (attribute)

Returns: a morphism in $\text{Hom}(a^\vee \otimes a, 1)$.

The argument is an object a . The output is the evaluation morphism $\text{ev}_a : a^\vee \otimes a \rightarrow 1$.

1.6.36 EvaluationForDualWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ EvaluationForDualWithGivenTensorProduct(s, a, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = a^\vee \otimes a$, an object a , and an object $r = 1$. The output is the evaluation morphism $\text{ev}_a : a^\vee \otimes a \rightarrow 1$.

1.6.37 MorphismToBidual (for IsCapCategoryObject)

- ▷ `MorphismToBidual(a)` (attribute)
Returns: a morphism in $\text{Hom}(a, (a^\vee)^\vee)$.
 The argument is an object a . The output is the morphism to the bidual $a \rightarrow (a^\vee)^\vee$.

1.6.38 MorphismToBidualWithGivenBidual (for IsCapCategoryObject, IsCapCategoryObject)

- ▷ `MorphismToBidualWithGivenBidual(a, r)` (operation)
Returns: a morphism in $\text{Hom}(a, r)$.
 The arguments are an object a , and an object $r = (a^\vee)^\vee$. The output is the morphism to the bidual $a \rightarrow (a^\vee)^\vee$.

1.6.39 TensorProductInternalHomCompatibilityMorphism (for IsList)

- ▷ `TensorProductInternalHomCompatibilityMorphism(list)` (operation)
Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(a, a') \otimes \underline{\text{Hom}}(b, b'), \underline{\text{Hom}}(a \otimes b, a' \otimes b'))$.
 The argument is a list of four objects $[a, a', b, b']$. The output is the natural morphism $\text{TensorProductInternalHomCompatibilityMorphism}_{a, a', b, b'} : \underline{\text{Hom}}(a, a') \otimes \underline{\text{Hom}}(b, b') \rightarrow \underline{\text{Hom}}(a \otimes b, a' \otimes b')$.

1.6.40 TensorProductInternalHomCompatibilityMorphismWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject)

- ▷ `TensorProductInternalHomCompatibilityMorphismWithGivenObjects(s, list, r)` (operation)
Returns: a morphism in $\text{Hom}(s, r)$.
 The arguments are a list of four objects $[a, a', b, b']$, and two objects $s = \underline{\text{Hom}}(a, a') \otimes \underline{\text{Hom}}(b, b')$ and $r = \underline{\text{Hom}}(a \otimes b, a' \otimes b')$. The output is the natural morphism $\text{TensorProductInternalHomCompatibilityMorphismWithGivenObjects}_{a, a', b, b'} : \underline{\text{Hom}}(a, a') \otimes \underline{\text{Hom}}(b, b') \rightarrow \underline{\text{Hom}}(a \otimes b, a' \otimes b')$.

1.6.41 TensorProductDualityCompatibilityMorphism (for IsCapCategoryObject, IsCapCategoryObject)

- ▷ `TensorProductDualityCompatibilityMorphism(a, b)` (operation)
Returns: a morphism in $\text{Hom}(a^\vee \otimes b^\vee, (a \otimes b)^\vee)$.
 The arguments are two objects a, b . The output is the natural morphism $\text{TensorProductDualityCompatibilityMorphism} : a^\vee \otimes b^\vee \rightarrow (a \otimes b)^\vee$.

1.6.42 TensorProductDualityCompatibilityMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ TensorProductDualityCompatibilityMorphismWithGivenObjects(s, a, b, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = a^\vee \otimes b^\vee$, two objects a, b , and an object $r = (a \otimes b)^\vee$. The output is the natural morphism $\text{TensorProductDualityCompatibilityMorphismWithGivenObjects}_{a,b} : a^\vee \otimes b^\vee \rightarrow (a \otimes b)^\vee$.

1.6.43 MorphismFromTensorProductToInternalHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ MorphismFromTensorProductToInternalHom(a, b) (operation)

Returns: a morphism in $\text{Hom}(a^\vee \otimes b, \underline{\text{Hom}}(a, b))$.

The arguments are two objects a, b . The output is the natural morphism $\text{MorphismFromTensorProductToInternalHom}_{a,b} : a^\vee \otimes b \rightarrow \underline{\text{Hom}}(a, b)$.

1.6.44 MorphismFromTensorProductToInternalHomWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ MorphismFromTensorProductToInternalHomWithGivenObjects(s, a, b, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = a^\vee \otimes b$, two objects a, b , and an object $r = \underline{\text{Hom}}(a, b)$. The output is the natural morphism $\text{MorphismFromTensorProductToInternalHomWithGivenObjects}_{a,b} : a^\vee \otimes b \rightarrow \underline{\text{Hom}}(a, b)$.

1.6.45 IsomorphismFromDualObjectToInternalHomIntoTensorUnit (for IsCapCategoryObject)

▷ IsomorphismFromDualObjectToInternalHomIntoTensorUnit(a) (attribute)

Returns: a morphism in $\text{Hom}(a^\vee, \underline{\text{Hom}}(a, 1))$.

The argument is an object a . The output is the isomorphism $\text{IsomorphismFromDualObjectToInternalHomIntoTensorUnit}_a : a^\vee \rightarrow \underline{\text{Hom}}(a, 1)$.

1.6.46 IsomorphismFromInternalHomIntoTensorUnitToDualObject (for IsCapCategoryObject)

▷ IsomorphismFromInternalHomIntoTensorUnitToDualObject(a) (attribute)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(a, 1), a^\vee)$.

The argument is an object a . The output is the isomorphism $\text{IsomorphismFromInternalHomIntoTensorUnitToDualObject}_a : \underline{\text{Hom}}(a, 1) \rightarrow a^\vee$.

1.6.47 UniversalPropertyOfDual (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ UniversalPropertyOfDual(t , a , α) (operation)

Returns: a morphism in $\text{Hom}(t, a^\vee)$.

The arguments are two objects t, a , and a morphism $\alpha : t \otimes a \rightarrow 1$. The output is the morphism $t \rightarrow a^\vee$ given by the universal property of a^\vee .

1.6.48 LambdaIntroduction (for IsCapCategoryMorphism)

▷ LambdaIntroduction(α) (attribute)

Returns: a morphism in $\text{Hom}(1, \underline{\text{Hom}}(a, b))$.

The argument is a morphism $\alpha : a \rightarrow b$. The output is the corresponding morphism $1 \rightarrow \underline{\text{Hom}}(a, b)$ under the tensor hom adjunction.

1.6.49 LambdaElimination (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ LambdaElimination(a , b , α) (operation)

Returns: a morphism in $\text{Hom}(a, b)$.

The arguments are two objects a, b , and a morphism $\alpha : 1 \rightarrow \underline{\text{Hom}}(a, b)$. The output is a morphism $a \rightarrow b$ corresponding to α under the tensor hom adjunction.

1.6.50 IsomorphismFromObjectToInternalHom (for IsCapCategoryObject)

▷ IsomorphismFromObjectToInternalHom(a) (attribute)

Returns: a morphism in $\text{Hom}(a, \underline{\text{Hom}}(1, a))$.

The argument is an object a . The output is the natural isomorphism $a \rightarrow \underline{\text{Hom}}(1, a)$.

1.6.51 IsomorphismFromObjectToInternalHomWithGivenInternalHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ IsomorphismFromObjectToInternalHomWithGivenInternalHom(a , r) (operation)

Returns: a morphism in $\text{Hom}(a, r)$.

The argument is an object a , and an object $r = \underline{\text{Hom}}(1, a)$. The output is the natural isomorphism $a \rightarrow \underline{\text{Hom}}(1, a)$.

1.6.52 IsomorphismFromInternalHomToObject (for IsCapCategoryObject)

▷ IsomorphismFromInternalHomToObject(a) (attribute)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(1, a), a)$.

The argument is an object a . The output is the natural isomorphism $\underline{\text{Hom}}(1, a) \rightarrow a$.

1.6.53 IsomorphismFromInternalHomToObjectWithGivenInternalHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ IsomorphismFromInternalHomToObjectWithGivenInternalHom(a , s) (operation)

Returns: a morphism in $\text{Hom}(s, a)$.

The argument is an object a , and an object $s = \underline{\text{Hom}}(1, a)$. The output is the natural isomorphism $\underline{\text{Hom}}(1, a) \rightarrow a$.

1.7 Left Coclosed Monoidal Categories

A monoidal category \mathbf{C} which has for each functor $- \otimes b : \mathbf{C} \rightarrow \mathbf{C}$ a left adjoint (denoted by $\underline{\text{coHom}}(-, b)$) is called a *left coclosed monoidal category*.

If no operations involving left coduals are installed manually, the left codual objects will be derived as $a_\vee := \underline{\text{coHom}}(1, a)$.

The corresponding GAP property is called `IsLeftCoclosedMonoidalCategory`.

1.7.1 LeftInternalCoHomOnObjects (for IsCapCategoryObject, IsCapCategoryObject)

▷ `LeftInternalCoHomOnObjects(a, b)` (operation)

Returns: an object

The arguments are two objects a, b . The output is the internal cohom object $\underline{\text{coHom}}_\ell(a, b)$.

1.7.2 LeftInternalCoHomOnMorphisms (for IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `LeftInternalCoHomOnMorphisms(alpha, beta)` (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}_\ell(a, b'), \underline{\text{coHom}}_\ell(a', b))$

The arguments are two morphisms $\alpha : a \rightarrow a', \beta : b \rightarrow b'$. The output is the internal cohom morphism $\underline{\text{coHom}}_\ell(\alpha, \beta) : \underline{\text{coHom}}_\ell(a, b') \rightarrow \underline{\text{coHom}}_\ell(a', b)$.

1.7.3 LeftInternalCoHomOnMorphismsWithGivenLeftInternalCoHoms (for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryObject)

▷ `LeftInternalCoHomOnMorphismsWithGivenLeftInternalCoHoms(s, alpha, beta, r)` (operation)

Returns: a morphism in $\text{Hom}(s, r)$

The arguments are an object $s = \underline{\text{coHom}}_\ell(a, b')$, two morphisms $\alpha : a \rightarrow a', \beta : b \rightarrow b'$, and an object $r = \underline{\text{coHom}}_\ell(a', b)$. The output is the internal cohom morphism $\underline{\text{coHom}}_\ell(\alpha, \beta) : \underline{\text{coHom}}_\ell(a, b') \rightarrow \underline{\text{coHom}}_\ell(a', b)$.

1.7.4 LeftCoclosedMonoidalEvaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ `LeftCoclosedMonoidalEvaluationMorphism(a, b)` (operation)

Returns: a morphism in $\text{Hom}(b, \underline{\text{coHom}}_\ell(b, a) \otimes a)$.

The arguments are two objects a, b . The output is the coclosed evaluation morphism $\text{coclev}_{a,b} : b \rightarrow \underline{\text{coHom}}_\ell(b, a) \otimes a$, i.e., the unit of the cohom tensor adjunction.

1.7.5 LeftCoclosedMonoidalEvaluationMorphismWithGivenRange (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftCoclosedMonoidalEvaluationMorphismWithGivenRange(a, b, r) (operation)

Returns: a morphism in $\text{Hom}(b, r)$.

The arguments are two objects a, b and an object $r = \text{coHom}_\ell(b, a) \otimes a$. The output is the coclosed evaluation morphism $\text{coclev}_{a,b} : b \rightarrow \text{coHom}_\ell(b, a) \otimes a$, i.e., the unit of the cohom tensor adjunction.

1.7.6 LeftCoclosedMonoidalCoevaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ LeftCoclosedMonoidalCoevaluationMorphism(a, b) (operation)

Returns: a morphism in $\text{Hom}(\text{coHom}_\ell(b \otimes a, a), b)$.

The arguments are two objects a, b . The output is the coclosed coevaluation morphism $\text{cocleov}_{a,b} : \text{coHom}_\ell(b \otimes a, a) \rightarrow b$, i.e., the counit of the cohom tensor adjunction.

1.7.7 LeftCoclosedMonoidalCoevaluationMorphismWithGivenSource (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftCoclosedMonoidalCoevaluationMorphismWithGivenSource(a, b, s) (operation)

Returns: a morphism in $\text{Hom}(s, b)$.

The arguments are two objects a, b and an object $s = \text{coHom}_\ell(b \otimes a, a)$. The output is the coclosed coevaluation morphism $\text{cocleov}_{a,b} : \text{coHom}_\ell(b \otimes a, a) \rightarrow b$, i.e., the unit of the cohom tensor adjunction.

1.7.8 TensorProductToLeftInternalCoHomAdjunctMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ TensorProductToLeftInternalCoHomAdjunctMorphism(b, c, g) (operation)

Returns: a morphism in $\text{Hom}(\text{coHom}_\ell(a, c), b)$.

The arguments are two objects b, c and a morphism $g : a \rightarrow b \otimes c$. The output is a morphism $f : \text{coHom}_\ell(a, c) \rightarrow b$ corresponding to g under the cohom tensor adjunction.

1.7.9 TensorProductToLeftInternalCoHomAdjunctMorphismWithGivenLeftInternalCoHom (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ TensorProductToLeftInternalCoHomAdjunctMorphismWithGivenLeftInternalCoHom(b, c, g, i) (operation)

Returns: a morphism in $\text{Hom}(i, b)$.

The arguments are two objects b, c , a morphism $g : a \rightarrow b \otimes c$ and an object $i = \text{coHom}_\ell(a, c)$. The output is a morphism $f : \text{coHom}_\ell(a, c) \rightarrow b$ corresponding to g under the cohom tensor adjunction.

1.7.10 LeftInternalCoHomToTensorProductAdjunctMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ LeftInternalCoHomToTensorProductAdjunctMorphism(a, c, f) (operation)

Returns: a morphism in $\text{Hom}(a, b \otimes c)$.

The arguments are two objects a, c and a morphism $f : \underline{\text{coHom}}_\ell(a, c) \rightarrow b$. The output is a morphism $g : a \rightarrow b \otimes c$ corresponding to f under the cohom tensor adjunction.

1.7.11 LeftInternalCoHomToTensorProductAdjunctMorphismWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ LeftInternalCoHomToTensorProductAdjunctMorphismWithGivenTensorProduct(a, c, f, t) (operation)

Returns: a morphism in $\text{Hom}(a, t)$.

The arguments are two objects a, c , a morphism $f : \underline{\text{coHom}}_\ell(a, c) \rightarrow b$ and an object $t = b \otimes c$. The output is a morphism $g : a \rightarrow b \otimes c$ corresponding to f under the cohom tensor adjunction.

1.7.12 LeftCoclosedMonoidalPreCoComposeMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftCoclosedMonoidalPreCoComposeMorphism(a, b, c) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}_\ell(a, c), \underline{\text{coHom}}_\ell(b, c) \otimes \underline{\text{coHom}}_\ell(a, b))$.

The arguments are three objects a, b, c . The output is the precocomposition morphism $\text{LeftCoclosedMonoidalPreCoComposeMorphism}_{a,b,c} : \underline{\text{coHom}}_\ell(a, c) \rightarrow \underline{\text{coHom}}_\ell(b, c) \otimes \underline{\text{coHom}}_\ell(a, b)$.

1.7.13 LeftCoclosedMonoidalPreCoComposeMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftCoclosedMonoidalPreCoComposeMorphismWithGivenObjects(s, a, b, c, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = \underline{\text{coHom}}_\ell(a, c)$, three objects a, b, c , and an object $r = \underline{\text{coHom}}_\ell(a, b) \otimes \underline{\text{coHom}}_\ell(b, c)$. The output is the precocomposition morphism $\text{LeftCoclosedMonoidalPreCoComposeMorphismWithGivenObjects}_{a,b,c} : \underline{\text{coHom}}_\ell(a, c) \rightarrow \underline{\text{coHom}}_\ell(b, c) \otimes \underline{\text{coHom}}_\ell(a, b)$.

1.7.14 LeftCoclosedMonoidalPostCoComposeMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftCoclosedMonoidalPostCoComposeMorphism(a, b, c) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}_\ell(a, c), \underline{\text{coHom}}_\ell(a, b) \otimes \underline{\text{coHom}}_\ell(b, c))$.

The arguments are three objects a, b, c . The output is the postcocomposition morphism $\text{LeftCoclosedMonoidalPostCoComposeMorphism}_{a,b,c} : \underline{\text{coHom}}_\ell(a, c) \rightarrow \underline{\text{coHom}}_\ell(a, b) \otimes \underline{\text{coHom}}_\ell(b, c)$.

1.7.15 LeftCoclosedMonoidalPostCoComposeMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftCoclosedMonoidalPostCoComposeMorphismWithGivenObjects(s, a, b, c, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = \text{coHom}_\ell(a, c)$, three objects a, b, c , and an object $r = \text{coHom}_\ell(b, c) \otimes \text{coHom}_\ell(a, b)$. The output is the postcomposition morphism $\text{LeftCoclosedMonoidalPostCoComposeMorphismWithGivenObjects}_{a,b,c} : \text{coHom}_\ell(a, c) \rightarrow \text{coHom}_\ell(a, b) \otimes \text{coHom}_\ell(b, c)$.

1.7.16 LeftCoDualOnObjects (for IsCapCategoryObject)

▷ LeftCoDualOnObjects(a) (attribute)

Returns: an object

The argument is an object a . The output is its codual object a_\vee .

1.7.17 LeftCoDualOnMorphisms (for IsCapCategoryMorphism)

▷ LeftCoDualOnMorphisms(α) (attribute)

Returns: a morphism in $\text{Hom}(b_\vee, a_\vee)$.

The argument is a morphism $\alpha : a \rightarrow b$. The output is its codual morphism $\alpha_\vee : b_\vee \rightarrow a_\vee$.

1.7.18 LeftCoDualOnMorphismsWithGivenLeftCoDuals (for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ LeftCoDualOnMorphismsWithGivenLeftCoDuals(s, α, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The argument is an object $s = b_\vee$, a morphism $\alpha : a \rightarrow b$, and an object $r = a_\vee$. The output is the dual morphism $\alpha_\vee : b^\vee \rightarrow a^\vee$.

1.7.19 LeftCoclosedMonoidalEvaluationForLeftCoDual (for IsCapCategoryObject)

▷ LeftCoclosedMonoidalEvaluationForLeftCoDual(a) (attribute)

Returns: a morphism in $\text{Hom}(1, a_\vee \otimes a)$.

The argument is an object a . The output is the coclosed evaluation morphism $\text{coclev}_a : 1 \rightarrow a_\vee \otimes a$.

1.7.20 LeftCoclosedMonoidalEvaluationForLeftCoDualWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ LeftCoclosedMonoidalEvaluationForLeftCoDualWithGivenTensorProduct(s, a, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = 1$, an object a , and an object $r = a_\vee \otimes a$. The output is the coclosed evaluation morphism $\text{coclev}_a : 1 \rightarrow a_\vee \otimes a$.

1.7.21 MorphismFromLeftCoBidual (for IsCapCategoryObject)

▷ `MorphismFromLeftCoBidual(a)` (attribute)

Returns: a morphism in $\text{Hom}((a_V)_V, a)$.

The argument is an object a . The output is the morphism from the cobidual $(a_V)_V \rightarrow a$.

1.7.22 MorphismFromLeftCoBidualWithGivenLeftCoBidual (for IsCapCategoryObject, IsCapCategoryObject)

▷ `MorphismFromLeftCoBidualWithGivenLeftCoBidual(a, s)` (operation)

Returns: a morphism in $\text{Hom}(s, a)$.

The arguments are an object a , and an object $s = (a_V)_V$. The output is the morphism from the cobidual $(a_V)_V \rightarrow a$.

1.7.23 LeftInternalCoHomTensorProductCompatibilityMorphism (for IsList)

▷ `LeftInternalCoHomTensorProductCompatibilityMorphism(list)` (operation)

Returns: a morphism in $\text{Hom}(\text{coHom}_\ell(a \otimes a', b \otimes b'), \text{coHom}_\ell(a, b) \otimes \text{coHom}_\ell(a', b'))$.

The argument is a list of four objects $[a, a', b, b']$. The output is the natural morphism $\text{LeftInternalCoHomTensorProductCompatibilityMorphism}_{a, a', b, b'} : \text{coHom}_\ell(a \otimes a', b \otimes b') \rightarrow \text{coHom}_\ell(a, b) \otimes \text{coHom}_\ell(a', b')$.

1.7.24 LeftInternalCoHomTensorProductCompatibilityMorphismWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject)

▷ `LeftInternalCoHomTensorProductCompatibilityMorphismWithGivenObjects(s, list, r)` (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are a list of four objects $[a, a', b, b']$, and two objects $s = \text{coHom}_\ell(a \otimes a', b \otimes b')$ and $r = \text{coHom}_\ell(a, b) \otimes \text{coHom}_\ell(a', b')$. The output is the natural morphism $\text{LeftInternalCoHomTensorProductCompatibilityMorphismWithGivenObjects}_{a, a', b, b'} : \text{coHom}_\ell(a \otimes a', b \otimes b') \rightarrow \text{coHom}_\ell(a, b) \otimes \text{coHom}_\ell(a', b')$.

1.7.25 LeftCoDualityTensorProductCompatibilityMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ `LeftCoDualityTensorProductCompatibilityMorphism(a, b)` (operation)

Returns: a morphism in $\text{Hom}((a \otimes b)_V, a_V \otimes b_V)$.

The arguments are two objects a, b . The output is the natural morphism $\text{LeftCoDualityTensorProductCompatibilityMorphism} : (a \otimes b)_V \rightarrow a_V \otimes b_V$.

1.7.26 LeftCoDualityTensorProductCompatibilityMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ `LeftCoDualityTensorProductCompatibilityMorphismWithGivenObjects(s, a, b, r)` (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = (a \otimes b)_V$, two objects a, b , and an object $r = a_V \otimes b_V$. The output is the natural morphism $\text{LeftCoDualityTensorProductCompatibilityMorphismWithGivenObjects}_{a,b} : (a \otimes b)_V \rightarrow a_V \otimes b_V$.

1.7.27 MorphismFromLeftInternalCoHomToTensorProduct (for IsCapCategoryObject, IsCapCategoryObject)

▷ $\text{MorphismFromLeftInternalCoHomToTensorProduct}(a, b)$ (operation)

Returns: a morphism in $\text{Hom}(\text{coHom}_\ell(a, b), b_V \otimes a)$.

The arguments are two objects a, b . The output is the natural morphism $\text{MorphismFromLeftInternalCoHomToTensorProduct}_{a,b} : \text{coHom}_\ell(a, b) \rightarrow b_V \otimes a$.

1.7.28 MorphismFromLeftInternalCoHomToTensorProductWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ $\text{MorphismFromLeftInternalCoHomToTensorProductWithGivenObjects}(s, a, b, r)$ (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = \text{coHom}_\ell(a, b)$, two objects a, b , and an object $r = b_V \otimes a$. The output is the natural morphism $\text{MorphismFromLeftInternalCoHomToTensorProductWithGivenObjects}_{a,b} : \text{coHom}_\ell(a, b) \rightarrow a \otimes b_V$.

1.7.29 IsomorphismFromLeftCoDualObjectToLeftInternalCoHomFromTensorUnit (for IsCapCategoryObject)

▷ $\text{IsomorphismFromLeftCoDualObjectToLeftInternalCoHomFromTensorUnit}(a)$ (attribute)

Returns: a morphism in $\text{Hom}(a_V, \text{coHom}_\ell(1, a))$.

The argument is an object a . The output is the isomorphism $\text{IsomorphismFromLeftCoDualObjectToLeftInternalCoHomFromTensorUnit}_a : a_V \rightarrow \text{coHom}_\ell(1, a)$.

1.7.30 IsomorphismFromLeftInternalCoHomFromTensorUnitToLeftCoDualObject (for IsCapCategoryObject)

▷ $\text{IsomorphismFromLeftInternalCoHomFromTensorUnitToLeftCoDualObject}(a)$ (attribute)

Returns: a morphism in $\text{Hom}(\text{coHom}_\ell(1, a), a_V)$.

The argument is an object a . The output is the isomorphism $\text{IsomorphismFromLeftInternalCoHomFromTensorUnitToLeftCoDualObject}_a : \text{coHom}_\ell(1, a) \rightarrow a_V$.

1.7.31 UniversalPropertyOfLeftCoDual (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ $\text{UniversalPropertyOfLeftCoDual}(t, a, \alpha)$ (operation)

Returns: a morphism in $\text{Hom}(a_V, t)$.

The arguments are two objects t, a , and a morphism $\alpha : 1 \rightarrow t \otimes a$. The output is the morphism $a_V \rightarrow t$ given by the universal property of a_V .

1.7.32 LeftCoclosedMonoidalLambdaIntroduction (for IsCapCategoryMorphism)

▷ LeftCoclosedMonoidalLambdaIntroduction(α) (attribute)

Returns: a morphism in $\text{Hom}(\text{coHom}_\ell(a, b), 1)$.

The argument is a morphism $\alpha : a \rightarrow b$. The output is the corresponding morphism $\text{coHom}_\ell(a, b) \rightarrow 1$ under the cohom tensor adjunction.

1.7.33 LeftCoclosedMonoidalLambdaElimination (for IsCapCategoryObject, IsCapCategoryMorphism)

▷ LeftCoclosedMonoidalLambdaElimination(a, b, α) (operation)

Returns: a morphism in $\text{Hom}(a, b)$.

The arguments are two objects a, b , and a morphism $\alpha : \text{coHom}_\ell(a, b) \rightarrow 1$. The output is a morphism $a \rightarrow b$ corresponding to α under the cohom tensor adjunction.

1.7.34 IsomorphismFromObjectToLeftInternalCoHom (for IsCapCategoryObject)

▷ IsomorphismFromObjectToLeftInternalCoHom(a) (attribute)

Returns: a morphism in $\text{Hom}(a, \text{coHom}_\ell(a, 1))$.

The argument is an object a . The output is the natural isomorphism $a \rightarrow \text{coHom}_\ell(a, 1)$.

1.7.35 IsomorphismFromObjectToLeftInternalCoHomWithGivenLeftInternalCoHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ IsomorphismFromObjectToLeftInternalCoHomWithGivenLeftInternalCoHom(a, r) (operation)

Returns: a morphism in $\text{Hom}(a, r)$.

The argument is an object a , and an object $r = \text{coHom}_\ell(a, 1)$. The output is the natural isomorphism $a \rightarrow \text{coHom}_\ell(a, 1)$.

1.7.36 IsomorphismFromLeftInternalCoHomToObject (for IsCapCategoryObject)

▷ IsomorphismFromLeftInternalCoHomToObject(a) (attribute)

Returns: a morphism in $\text{Hom}(\text{coHom}_\ell(a, 1), a)$.

The argument is an object a . The output is the natural isomorphism $\text{coHom}_\ell(a, 1) \rightarrow a$.

1.7.37 IsomorphismFromLeftInternalCoHomToObjectWithGivenLeftInternalCoHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ IsomorphismFromLeftInternalCoHomToObjectWithGivenLeftInternalCoHom(a, s) (operation)

Returns: a morphism in $\text{Hom}(s, a)$.

The argument is an object a , and an object $s = \text{coHom}_\ell(a, 1)$. The output is the natural isomorphism $\text{coHom}_\ell(a, 1) \rightarrow a$.

1.8 Coclosed Monoidal Categories

A monoidal category \mathbf{C} which has for each functor $- \otimes b : \mathbf{C} \rightarrow \mathbf{C}$ a left adjoint (denoted by $\underline{\text{coHom}}(-, b)$) is called a *coclosed monoidal category*.

If no operations involving coduals are installed manually, the codual objects will be derived as $a_{\vee} := \underline{\text{coHom}}(1, a)$.

The corresponding GAP property is called `IsCoclosedMonoidalCategory`.

1.8.1 InternalCoHomOnObjects (for IsCapCategoryObject, IsCapCategoryObject)

▷ `InternalCoHomOnObjects(a, b)` (operation)

Returns: an object

The arguments are two objects a, b . The output is the internal cohom object $\underline{\text{coHom}}(a, b)$.

1.8.2 InternalCoHomOnMorphisms (for IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `InternalCoHomOnMorphisms(alpha, beta)` (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}(a, b'), \underline{\text{coHom}}(a', b))$

The arguments are two morphisms $\alpha : a \rightarrow a', \beta : b \rightarrow b'$. The output is the internal cohom morphism $\underline{\text{coHom}}(\alpha, \beta) : \underline{\text{coHom}}(a, b') \rightarrow \underline{\text{coHom}}(a', b)$.

1.8.3 InternalCoHomOnMorphismsWithGivenInternalCoHoms (for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryObject)

▷ `InternalCoHomOnMorphismsWithGivenInternalCoHoms(s, alpha, beta, r)` (operation)

Returns: a morphism in $\text{Hom}(s, r)$

The arguments are an object $s = \underline{\text{coHom}}(a, b')$, two morphisms $\alpha : a \rightarrow a', \beta : b \rightarrow b'$, and an object $r = \underline{\text{coHom}}(a', b)$. The output is the internal cohom morphism $\underline{\text{coHom}}(\alpha, \beta) : \underline{\text{coHom}}(a, b') \rightarrow \underline{\text{coHom}}(a', b)$.

1.8.4 CoclosedMonoidalRightEvaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ `CoclosedMonoidalRightEvaluationMorphism(a, b)` (operation)

Returns: a morphism in $\text{Hom}(b, a \otimes \underline{\text{coHom}}(b, a))$.

The arguments are two objects a, b . The output is the coclosed right evaluation morphism $\text{coclev}_{a,b} : b \rightarrow a \otimes \underline{\text{coHom}}(b, a)$, i.e., the unit of the cohom tensor adjunction.

1.8.5 CoclosedMonoidalRightEvaluationMorphismWithGivenRange (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ `CoclosedMonoidalRightEvaluationMorphismWithGivenRange(a, b, r)` (operation)

Returns: a morphism in $\text{Hom}(b, r)$.

The arguments are two objects a, b and an object $r = a \otimes \underline{\text{coHom}}(b, a)$. The output is the coclosed right evaluation morphism $\text{coclev}_{a,b} : b \rightarrow a \otimes \underline{\text{coHom}}(b, a)$, i.e., the unit of the cohom tensor adjunction.

1.8.6 CoclosedMonoidalRightCoevaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ CoclosedMonoidalRightCoevaluationMorphism(a, b) (operation)

Returns: a morphism in $\text{Hom}(\text{coHom}(a \otimes b, a), b)$.

The arguments are two objects a, b . The output is the coclosed right coevaluation morphism $\text{coclcoev}_{a,b} : \text{coHom}(a \otimes b, a) \rightarrow b$, i.e., the counit of the cohom tensor adjunction.

1.8.7 CoclosedMonoidalRightCoevaluationMorphismWithGivenSource (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ CoclosedMonoidalRightCoevaluationMorphismWithGivenSource(a, b, s) (operation)

Returns: a morphism in $\text{Hom}(s, b)$.

The arguments are two objects a, b and an object $s = \text{coHom}(a \otimes b, a)$. The output is the coclosed right coevaluation morphism $\text{coclcoev}_{a,b} : \text{coHom}(a \otimes b, a) \rightarrow b$, i.e., the unit of the cohom tensor adjunction.

1.8.8 TensorProductToInternalCoHomRightAdjunctMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ TensorProductToInternalCoHomRightAdjunctMorphism(b, c, g) (operation)

Returns: a morphism in $\text{Hom}(\text{coHom}(a, b), c)$.

The arguments are two objects b, c and a morphism $g : a \rightarrow b \otimes c$. The output is a morphism $f : \text{coHom}(a, b) \rightarrow c$ corresponding to g under the cohom tensor adjunction.

1.8.9 TensorProductToInternalCoHomRightAdjunctMorphismWithGivenInternalCoHom (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ TensorProductToInternalCoHomRightAdjunctMorphismWithGivenInternalCoHom(b, c, g, i) (operation)

Returns: a morphism in $\text{Hom}(i, c)$.

The arguments are two objects b, c , a morphism $g : a \rightarrow b \otimes c$ and an object $i = \text{coHom}(a, b)$. The output is a morphism $f : \text{coHom}(a, b) \rightarrow c$ corresponding to g under the cohom tensor adjunction.

1.8.10 InternalCoHomToTensorProductRightAdjunctMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ InternalCoHomToTensorProductRightAdjunctMorphism(a, b, f) (operation)

Returns: a morphism in $\text{Hom}(a, b \otimes c)$.

The arguments are two objects a, b and a morphism $f : \text{coHom}(a, b) \rightarrow c$. The output is a morphism $g : a \rightarrow b \otimes c$ corresponding to f under the cohom tensor adjunction.

1.8.11 InternalCoHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ InternalCoHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct(a, b, f, t) (operation)

Returns: a morphism in $\text{Hom}(a, t)$.

The arguments are two objects a, b , a morphism $f : \underline{\text{coHom}}(a, b) \rightarrow c$ and an object $t = b \otimes c$. The output is a morphism $g : a \rightarrow t$ corresponding to f under the cohom tensor adjunction.

1.8.12 CoclosedMonoidalLeftEvaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ CoclosedMonoidalLeftEvaluationMorphism(a, b) (operation)

Returns: a morphism in $\text{Hom}(b, \underline{\text{coHom}}(b, a) \otimes a)$.

The arguments are two objects a, b . The output is the coclosed left evaluation morphism $\text{coclev}_{a,b} : b \rightarrow \underline{\text{coHom}}(b, a) \otimes a$, i.e., the unit of the cohom tensor adjunction.

1.8.13 CoclosedMonoidalLeftEvaluationMorphismWithGivenRange (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ CoclosedMonoidalLeftEvaluationMorphismWithGivenRange(a, b, r) (operation)

Returns: a morphism in $\text{Hom}(b, r)$.

The arguments are two objects a, b and an object $r = \underline{\text{coHom}}(b, a) \otimes a$. The output is the coclosed left evaluation morphism $\text{coclev}_{a,b} : b \rightarrow \underline{\text{coHom}}(b, a) \otimes a$, i.e., the unit of the cohom tensor adjunction.

1.8.14 CoclosedMonoidalLeftCoevaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ CoclosedMonoidalLeftCoevaluationMorphism(a, b) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}(b \otimes a, a), b)$.

The arguments are two objects a, b . The output is the coclosed left coevaluation morphism $\text{coclcoev}_{a,b} : \underline{\text{coHom}}(b \otimes a, a) \rightarrow b$, i.e., the counit of the cohom tensor adjunction.

1.8.15 CoclosedMonoidalLeftCoevaluationMorphismWithGivenSource (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ CoclosedMonoidalLeftCoevaluationMorphismWithGivenSource(a, b, s) (operation)

Returns: a morphism in $\text{Hom}(s, b)$.

The arguments are two objects a, b and an object $s = \underline{\text{coHom}}(b \otimes a, a)$. The output is the coclosed left coevaluation morphism $\text{coclcoev}_{a,b} : \underline{\text{coHom}}(b \otimes a, a) \rightarrow b$, i.e., the unit of the cohom tensor adjunction.

1.8.16 TensorProductToInternalCoHomLeftAdjunctMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ `TensorProductToInternalCoHomLeftAdjunctMorphism(b, c, g)` (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}(a, c), b)$.

The arguments are two objects b, c and a morphism $g : a \rightarrow b \otimes c$. The output is a morphism $f : \underline{\text{coHom}}(a, c) \rightarrow b$ corresponding to g under the cohom tensor adjunction.

1.8.17 TensorProductToInternalCoHomLeftAdjunctMorphismWithGivenInternalCoHom (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ `TensorProductToInternalCoHomLeftAdjunctMorphismWithGivenInternalCoHom(b, c, g, i)` (operation)

Returns: a morphism in $\text{Hom}(i, b)$.

The arguments are two objects b, c , a morphism $g : a \rightarrow b \otimes c$ and an object $i = \underline{\text{coHom}}(a, c)$. The output is a morphism $f : \underline{\text{coHom}}(a, c) \rightarrow b$ corresponding to g under the cohom tensor adjunction.

1.8.18 InternalCoHomToTensorProductLeftAdjunctMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ `InternalCoHomToTensorProductLeftAdjunctMorphism(a, c, f)` (operation)

Returns: a morphism in $\text{Hom}(a, b \otimes c)$.

The arguments are two objects a, c and a morphism $f : \underline{\text{coHom}}(a, c) \rightarrow b$. The output is a morphism $g : a \rightarrow b \otimes c$ corresponding to f under the cohom tensor adjunction.

1.8.19 InternalCoHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ `InternalCoHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct(a, c, f, t)` (operation)

Returns: a morphism in $\text{Hom}(a, t)$.

The arguments are two objects a, c , a morphism $f : \underline{\text{coHom}}(a, c) \rightarrow b$ and an object $t = b \otimes c$. The output is a morphism $g : a \rightarrow t$ corresponding to f under the cohom tensor adjunction.

1.8.20 MonoidalPreCoComposeMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ `MonoidalPreCoComposeMorphism(a, b, c)` (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}(a, c), \underline{\text{coHom}}(b, c) \otimes \underline{\text{coHom}}(a, b))$.

The arguments are three objects a, b, c . The output is the precocomposition morphism $\text{MonoidalPreCoComposeMorphism}_{a,b,c} : \underline{\text{coHom}}(a, c) \rightarrow \underline{\text{coHom}}(b, c) \otimes \underline{\text{coHom}}(a, b)$.

1.8.21 MonoidalPreCoComposeMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ MonoidalPreCoComposeMorphismWithGivenObjects(s, a, b, c, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = \underline{\text{coHom}}(a, c)$, three objects a, b, c , and an object $r = \underline{\text{coHom}}(a, b) \otimes \underline{\text{coHom}}(b, c)$. The output is the precocomposition morphism $\text{MonoidalPreCoComposeMorphismWithGivenObjects}_{a,b,c} : \underline{\text{coHom}}(a, c) \rightarrow \underline{\text{coHom}}(a, b) \otimes \underline{\text{coHom}}(b, c)$.

1.8.22 MonoidalPostCoComposeMorphism (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ MonoidalPostCoComposeMorphism(a, b, c) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}(a, c), \underline{\text{coHom}}(a, b) \otimes \underline{\text{coHom}}(b, c))$.

The arguments are three objects a, b, c . The output is the postcocomposition morphism $\text{MonoidalPostCoComposeMorphism}_{a,b,c} : \underline{\text{coHom}}(a, c) \rightarrow \underline{\text{coHom}}(a, b) \otimes \underline{\text{coHom}}(b, c)$.

1.8.23 MonoidalPostCoComposeMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ MonoidalPostCoComposeMorphismWithGivenObjects(s, a, b, c, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = \underline{\text{coHom}}(a, c)$, three objects a, b, c , and an object $r = \underline{\text{coHom}}(b, c) \otimes \underline{\text{coHom}}(a, b)$. The output is the postcocomposition morphism $\text{MonoidalPostCoComposeMorphismWithGivenObjects}_{a,b,c} : \underline{\text{coHom}}(a, c) \rightarrow \underline{\text{coHom}}(a, b) \otimes \underline{\text{coHom}}(b, c)$.

1.8.24 CoDualOnObjects (for IsCapCategoryObject)

▷ CoDualOnObjects(a) (attribute)

Returns: an object

The argument is an object a . The output is its codual object a_\vee .

1.8.25 CoDualOnMorphisms (for IsCapCategoryMorphism)

▷ CoDualOnMorphisms(α) (attribute)

Returns: a morphism in $\text{Hom}(b_\vee, a_\vee)$.

The argument is a morphism $\alpha : a \rightarrow b$. The output is its codual morphism $\alpha_\vee : b_\vee \rightarrow a_\vee$.

1.8.26 CoDualOnMorphismsWithGivenCoDuals (for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject)

▷ CoDualOnMorphismsWithGivenCoDuals(s, α, r) (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The argument is an object $s = b_\vee$, a morphism $\alpha : a \rightarrow b$, and an object $r = a_\vee$. The output is the dual morphism $\alpha_\vee : b^\vee \rightarrow a^\vee$.

1.8.27 CoclosedEvaluationForCoDual (for IsCapCategoryObject)

▷ `CoclosedEvaluationForCoDual(a)` (attribute)
Returns: a morphism in $\text{Hom}(1, a_\vee \otimes a)$.
 The argument is an object a . The output is the coclosed evaluation morphism $\text{coclev}_a : 1 \rightarrow a_\vee \otimes a$.

1.8.28 CoclosedEvaluationForCoDualWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ `CoclosedEvaluationForCoDualWithGivenTensorProduct(s, a, r)` (operation)
Returns: a morphism in $\text{Hom}(s, r)$.
 The arguments are an object $s = 1$, an object a , and an object $r = a_\vee \otimes a$. The output is the coclosed evaluation morphism $\text{coclev}_a : 1 \rightarrow a_\vee \otimes a$.

1.8.29 MorphismFromCoBidual (for IsCapCategoryObject)

▷ `MorphismFromCoBidual(a)` (attribute)
Returns: a morphism in $\text{Hom}((a_\vee)_\vee, a)$.
 The argument is an object a . The output is the morphism from the cobidual $(a_\vee)_\vee \rightarrow a$.

1.8.30 MorphismFromCoBidualWithGivenCoBidual (for IsCapCategoryObject, IsCapCategoryObject)

▷ `MorphismFromCoBidualWithGivenCoBidual(a, s)` (operation)
Returns: a morphism in $\text{Hom}(s, a)$.
 The arguments are an object a , and an object $s = (a_\vee)_\vee$. The output is the morphism from the cobidual $(a_\vee)_\vee \rightarrow a$.

1.8.31 InternalCoHomTensorProductCompatibilityMorphism (for IsList)

▷ `InternalCoHomTensorProductCompatibilityMorphism(list)` (operation)
Returns: a morphism in $\text{Hom}(\text{coHom}(a \otimes a', b \otimes b'), \text{coHom}(a, b) \otimes \text{coHom}(a', b'))$.
 The argument is a list of four objects $[a, a', b, b']$. The output is the natural morphism $\text{InternalCoHomTensorProductCompatibilityMorphism}_{a, a', b, b'} : \text{coHom}(a \otimes a', b \otimes b') \rightarrow \text{coHom}(a, b) \otimes \text{coHom}(a', b')$.

1.8.32 InternalCoHomTensorProductCompatibilityMorphismWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject)

▷ `InternalCoHomTensorProductCompatibilityMorphismWithGivenObjects(s, list, r)` (operation)
Returns: a morphism in $\text{Hom}(s, r)$.
 The arguments are a list of four objects $[a, a', b, b']$, and two objects $s = \text{coHom}(a \otimes a', b \otimes b')$ and $r = \text{coHom}(a, b) \otimes \text{coHom}(a', b')$. The output is the natural morphism

$\text{InternalCoHomTensorProductCompatibilityMorphismWithGivenObjects}_{a,a',b,b'} : \underline{\text{coHom}}(a \otimes a', b \otimes b') \rightarrow \underline{\text{coHom}}(a, b) \otimes \underline{\text{coHom}}(a', b').$

1.8.33 CoDualityTensorProductCompatibilityMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ $\text{CoDualityTensorProductCompatibilityMorphism}(a, b)$ (operation)

Returns: a morphism in $\text{Hom}((a \otimes b)_\vee, a_\vee \otimes b_\vee)$.

The arguments are two objects a, b . The output is the natural morphism $\text{CoDualityTensorProductCompatibilityMorphism} : (a \otimes b)_\vee \rightarrow a_\vee \otimes b_\vee$.

1.8.34 CoDualityTensorProductCompatibilityMorphismWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ $\text{CoDualityTensorProductCompatibilityMorphismWithGivenObjects}(s, a, b, r)$ (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = (a \otimes b)_\vee$, two objects a, b , and an object $r = a_\vee \otimes b_\vee$. The output is the natural morphism $\text{CoDualityTensorProductCompatibilityMorphismWithGivenObjects}_{a,b} : (a \otimes b)_\vee \rightarrow a_\vee \otimes b_\vee$.

1.8.35 MorphismFromInternalCoHomToTensorProduct (for IsCapCategoryObject, IsCapCategoryObject)

▷ $\text{MorphismFromInternalCoHomToTensorProduct}(a, b)$ (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}(a, b), b_\vee \otimes a)$.

The arguments are two objects a, b . The output is the natural morphism $\text{MorphismFromInternalCoHomToTensorProduct}_{a,b} : \underline{\text{coHom}}(a, b) \rightarrow b_\vee \otimes a$.

1.8.36 MorphismFromInternalCoHomToTensorProductWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ $\text{MorphismFromInternalCoHomToTensorProductWithGivenObjects}(s, a, b, r)$ (operation)

Returns: a morphism in $\text{Hom}(s, r)$.

The arguments are an object $s = \underline{\text{coHom}}(a, b)$, two objects a, b , and an object $r = b_\vee \otimes a$. The output is the natural morphism $\text{MorphismFromInternalCoHomToTensorProductWithGivenObjects}_{a,b} : \underline{\text{coHom}}(a, b) \rightarrow a \otimes b_\vee$.

1.8.37 IsomorphismFromCoDualObjectToInternalCoHomFromTensorUnit (for IsCapCategoryObject)

▷ $\text{IsomorphismFromCoDualObjectToInternalCoHomFromTensorUnit}(a)$ (attribute)

Returns: a morphism in $\text{Hom}(a_\vee, \underline{\text{coHom}}(1, a))$.

The argument is an object a . The output is the isomorphism $\text{IsomorphismFromCoDualObjectToInternalCoHomFromTensorUnit}_a : a_\vee \rightarrow \underline{\text{coHom}}(1, a)$.

1.8.38 IsomorphismFromInternalCoHomFromTensorUnitToCoDualObject (for IsCapCategoryObject)

▷ `IsomorphismFromInternalCoHomFromTensorUnitToCoDualObject(a)` (attribute)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}(1, a), a_\vee)$.

The argument is an object a . The output is the isomorphism $\text{IsomorphismFromInternalCoHomFromTensorUnitToCoDualObject}_a : \underline{\text{coHom}}(1, a) \rightarrow a_\vee$.

1.8.39 UniversalPropertyOfCoDual (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ `UniversalPropertyOfCoDual(t, a, alpha)` (operation)

Returns: a morphism in $\text{Hom}(a_\vee, t)$.

The arguments are two objects t, a , and a morphism $\alpha : 1 \rightarrow t \otimes a$. The output is the morphism $a_\vee \rightarrow t$ given by the universal property of a_\vee .

1.8.40 CoLambdaIntroduction (for IsCapCategoryMorphism)

▷ `CoLambdaIntroduction(alpha)` (attribute)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}(a, b), 1)$.

The argument is a morphism $\alpha : a \rightarrow b$. The output is the corresponding morphism $\underline{\text{coHom}}(a, b) \rightarrow 1$ under the cohom tensor adjunction.

1.8.41 CoLambdaElimination (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism)

▷ `CoLambdaElimination(a, b, alpha)` (operation)

Returns: a morphism in $\text{Hom}(a, b)$.

The arguments are two objects a, b , and a morphism $\alpha : \underline{\text{coHom}}(a, b) \rightarrow 1$. The output is a morphism $a \rightarrow b$ corresponding to α under the cohom tensor adjunction.

1.8.42 IsomorphismFromObjectToInternalCoHom (for IsCapCategoryObject)

▷ `IsomorphismFromObjectToInternalCoHom(a)` (attribute)

Returns: a morphism in $\text{Hom}(a, \underline{\text{coHom}}(a, 1))$.

The argument is an object a . The output is the natural isomorphism $a \rightarrow \underline{\text{coHom}}(a, 1)$.

1.8.43 IsomorphismFromObjectToInternalCoHomWithGivenInternalCoHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ `IsomorphismFromObjectToInternalCoHomWithGivenInternalCoHom(a, r)` (operation)

Returns: a morphism in $\text{Hom}(a, r)$.

The argument is an object a , and an object $r = \underline{\text{coHom}}(a, 1)$. The output is the natural isomorphism $a \rightarrow \underline{\text{coHom}}(a, 1)$.

1.8.44 IsomorphismFromInternalCoHomToObject (for IsCapCategoryObject)

▷ `IsomorphismFromInternalCoHomToObject(a)` (attribute)

Returns: a morphism in $\text{Hom}(\text{coHom}(a, 1), a)$.

The argument is an object a . The output is the natural isomorphism $\text{coHom}(a, 1) \rightarrow a$.

1.8.45 IsomorphismFromInternalCoHomToObjectWithGivenInternalCoHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ `IsomorphismFromInternalCoHomToObjectWithGivenInternalCoHom(a, s)` (operation)

Returns: a morphism in $\text{Hom}(s, a)$.

The argument is an object a , and an object $s = \text{coHom}(a, 1)$. The output is the natural isomorphism $\text{coHom}(a, 1) \rightarrow a$.

1.9 Symmetric Closed Monoidal Categories

A monoidal category \mathbf{C} which is symmetric and closed is called a *symmetric closed monoidal category*.

The corresponding GAP property is given by `IsSymmetricClosedMonoidalCategory`.

1.10 Symmetric Coclosed Monoidal Categories

A monoidal category \mathbf{C} which is symmetric and coclosed is called a *symmetric coclosed monoidal category*.

The corresponding GAP property is given by `IsSymmetricCoclosedMonoidalCategory`.

1.11 Rigid Symmetric Closed Monoidal Categories

A symmetric closed monoidal category \mathbf{C} satisfying

- the natural morphism

$\text{Hom}_\ell(a, a') \otimes \text{Hom}_\ell(b, b') \rightarrow \text{Hom}_\ell(a \otimes b, a' \otimes b')$ is an isomorphism,

- the natural morphism

$a \rightarrow \text{Hom}_\ell(\text{Hom}_\ell(a, 1), 1)$ is an isomorphism is called a *rigid symmetric closed monoidal category*.

If no operations involving the closed structure are installed manually, the internal hom objects will be derived as $\text{Hom}_\ell(a, b) := a^\vee \otimes b$ and, in particular, $\text{Hom}_\ell(a, 1) := a^\vee \otimes 1$.

The corresponding GAP property is given by `IsRigidSymmetricClosedMonoidalCategory`.

1.11.1 IsomorphismFromTensorProductWithDualObjectToInternalHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ `IsomorphismFromTensorProductWithDualObjectToInternalHom(a, b)` (operation)

Returns: a morphism in $\text{Hom}(a^\vee \otimes b, \text{Hom}(a, b))$.

The arguments are two objects a, b . The output is the natural morphism $\text{IsomorphismFromTensorProductWithDualObjectToInternalHom}_{a,b} : a^\vee \otimes b \rightarrow \text{Hom}(a, b)$.

1.11.2 IsomorphismFromInternalHomToTensorProductWithDualObject (for IsCap-CategoryObject, IsCapCategoryObject)

▷ IsomorphismFromInternalHomToTensorProductWithDualObject(a, b) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(a, b), a^\vee \otimes b)$.

The arguments are two objects a, b . The output is the inverse of IsomorphismFromTensorProductWithDualObjectToInternalHom, namely IsomorphismFromInternalHomToTensorProductWithDualObject_{a,b} : $\underline{\text{Hom}}(a, b) \rightarrow a^\vee \otimes b$.

1.11.3 MorphismFromInternalHomToTensorProduct (for IsCapCategoryObject, Is-CapCategoryObject)

▷ MorphismFromInternalHomToTensorProduct(a, b) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(a, b), a^\vee \otimes b)$.

The arguments are two objects a, b . The output is the inverse of MorphismFromTensorProductToInternalHomWithGivenObjects, namely MorphismFromInternalHomToTensorProductWithGivenObjects_{a,b} : $\underline{\text{Hom}}(a, b) \rightarrow a^\vee \otimes b$.

1.11.4 MorphismFromInternalHomToTensorProductWithGivenObjects (for IsCap-CategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ MorphismFromInternalHomToTensorProductWithGivenObjects(s, a, b, r) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(a, b), a^\vee \otimes b)$.

The arguments are an object $s = \underline{\text{Hom}}(a, b)$, two objects a, b , and an object $r = a^\vee \otimes b$. The output is the inverse of MorphismFromTensorProductToInternalHomWithGivenObjects, namely MorphismFromInternalHomToTensorProductWithGivenObjects_{a,b} : $\underline{\text{Hom}}(a, b) \rightarrow a^\vee \otimes b$.

1.11.5 TensorProductInternalHomCompatibilityMorphismInverse (for IsList)

▷ TensorProductInternalHomCompatibilityMorphismInverse(list) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(a \otimes b, a' \otimes b'), \underline{\text{Hom}}(a, a') \otimes \underline{\text{Hom}}(b, b'))$.

The argument is a list of four objects $[a, a', b, b']$. The output is the natural morphism TensorProductInternalHomCompatibilityMorphismInverseWithGivenObjects_{a,a',b,b'} : $\underline{\text{Hom}}(a \otimes b, a' \otimes b') \rightarrow \underline{\text{Hom}}(a, a') \otimes \underline{\text{Hom}}(b, b')$.

1.11.6 TensorProductInternalHomCompatibilityMorphismInverseWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject)

▷ TensorProductInternalHomCompatibilityMorphismInverseWithGivenObjects(s, list, r) (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{Hom}}(a \otimes b, a' \otimes b'), \underline{\text{Hom}}(a, a') \otimes \underline{\text{Hom}}(b, b'))$.

The arguments are a list of four objects $[a, a', b, b']$, and two objects $s = \underline{\text{Hom}}(a \otimes b, a' \otimes b')$ and $r = \underline{\text{Hom}}(a, a') \otimes \underline{\text{Hom}}(b, b')$. The output is the natural morphism TensorProductInternalHomCompatibilityMorphismInverseWithGivenObjects_{a,a',b,b'} : $\underline{\text{Hom}}(a \otimes b, a' \otimes b') \rightarrow \underline{\text{Hom}}(a, a') \otimes \underline{\text{Hom}}(b, b')$.

1.11.7 CoevaluationForDual (for IsCapCategoryObject)

▷ `CoevaluationForDual(a)` (attribute)

Returns: a morphism in $\text{Hom}(1, a \otimes a^\vee)$.

The argument is an object a . The output is the coevaluation morphism $\text{coev}_a : 1 \rightarrow a \otimes a^\vee$.

1.11.8 CoevaluationForDualWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ `CoevaluationForDualWithGivenTensorProduct(s, a, r)` (operation)

Returns: a morphism in $\text{Hom}(1, a \otimes a^\vee)$.

The arguments are an object $s = 1$, an object a , and an object $r = a \otimes a^\vee$. The output is the coevaluation morphism $\text{coev}_a : 1 \rightarrow a \otimes a^\vee$.

1.11.9 TraceMap (for IsCapCategoryMorphism)

▷ `TraceMap(alpha)` (attribute)

Returns: a morphism in $\text{Hom}(1, 1)$.

The argument is an endomorphism $\alpha : a \rightarrow a$. The output is the trace morphism $\text{trace}_\alpha : 1 \rightarrow 1$.

1.11.10 RankMorphism (for IsCapCategoryObject)

▷ `RankMorphism(a)` (attribute)

Returns: a morphism in $\text{Hom}(1, 1)$.

The argument is an object a . The output is the rank morphism $\text{rank}_a : 1 \rightarrow 1$.

1.11.11 MorphismFromBidual (for IsCapCategoryObject)

▷ `MorphismFromBidual(a)` (attribute)

Returns: a morphism in $\text{Hom}((a^\vee)^\vee, a)$.

The argument is an object a . The output is the inverse of the morphism to the bidual $(a^\vee)^\vee \rightarrow a$.

1.11.12 MorphismFromBidualWithGivenBidual (for IsCapCategoryObject, IsCapCategoryObject)

▷ `MorphismFromBidualWithGivenBidual(a, s)` (operation)

Returns: a morphism in $\text{Hom}((a^\vee)^\vee, a)$.

The argument is an object a , and an object $s = (a^\vee)^\vee$. The output is the inverse of the morphism to the bidual $(a^\vee)^\vee \rightarrow a$.

1.12 Rigid Symmetric Coclosed Monoidal Categories

A symmetric coclosed monoidal category \mathbf{C} satisfying

- the natural morphism

$\text{coHom}(a \otimes a', b \otimes b') \rightarrow \text{coHom}(a, b) \otimes \text{coHom}(a', b')$ is an isomorphism,

- the natural morphism

$\text{coHom}(1, \text{coHom}(1, a)) \rightarrow a$ is an isomorphism is called a *rigid symmetric coclosed monoidal category*.

If no operations involving the coclosed structure are installed manually, the internal cohom objects will be derived as $\text{coHom}(a, b) := a \otimes b_V$ and, in particular, $\text{coHom}(1, a) := 1 \otimes a_V$.

The corresponding GAP property is given by `IsRigidSymmetricCoclosedMonoidalCategory`.

1.12.1 IsomorphismFromInternalCoHomToTensorProductWithCoDualObject (for IsCapCategoryObject, IsCapCategoryObject)

▷ `IsomorphismFromInternalCoHomToTensorProductWithCoDualObject(a, b)` (operation)

Returns: a morphism in $\text{Hom}(\text{coHom}(a, b), b_V \otimes a)$.

The arguments are two objects a, b . The output is the natural morphism `IsomorphismFromInternalCoHomToTensorProductWithCoDualObjectWithGivenObjectsa,b` : $\text{coHom}(a, b) \rightarrow b_V \otimes a$.

1.12.2 IsomorphismFromTensorProductWithCoDualObjectToInternalCoHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ `IsomorphismFromTensorProductWithCoDualObjectToInternalCoHom(a, b)` (operation)

Returns: a morphism in $\text{Hom}(a_V \otimes b, \text{coHom}(b, a))$.

The arguments are two objects a, b . The output is the inverse of `IsomorphismFromInternalCoHomToTensorProductWithCoDualObject`, namely `IsomorphismFromTensorProductWithCoDualObjectToInternalCoHoma,b` : $a_V \otimes b \rightarrow \text{coHom}(b, a)$.

1.12.3 MorphismFromTensorProductToInternalCoHom (for IsCapCategoryObject, IsCapCategoryObject)

▷ `MorphismFromTensorProductToInternalCoHom(a, b)` (operation)

Returns: a morphism in $\text{Hom}(a_V \otimes b, \text{coHom}(b, a))$.

The arguments are two objects a, b . The output is the inverse of `MorphismFromInternalCoHomToTensorProductWithGivenObjects`, namely `MorphismFromTensorProductToInternalCoHomWithGivenObjectsa,b` : $a_V \otimes b \rightarrow \text{coHom}(b, a)$.

1.12.4 MorphismFromTensorProductToInternalCoHomWithGivenObjects (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ `MorphismFromTensorProductToInternalCoHomWithGivenObjects(s, a, b, r)` (operation)

Returns: a morphism in $\text{Hom}(a_V \otimes b, \text{coHom}(b, a))$.

The arguments are an object $s_V = a \otimes b$, two objects a, b , and an object $r = \text{coHom}(b, a)$. The output is the inverse of `MorphismFromInternalCoHomToTensorProductWithGivenObjects`, namely `MorphismFromTensorProductToInternalCoHomWithGivenObjectsa,b` : $a_V \otimes b \rightarrow \text{coHom}(b, a)$.

1.12.5 InternalCoHomTensorProductCompatibilityMorphismInverse (for IsList)

▷ `InternalCoHomTensorProductCompatibilityMorphismInverse(list)` (operation)

Returns: a morphism in $\text{Hom}(\text{coHom}(a, b) \otimes \text{coHom}(a', b'), \text{coHom}(a \otimes a', b \otimes b'))$.

The argument is a list of four objects $[a, a', b, b']$. The output is the natural morphism $\text{InternalCoHomTensorProductCompatibilityMorphismInverseWithGivenObjects}_{a, a', b, b'} : \underline{\text{coHom}}(a, b) \otimes \underline{\text{coHom}}(a', b') \rightarrow \underline{\text{coHom}}(a \otimes a', b \otimes b')$.

1.12.6 InternalCoHomTensorProductCompatibilityMorphismInverseWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject)

▷ $\text{InternalCoHomTensorProductCompatibilityMorphismInverseWithGivenObjects}(s, \text{list}, r)$ (operation)

Returns: a morphism in $\text{Hom}(\underline{\text{coHom}}(a, b) \otimes \underline{\text{coHom}}(a', b'), \underline{\text{coHom}}(a \otimes a', b \otimes b'))$.

The arguments are a list of four objects $[a, a', b, b']$, and two objects $s = \underline{\text{coHom}}(a, b) \otimes \underline{\text{coHom}}(a', b')$ and $r = \underline{\text{coHom}}(a \otimes a', b \otimes b')$. The output is the natural morphism $\text{InternalCoHomTensorProductCompatibilityMorphismInverseWithGivenObjects}_{a, a', b, b'} : \underline{\text{coHom}}(a, b) \otimes \underline{\text{coHom}}(a', b') \rightarrow \underline{\text{coHom}}(a \otimes a', b \otimes b')$.

1.12.7 CoclosedCoevaluationForCoDual (for IsCapCategoryObject)

▷ $\text{CoclosedCoevaluationForCoDual}(a)$ (attribute)

Returns: a morphism in $\text{Hom}(a \otimes a_\vee, 1)$.

The argument is an object a . The output is the coclosed coevaluation morphism $\text{coclcov}_a : a \otimes a_\vee \rightarrow 1$.

1.12.8 CoclosedCoevaluationForCoDualWithGivenTensorProduct (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ $\text{CoclosedCoevaluationForCoDualWithGivenTensorProduct}(s, a, r)$ (operation)

Returns: a morphism in $\text{Hom}(a \otimes a_\vee, 1)$.

The arguments are an object $s = a \otimes a_\vee$, an object a , and an object $r = 1$. The output is the coclosed coevaluation morphism $\text{coclcov}_a : a \otimes a_\vee \rightarrow 1$.

1.12.9 CoTraceMap (for IsCapCategoryMorphism)

▷ $\text{CoTraceMap}(\alpha)$ (attribute)

Returns: a morphism in $\text{Hom}(1, 1)$.

The argument is an endomorphism $\alpha : a \rightarrow a$. The output is the cotrace morphism $\text{cotrace}_\alpha : 1 \rightarrow 1$.

1.12.10 CoRankMorphism (for IsCapCategoryObject)

▷ $\text{CoRankMorphism}(a)$ (attribute)

Returns: a morphism in $\text{Hom}(1, 1)$.

The argument is an object a . The output is the corank morphism $\text{corank}_a : 1 \rightarrow 1$.

1.12.11 MorphismToCoBidual (for IsCapCategoryObject)

▷ $\text{MorphismToCoBidual}(a)$ (attribute)

Returns: a morphism in $\text{Hom}(a, (a_\vee)_\vee)$.

The argument is an object a . The output is the inverse of the morphism from the cobidual $a \rightarrow (a_\vee)_\vee$.

1.12.12 MorphismToCoBidualWithGivenCoBidual (for IsCapCategoryObject, IsCapCategoryObject)

▷ `MorphismToCoBidualWithGivenCoBidual(a, r)` (operation)

Returns: a morphism in $\text{Hom}(a, (a_\vee)_\vee)$.

The argument is an object a , and an object $r = (a_\vee)_\vee$. The output is the inverse of the morphism from the cobidual $a \rightarrow (a_\vee)_\vee$.

1.13 Convenience Methods

1.13.1 InternalHom (for IsCapCategoryCell, IsCapCategoryCell)

▷ `InternalHom(a, b)` (operation)

Returns: a cell

This is a convenience method. The arguments are two cells a, b . The output is the internal hom cell. If a, b are two CAP objects the output is the internal Hom object $\underline{\text{Hom}}(a, b)$. If at least one of the arguments is a CAP morphism the output is a CAP morphism, namely the internal hom on morphisms, where any object is replaced by its identity morphism.

1.13.2 InternalCoHom (for IsCapCategoryCell, IsCapCategoryCell)

▷ `InternalCoHom(a, b)` (operation)

Returns: a cell

This is a convenience method. The arguments are two cells a, b . The output is the internal cohom cell. If a, b are two CAP objects the output is the internal cohom object $\underline{\text{coHom}}(a, b)$. If at least one of the arguments is a CAP morphism the output is a CAP morphism, namely the internal cohom on morphisms, where any object is replaced by its identity morphism.

1.13.3 LeftInternalHom (for IsCapCategoryCell, IsCapCategoryCell)

▷ `LeftInternalHom(a, b)` (operation)

Returns: a cell

This is a convenience method. The arguments are two cells a, b . The output is the internal hom cell. If a, b are two CAP objects the output is the internal Hom object $\underline{\text{Hom}}_\ell(a, b)$. If at least one of the arguments is a CAP morphism the output is a CAP morphism, namely the internal hom on morphisms, where any object is replaced by its identity morphism.

1.13.4 LeftInternalCoHom (for IsCapCategoryCell, IsCapCategoryCell)

▷ `LeftInternalCoHom(a, b)` (operation)

Returns: a cell

This is a convenience method. The arguments are two cells a, b . The output is the internal cohom cell. If a, b are two CAP objects the output is the internal cohom object $\underline{\text{coHom}}_\ell(a, b)$. If at least one of the arguments is a CAP morphism the output is a CAP morphism, namely the internal cohom on morphisms, where any object is replaced by its identity morphism.

1.14 Add-methods

1.14.1 AddLeftDistributivityExpanding (for IsCapCategory, IsFunction)

▷ AddLeftDistributivityExpanding(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftDistributivityExpanding. $F : (a, L) \mapsto \text{LeftDistributivityExpanding}(a, L)$.

1.14.2 AddLeftDistributivityExpandingWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddLeftDistributivityExpandingWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftDistributivityExpandingWithGivenObjects. $F : (s, a, L, r) \mapsto \text{LeftDistributivityExpandingWithGivenObjects}(s, a, L, r)$.

1.14.3 AddLeftDistributivityFactoring (for IsCapCategory, IsFunction)

▷ AddLeftDistributivityFactoring(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftDistributivityFactoring. $F : (a, L) \mapsto \text{LeftDistributivityFactoring}(a, L)$.

1.14.4 AddLeftDistributivityFactoringWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddLeftDistributivityFactoringWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftDistributivityFactoringWithGivenObjects. $F : (s, a, L, r) \mapsto \text{LeftDistributivityFactoringWithGivenObjects}(s, a, L, r)$.

1.14.5 AddRightDistributivityExpanding (for IsCapCategory, IsFunction)

▷ AddRightDistributivityExpanding(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation RightDistributivityExpanding. $F : (L, a) \mapsto \text{RightDistributivityExpanding}(L, a)$.

1.14.6 AddRightDistributivityExpandingWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddRightDistributivityExpandingWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation RightDistributivityExpandingWithGivenObjects. $F : (s, L, a, r) \mapsto \text{RightDistributivityExpandingWithGivenObjects}(s, L, a, r)$.

1.14.7 AddRightDistributivityFactoring (for IsCapCategory, IsFunction)

▷ AddRightDistributivityFactoring(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation RightDistributivityFactoring. $F : (L, a) \mapsto \text{RightDistributivityFactoring}(L, a)$.

1.14.8 AddRightDistributivityFactoringWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddRightDistributivityFactoringWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation RightDistributivityFactoringWithGivenObjects. $F : (s, L, a, r) \mapsto \text{RightDistributivityFactoringWithGivenObjects}(s, L, a, r)$.

1.14.9 AddBraiding (for IsCapCategory, IsFunction)

▷ AddBraiding(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation Braiding. $F : (a, b) \mapsto \text{Braiding}(a, b)$.

1.14.10 AddBraidingInverse (for IsCapCategory, IsFunction)

▷ AddBraidingInverse(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation BraidingInverse. $F : (a, b) \mapsto \text{BraidingInverse}(a, b)$.

1.14.11 AddBraidingInverseWithGivenTensorProducts (for IsCapCategory, IsFunction)

▷ AddBraidingInverseWithGivenTensorProducts(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation BraidingInverseWithGivenTensorProducts. $F : (s, a, b, r) \mapsto \text{BraidingInverseWithGivenTensorProducts}(s, a, b, r)$.

1.14.12 AddBraidingWithGivenTensorProducts (for IsCapCategory, IsFunction)

▷ AddBraidingWithGivenTensorProducts(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation BraidingWithGivenTensorProducts. $F : (s, a, b, r) \mapsto \text{BraidingWithGivenTensorProducts}(s, a, b, r)$.

1.14.13 AddClosedMonoidalLeftCoevaluationMorphism (for IsCapCategory, IsFunction)

▷ AddClosedMonoidalLeftCoevaluationMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation ClosedMonoidalLeftCoevaluationMorphism. $F : (a, b) \mapsto \text{ClosedMonoidalLeftCoevaluationMorphism}(a, b)$.

1.14.14 AddClosedMonoidalLeftCoevaluationMorphismWithGivenRange (for IsCapCategory, IsFunction)

▷ AddClosedMonoidalLeftCoevaluationMorphismWithGivenRange(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation ClosedMonoidalLeftCoevaluationMorphismWithGivenRange. $F : (a, b, r) \mapsto \text{ClosedMonoidalLeftCoevaluationMorphismWithGivenRange}(a, b, r)$.

1.14.15 AddClosedMonoidalLeftEvaluationMorphism (for IsCapCategory, IsFunction)

▷ AddClosedMonoidalLeftEvaluationMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation ClosedMonoidalLeftEvaluationMorphism. $F : (a, b) \mapsto \text{ClosedMonoidalLeftEvaluationMorphism}(a, b)$.

1.14.16 AddClosedMonoidalLeftEvaluationMorphismWithGivenSource (for IsCapCategory, IsFunction)

▷ AddClosedMonoidalLeftEvaluationMorphismWithGivenSource(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation ClosedMonoidalLeftEvaluationMorphismWithGivenSource. $F : (a, b, s) \mapsto \text{ClosedMonoidalLeftEvaluationMorphismWithGivenSource}(a, b, s)$.

1.14.17 AddClosedMonoidalRightCoevaluationMorphism (for IsCapCategory, IsFunction)

▷ AddClosedMonoidalRightCoevaluationMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation ClosedMonoidalRightCoevaluationMorphism. $F : (a, b) \mapsto \text{ClosedMonoidalRightCoevaluationMorphism}(a, b)$.

1.14.18 AddClosedMonoidalRightCoevaluationMorphismWithGivenRange (for IsCapCategory, IsFunction)

▷ AddClosedMonoidalRightCoevaluationMorphismWithGivenRange(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation ClosedMonoidalRightCoevaluationMorphismWithGivenRange. $F : (a, b, r) \mapsto \text{ClosedMonoidalRightCoevaluationMorphismWithGivenRange}(a, b, r)$.

1.14.19 AddClosedMonoidalRightEvaluationMorphism (for IsCapCategory, IsFunction)

▷ AddClosedMonoidalRightEvaluationMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation ClosedMonoidalRightEvaluationMorphism. $F : (a, b) \mapsto \text{ClosedMonoidalRightEvaluationMorphism}(a, b)$.

1.14.20 AddClosedMonoidalRightEvaluationMorphismWithGivenSource (for IsCapCategory, IsFunction)

▷ AddClosedMonoidalRightEvaluationMorphismWithGivenSource(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation ClosedMonoidalRightEvaluationMorphismWithGivenSource. $F : (a, b, s) \mapsto \text{ClosedMonoidalRightEvaluationMorphismWithGivenSource}(a, b, s)$.

1.14.21 AddDualOnMorphisms (for IsCapCategory, IsFunction)

▷ AddDualOnMorphisms(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation DualOnMorphisms. $F : (\alpha) \mapsto \text{DualOnMorphisms}(\alpha)$.

1.14.22 AddDualOnMorphismsWithGivenDuals (for IsCapCategory, IsFunction)

▷ AddDualOnMorphismsWithGivenDuals(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `DualOnMorphismsWithGivenDuals`. $F : (s, \alpha, r) \mapsto \text{DualOnMorphismsWithGivenDuals}(s, \alpha, r)$.

1.14.23 AddDualOnObjects (for IsCapCategory, IsFunction)

▷ `AddDualOnObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `DualOnObjects`. $F : (a) \mapsto \text{DualOnObjects}(a)$.

1.14.24 AddEvaluationForDual (for IsCapCategory, IsFunction)

▷ `AddEvaluationForDual(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `EvaluationForDual`. $F : (a) \mapsto \text{EvaluationForDual}(a)$.

1.14.25 AddEvaluationForDualWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ `AddEvaluationForDualWithGivenTensorProduct(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `EvaluationForDualWithGivenTensorProduct`. $F : (s, a, r) \mapsto \text{EvaluationForDualWithGivenTensorProduct}(s, a, r)$.

1.14.26 AddInternalHomOnMorphisms (for IsCapCategory, IsFunction)

▷ `AddInternalHomOnMorphisms(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `InternalHomOnMorphisms`. $F : (\alpha, \beta) \mapsto \text{InternalHomOnMorphisms}(\alpha, \beta)$.

1.14.27 AddInternalHomOnMorphismsWithGivenInternalHoms (for IsCapCategory, IsFunction)

▷ `AddInternalHomOnMorphismsWithGivenInternalHoms(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `InternalHomOnMorphismsWithGivenInternalHoms`. $F : (s, \alpha, \beta, r) \mapsto \text{InternalHomOnMorphismsWithGivenInternalHoms}(s, \alpha, \beta, r)$.

1.14.28 AddInternalHomOnObjects (for IsCapCategory, IsFunction)

▷ `AddInternalHomOnObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `InternalHomOnObjects`. $F : (a,b) \mapsto \text{InternalHomOnObjects}(a,b)$.

1.14.29 AddInternalHomToTensorProductLeftAdjunctMorphism (for IsCapCategory, IsFunction)

▷ `AddInternalHomToTensorProductLeftAdjunctMorphism(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `InternalHomToTensorProductLeftAdjunctMorphism`. $F : (b,c,g) \mapsto \text{InternalHomToTensorProductLeftAdjunctMorphism}(b,c,g)$.

1.14.30 AddInternalHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ `AddInternalHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `InternalHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct`. $F : (b,c,g,s) \mapsto \text{InternalHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct}(b,c,g,s)$.

1.14.31 AddInternalHomToTensorProductLeftAdjunctionIsomorphism (for IsCapCategory, IsFunction)

▷ `AddInternalHomToTensorProductLeftAdjunctionIsomorphism(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `InternalHomToTensorProductLeftAdjunctionIsomorphism`. $F : (a,b,c) \mapsto \text{InternalHomToTensorProductLeftAdjunctionIsomorphism}(a,b,c)$.

1.14.32 AddInternalHomToTensorProductLeftAdjunctionIsomorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ `AddInternalHomToTensorProductLeftAdjunctionIsomorphismWithGivenObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `InternalHomToTensorProductLeftAdjunctionIsomorphismWithGivenObjects`. $F : (s,a,b,c,r) \mapsto \text{InternalHomToTensorProductLeftAdjunctionIsomorphismWithGivenObjects}(s,a,b,c,r)$.

1.14.33 AddInternalHomToTensorProductRightAdjunctMorphism (for IsCapCategory, IsFunction)

▷ AddInternalHomToTensorProductRightAdjunctMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalHomToTensorProductRightAdjunctMorphism. $F : (a, c, g) \mapsto \text{InternalHomToTensorProductRightAdjunctMorphism}(a, c, g)$.

1.14.34 AddInternalHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddInternalHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct. $F : (a, c, g, s) \mapsto \text{InternalHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct}(a, c, g, s)$.

1.14.35 AddInternalHomToTensorProductRightAdjunctionIsomorphism (for IsCapCategory, IsFunction)

▷ AddInternalHomToTensorProductRightAdjunctionIsomorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalHomToTensorProductRightAdjunctionIsomorphism. $F : (a, b, c) \mapsto \text{InternalHomToTensorProductRightAdjunctionIsomorphism}(a, b, c)$.

1.14.36 AddInternalHomToTensorProductRightAdjunctionIsomorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddInternalHomToTensorProductRightAdjunctionIsomorphismWithGivenObjects(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalHomToTensorProductRightAdjunctionIsomorphismWithGivenObjects. $F : (s, a, b, c, r) \mapsto \text{InternalHomToTensorProductRightAdjunctionIsomorphismWithGivenObjects}(s, a, b, c, r)$.

1.14.37 AddIsomorphismFromDualObjectToInternalHomIntoTensorUnit (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromDualObjectToInternalHomIntoTensorUnit(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromDualObjectToInternalHomIntoTensorUnit`.
 $F : (a) \mapsto \text{IsomorphismFromDualObjectToInternalHomIntoTensorUnit}(a)$.

1.14.38 AddIsomorphismFromInternalHomIntoTensorUnitToDualObject (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromInternalHomIntoTensorUnitToDualObject(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromInternalHomIntoTensorUnitToDualObject`.
 $F : (a) \mapsto \text{IsomorphismFromInternalHomIntoTensorUnitToDualObject}(a)$.

1.14.39 AddIsomorphismFromInternalHomToObject (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromInternalHomToObject(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromInternalHomToObject`. $F : (a) \mapsto \text{IsomorphismFromInternalHomToObject}(a)$.

1.14.40 AddIsomorphismFromInternalHomToObjectWithGivenInternalHom (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromInternalHomToObjectWithGivenInternalHom(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromInternalHomToObjectWithGivenInternalHom`. $F : (a, s) \mapsto \text{IsomorphismFromInternalHomToObjectWithGivenInternalHom}(a, s)$.

1.14.41 AddIsomorphismFromObjectToInternalHom (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromObjectToInternalHom(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromObjectToInternalHom`. $F : (a) \mapsto \text{IsomorphismFromObjectToInternalHom}(a)$.

1.14.42 AddIsomorphismFromObjectToInternalHomWithGivenInternalHom (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromObjectToInternalHomWithGivenInternalHom(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation

IsomorphismFromObjectToInternalHomWithGivenInternalHom. $F : (a, r) \mapsto$
 IsomorphismFromObjectToInternalHomWithGivenInternalHom(a, r).

1.14.43 AddLambdaElimination (for IsCapCategory, IsFunction)

▷ AddLambdaElimination(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LambdaElimination. $F : (a, b, \alpha) \mapsto$ LambdaElimination(a, b, α).

1.14.44 AddLambdaIntroduction (for IsCapCategory, IsFunction)

▷ AddLambdaIntroduction(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LambdaIntroduction. $F : (\alpha) \mapsto$ LambdaIntroduction(α).

1.14.45 AddMonoidalPostComposeMorphism (for IsCapCategory, IsFunction)

▷ AddMonoidalPostComposeMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MonoidalPostComposeMorphism. $F : (a, b, c) \mapsto$ MonoidalPostComposeMorphism(a, b, c).

1.14.46 AddMonoidalPostComposeMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddMonoidalPostComposeMorphismWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MonoidalPostComposeMorphismWithGivenObjects. $F : (s, a, b, c, r) \mapsto$ MonoidalPostComposeMorphismWithGivenObjects(s, a, b, c, r).

1.14.47 AddMonoidalPreComposeMorphism (for IsCapCategory, IsFunction)

▷ AddMonoidalPreComposeMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MonoidalPreComposeMorphism. $F : (a, b, c) \mapsto$ MonoidalPreComposeMorphism(a, b, c).

1.14.48 AddMonoidalPreComposeMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddMonoidalPreComposeMorphismWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MonoidalPreComposeMorphismWithGivenObjects. $F : (s, a, b, c, r) \mapsto \text{MonoidalPreComposeMorphismWithGivenObjects}(s, a, b, c, r)$.

1.14.49 AddMorphismFromTensorProductToInternalHom (for IsCapCategory, IsFunction)

▷ AddMorphismFromTensorProductToInternalHom(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromTensorProductToInternalHom. $F : (a, b) \mapsto \text{MorphismFromTensorProductToInternalHom}(a, b)$.

1.14.50 AddMorphismFromTensorProductToInternalHomWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddMorphismFromTensorProductToInternalHomWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromTensorProductToInternalHomWithGivenObjects. $F : (s, a, b, r) \mapsto \text{MorphismFromTensorProductToInternalHomWithGivenObjects}(s, a, b, r)$.

1.14.51 AddMorphismToBidual (for IsCapCategory, IsFunction)

▷ AddMorphismToBidual(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismToBidual. $F : (a) \mapsto \text{MorphismToBidual}(a)$.

1.14.52 AddMorphismToBidualWithGivenBidual (for IsCapCategory, IsFunction)

▷ AddMorphismToBidualWithGivenBidual(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismToBidualWithGivenBidual. $F : (a, r) \mapsto \text{MorphismToBidualWithGivenBidual}(a, r)$.

1.14.53 AddTensorProductDualityCompatibilityMorphism (for IsCapCategory, IsFunction)

▷ AddTensorProductDualityCompatibilityMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `TensorProductDualityCompatibilityMorphism`. $F : (a, b) \mapsto \text{TensorProductDualityCompatibilityMorphism}(a, b)$.

1.14.54 AddTensorProductDualityCompatibilityMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ `AddTensorProductDualityCompatibilityMorphismWithGivenObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `TensorProductDualityCompatibilityMorphismWithGivenObjects`. $F : (s, a, b, r) \mapsto \text{TensorProductDualityCompatibilityMorphismWithGivenObjects}(s, a, b, r)$.

1.14.55 AddTensorProductInternalHomCompatibilityMorphism (for IsCapCategory, IsFunction)

▷ `AddTensorProductInternalHomCompatibilityMorphism(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `TensorProductInternalHomCompatibilityMorphism`. $F : (list) \mapsto \text{TensorProductInternalHomCompatibilityMorphism}(list)$.

1.14.56 AddTensorProductInternalHomCompatibilityMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ `AddTensorProductInternalHomCompatibilityMorphismWithGivenObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `TensorProductInternalHomCompatibilityMorphismWithGivenObjects`. $F : (source, list, range) \mapsto \text{TensorProductInternalHomCompatibilityMorphismWithGivenObjects}(source, list, range)$.

1.14.57 AddTensorProductToInternalHomLeftAdjunctMorphism (for IsCapCategory, IsFunction)

▷ `AddTensorProductToInternalHomLeftAdjunctMorphism(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `TensorProductToInternalHomLeftAdjunctMorphism`. $F : (a, b, f) \mapsto \text{TensorProductToInternalHomLeftAdjunctMorphism}(a, b, f)$.

1.14.58 AddTensorProductToInternalHomLeftAdjunctMorphismWithGivenInternalHom (for IsCapCategory, IsFunction)

▷ AddTensorProductToInternalHomLeftAdjunctMorphismWithGivenInternalHom(C, F)
(operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToInternalHomLeftAdjunctMorphismWithGivenInternalHom. $F : (a, b, f, i) \mapsto \text{TensorProductToInternalHomLeftAdjunctMorphismWithGivenInternalHom}(a, b, f, i)$.

1.14.59 AddTensorProductToInternalHomLeftAdjunctionIsomorphism (for IsCapCategory, IsFunction)

▷ AddTensorProductToInternalHomLeftAdjunctionIsomorphism(C, F)
(operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToInternalHomLeftAdjunctionIsomorphism. $F : (a, b, c) \mapsto \text{TensorProductToInternalHomLeftAdjunctionIsomorphism}(a, b, c)$.

1.14.60 AddTensorProductToInternalHomLeftAdjunctionIsomorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddTensorProductToInternalHomLeftAdjunctionIsomorphismWithGivenObjects(C, F)
(operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToInternalHomLeftAdjunctionIsomorphismWithGivenObjects. $F : (s, a, b, c, r) \mapsto \text{TensorProductToInternalHomLeftAdjunctionIsomorphismWithGivenObjects}(s, a, b, c, r)$.

1.14.61 AddTensorProductToInternalHomRightAdjunctMorphism (for IsCapCategory, IsFunction)

▷ AddTensorProductToInternalHomRightAdjunctMorphism(C, F)
(operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToInternalHomRightAdjunctMorphism. $F : (a, b, f) \mapsto \text{TensorProductToInternalHomRightAdjunctMorphism}(a, b, f)$.

1.14.62 AddTensorProductToInternalHomRightAdjunctMorphismWithGivenInternalHom (for IsCapCategory, IsFunction)

▷ AddTensorProductToInternalHomRightAdjunctMorphismWithGivenInternalHom(C, F)
(operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `TensorProductToInternalHomRightAdjunctMorphismWithGivenInternalHom`. $F : (a, b, f, i) \mapsto \text{TensorProductToInternalHomRightAdjunctMorphismWithGivenInternalHom}(a, b, f, i)$.

1.14.63 AddTensorProductToInternalHomRightAdjunctionIsomorphism (for IsCapCategory, IsFunction)

▷ `AddTensorProductToInternalHomRightAdjunctionIsomorphism(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `TensorProductToInternalHomRightAdjunctionIsomorphism`. $F : (a, b, c) \mapsto \text{TensorProductToInternalHomRightAdjunctionIsomorphism}(a, b, c)$.

1.14.64 AddTensorProductToInternalHomRightAdjunctionIsomorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ `AddTensorProductToInternalHomRightAdjunctionIsomorphismWithGivenObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `TensorProductToInternalHomRightAdjunctionIsomorphismWithGivenObjects`. $F : (s, a, b, c, r) \mapsto \text{TensorProductToInternalHomRightAdjunctionIsomorphismWithGivenObjects}(s, a, b, c, r)$.

1.14.65 AddUniversalPropertyOfDual (for IsCapCategory, IsFunction)

▷ `AddUniversalPropertyOfDual(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `UniversalPropertyOfDual`. $F : (t, a, \alpha) \mapsto \text{UniversalPropertyOfDual}(t, a, \alpha)$.

1.14.66 AddCoDualOnMorphisms (for IsCapCategory, IsFunction)

▷ `AddCoDualOnMorphisms(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `CoDualOnMorphisms`. $F : (\alpha) \mapsto \text{CoDualOnMorphisms}(\alpha)$.

1.14.67 AddCoDualOnMorphismsWithGivenCoDuals (for IsCapCategory, IsFunction)

▷ `AddCoDualOnMorphismsWithGivenCoDuals(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `CoDualOnMorphismsWithGivenCoDuals`. $F : (s, \alpha, r) \mapsto \text{CoDualOnMorphismsWithGivenCoDuals}(s, \alpha, r)$.

1.14.68 AddCoDualOnObjects (for IsCapCategory, IsFunction)

▷ `AddCoDualOnObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `CoDualOnObjects`. $F : (a) \mapsto \text{CoDualOnObjects}(a)$.

1.14.69 AddCoDualityTensorProductCompatibilityMorphism (for IsCapCategory, IsFunction)

▷ `AddCoDualityTensorProductCompatibilityMorphism(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `CoDualityTensorProductCompatibilityMorphism`. $F : (a, b) \mapsto \text{CoDualityTensorProductCompatibilityMorphism}(a, b)$.

1.14.70 AddCoDualityTensorProductCompatibilityMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ `AddCoDualityTensorProductCompatibilityMorphismWithGivenObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `CoDualityTensorProductCompatibilityMorphismWithGivenObjects`. $F : (s, a, b, r) \mapsto \text{CoDualityTensorProductCompatibilityMorphismWithGivenObjects}(s, a, b, r)$.

1.14.71 AddCoLambdaElimination (for IsCapCategory, IsFunction)

▷ `AddCoLambdaElimination(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `CoLambdaElimination`. $F : (a, b, \alpha) \mapsto \text{CoLambdaElimination}(a, b, \alpha)$.

1.14.72 AddCoLambdaIntroduction (for IsCapCategory, IsFunction)

▷ `AddCoLambdaIntroduction(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `CoLambdaIntroduction`. $F : (\alpha) \mapsto \text{CoLambdaIntroduction}(\alpha)$.

1.14.73 AddCoclosedEvaluationForCoDual (for IsCapCategory, IsFunction)

▷ AddCoclosedEvaluationForCoDual(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedEvaluationForCoDual. $F : (a) \mapsto \text{CoclosedEvaluationForCoDual}(a)$.

1.14.74 AddCoclosedEvaluationForCoDualWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddCoclosedEvaluationForCoDualWithGivenTensorProduct(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedEvaluationForCoDualWithGivenTensorProduct. $F : (s, a, r) \mapsto \text{CoclosedEvaluationForCoDualWithGivenTensorProduct}(s, a, r)$.

1.14.75 AddCoclosedMonoidalLeftCoevaluationMorphism (for IsCapCategory, IsFunction)

▷ AddCoclosedMonoidalLeftCoevaluationMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedMonoidalLeftCoevaluationMorphism. $F : (a, b) \mapsto \text{CoclosedMonoidalLeftCoevaluationMorphism}(a, b)$.

1.14.76 AddCoclosedMonoidalLeftCoevaluationMorphismWithGivenSource (for IsCapCategory, IsFunction)

▷ AddCoclosedMonoidalLeftCoevaluationMorphismWithGivenSource(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedMonoidalLeftCoevaluationMorphismWithGivenSource. $F : (a, b, s) \mapsto \text{CoclosedMonoidalLeftCoevaluationMorphismWithGivenSource}(a, b, s)$.

1.14.77 AddCoclosedMonoidalLeftEvaluationMorphism (for IsCapCategory, IsFunction)

▷ AddCoclosedMonoidalLeftEvaluationMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedMonoidalLeftEvaluationMorphism. $F : (a, b) \mapsto \text{CoclosedMonoidalLeftEvaluationMorphism}(a, b)$.

1.14.78 AddCoclosedMonoidalLeftEvaluationMorphismWithGivenRange (for IsCapCategory, IsFunction)

▷ AddCoclosedMonoidalLeftEvaluationMorphismWithGivenRange(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedMonoidalLeftEvaluationMorphismWithGivenRange. $F : (a, b, r) \mapsto \text{CoclosedMonoidalLeftEvaluationMorphismWithGivenRange}(a, b, r)$.

1.14.79 AddCoclosedMonoidalRightCoevaluationMorphism (for IsCapCategory, IsFunction)

▷ AddCoclosedMonoidalRightCoevaluationMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedMonoidalRightCoevaluationMorphism. $F : (a, b) \mapsto \text{CoclosedMonoidalRightCoevaluationMorphism}(a, b)$.

1.14.80 AddCoclosedMonoidalRightCoevaluationMorphismWithGivenSource (for IsCapCategory, IsFunction)

▷ AddCoclosedMonoidalRightCoevaluationMorphismWithGivenSource(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedMonoidalRightCoevaluationMorphismWithGivenSource. $F : (a, b, s) \mapsto \text{CoclosedMonoidalRightCoevaluationMorphismWithGivenSource}(a, b, s)$.

1.14.81 AddCoclosedMonoidalRightEvaluationMorphism (for IsCapCategory, IsFunction)

▷ AddCoclosedMonoidalRightEvaluationMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedMonoidalRightEvaluationMorphism. $F : (a, b) \mapsto \text{CoclosedMonoidalRightEvaluationMorphism}(a, b)$.

1.14.82 AddCoclosedMonoidalRightEvaluationMorphismWithGivenRange (for IsCapCategory, IsFunction)

▷ AddCoclosedMonoidalRightEvaluationMorphismWithGivenRange(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedMonoidalRightEvaluationMorphismWithGivenRange. $F : (a, b, r) \mapsto \text{CoclosedMonoidalRightEvaluationMorphismWithGivenRange}(a, b, r)$.

1.14.83 AddInternalCoHomOnMorphisms (for IsCapCategory, IsFunction)

▷ AddInternalCoHomOnMorphisms(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalCoHomOnMorphisms. $F : (\alpha, \beta) \mapsto \text{InternalCoHomOnMorphisms}(\alpha, \beta)$.

1.14.84 AddInternalCoHomOnMorphismsWithGivenInternalCoHoms (for IsCapCategory, IsFunction)

▷ AddInternalCoHomOnMorphismsWithGivenInternalCoHoms(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalCoHomOnMorphismsWithGivenInternalCoHoms. $F : (s, \alpha, \beta, r) \mapsto \text{InternalCoHomOnMorphismsWithGivenInternalCoHoms}(s, \alpha, \beta, r)$.

1.14.85 AddInternalCoHomOnObjects (for IsCapCategory, IsFunction)

▷ AddInternalCoHomOnObjects(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalCoHomOnObjects. $F : (a, b) \mapsto \text{InternalCoHomOnObjects}(a, b)$.

1.14.86 AddInternalCoHomTensorProductCompatibilityMorphism (for IsCapCategory, IsFunction)

▷ AddInternalCoHomTensorProductCompatibilityMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalCoHomTensorProductCompatibilityMorphism. $F : (list) \mapsto \text{InternalCoHomTensorProductCompatibilityMorphism}(list)$.

1.14.87 AddInternalCoHomTensorProductCompatibilityMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddInternalCoHomTensorProductCompatibilityMorphismWithGivenObjects(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalCoHomTensorProductCompatibilityMorphismWithGivenObjects. $F : (source, list, range) \mapsto \text{InternalCoHomTensorProductCompatibilityMorphismWithGivenObjects}(source, list, range)$.

1.14.88 AddInternalCoHomToTensorProductLeftAdjunctMorphism (for IsCapCategory, IsFunction)

▷ AddInternalCoHomToTensorProductLeftAdjunctMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalCoHomToTensorProductLeftAdjunctMorphism. $F : (a, c, f) \mapsto \text{InternalCoHomToTensorProductLeftAdjunctMorphism}(a, c, f)$.

1.14.89 AddInternalCoHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddInternalCoHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalCoHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct. $F : (a, c, f, t) \mapsto \text{InternalCoHomToTensorProductLeftAdjunctMorphismWithGivenTensorProduct}(a, c, f, t)$.

1.14.90 AddInternalCoHomToTensorProductRightAdjunctMorphism (for IsCapCategory, IsFunction)

▷ AddInternalCoHomToTensorProductRightAdjunctMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalCoHomToTensorProductRightAdjunctMorphism. $F : (a, b, f) \mapsto \text{InternalCoHomToTensorProductRightAdjunctMorphism}(a, b, f)$.

1.14.91 AddInternalCoHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddInternalCoHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalCoHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct. $F : (a, b, f, t) \mapsto \text{InternalCoHomToTensorProductRightAdjunctMorphismWithGivenTensorProduct}(a, b, f, t)$.

1.14.92 AddIsomorphismFromCoDualObjectToInternalCoHomFromTensorUnit (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromCoDualObjectToInternalCoHomFromTensorUnit(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromCoDualObjectToInternalCoHomFromTensorUnit`. $F : (a) \mapsto \text{IsomorphismFromCoDualObjectToInternalCoHomFromTensorUnit}(a)$.

1.14.93 AddIsomorphismFromInternalCoHomFromTensorUnitToCoDualObject (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromInternalCoHomFromTensorUnitToCoDualObject(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromInternalCoHomFromTensorUnitToCoDualObject`. $F : (a) \mapsto \text{IsomorphismFromInternalCoHomFromTensorUnitToCoDualObject}(a)$.

1.14.94 AddIsomorphismFromInternalCoHomToObject (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromInternalCoHomToObject(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromInternalCoHomToObject`. $F : (a) \mapsto \text{IsomorphismFromInternalCoHomToObject}(a)$.

1.14.95 AddIsomorphismFromInternalCoHomToObjectWithGivenInternalCoHom (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromInternalCoHomToObjectWithGivenInternalCoHom(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromInternalCoHomToObjectWithGivenInternalCoHom`. $F : (a, s) \mapsto \text{IsomorphismFromInternalCoHomToObjectWithGivenInternalCoHom}(a, s)$.

1.14.96 AddIsomorphismFromObjectToInternalCoHom (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromObjectToInternalCoHom(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromObjectToInternalCoHom`. $F : (a) \mapsto \text{IsomorphismFromObjectToInternalCoHom}(a)$.

1.14.97 AddIsomorphismFromObjectToInternalCoHomWithGivenInternalCoHom (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromObjectToInternalCoHomWithGivenInternalCoHom(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromObjectToInternalCoHomWithGivenInternalCoHom`. $F : (a, r) \mapsto \text{IsomorphismFromObjectToInternalCoHomWithGivenInternalCoHom}(a, r)$.

1.14.98 AddMonoidalPostCoComposeMorphism (for IsCapCategory, IsFunction)

▷ `AddMonoidalPostCoComposeMorphism(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `MonoidalPostCoComposeMorphism`. $F : (a, b, c) \mapsto \text{MonoidalPostCoComposeMorphism}(a, b, c)$.

1.14.99 AddMonoidalPostCoComposeMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ `AddMonoidalPostCoComposeMorphismWithGivenObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `MonoidalPostCoComposeMorphismWithGivenObjects`. $F : (s, a, b, c, r) \mapsto \text{MonoidalPostCoComposeMorphismWithGivenObjects}(s, a, b, c, r)$.

1.14.100 AddMonoidalPreCoComposeMorphism (for IsCapCategory, IsFunction)

▷ `AddMonoidalPreCoComposeMorphism(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `MonoidalPreCoComposeMorphism`. $F : (a, b, c) \mapsto \text{MonoidalPreCoComposeMorphism}(a, b, c)$.

1.14.101 AddMonoidalPreCoComposeMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ `AddMonoidalPreCoComposeMorphismWithGivenObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `MonoidalPreCoComposeMorphismWithGivenObjects`. $F : (s, a, b, c, r) \mapsto \text{MonoidalPreCoComposeMorphismWithGivenObjects}(s, a, b, c, r)$.

1.14.102 AddMorphismFromCoBidual (for IsCapCategory, IsFunction)

▷ `AddMorphismFromCoBidual(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `MorphismFromCoBidual`. $F : (a) \mapsto \text{MorphismFromCoBidual}(a)$.

1.14.103 AddMorphismFromCoBidualWithGivenCoBidual (for IsCapCategory, IsFunction)

▷ AddMorphismFromCoBidualWithGivenCoBidual(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromCoBidualWithGivenCoBidual. $F : (a, s) \mapsto \text{MorphismFromCoBidualWithGivenCoBidual}(a, s)$.

1.14.104 AddMorphismFromInternalCoHomToTensorProduct (for IsCapCategory, IsFunction)

▷ AddMorphismFromInternalCoHomToTensorProduct(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromInternalCoHomToTensorProduct. $F : (a, b) \mapsto \text{MorphismFromInternalCoHomToTensorProduct}(a, b)$.

1.14.105 AddMorphismFromInternalCoHomToTensorProductWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddMorphismFromInternalCoHomToTensorProductWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromInternalCoHomToTensorProductWithGivenObjects. $F : (s, a, b, r) \mapsto \text{MorphismFromInternalCoHomToTensorProductWithGivenObjects}(s, a, b, r)$.

1.14.106 AddTensorProductToInternalCoHomLeftAdjunctMorphism (for IsCapCategory, IsFunction)

▷ AddTensorProductToInternalCoHomLeftAdjunctMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToInternalCoHomLeftAdjunctMorphism. $F : (b, c, g) \mapsto \text{TensorProductToInternalCoHomLeftAdjunctMorphism}(b, c, g)$.

1.14.107 AddTensorProductToInternalCoHomLeftAdjunctMorphismWithGivenInternalCoHom (for IsCapCategory, IsFunction)

▷ AddTensorProductToInternalCoHomLeftAdjunctMorphismWithGivenInternalCoHom(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToInternalCoHomLeftAdjunctMorphismWithGivenInternalCoHom. $F : (b, c, g, i) \mapsto \text{TensorProductToInternalCoHomLeftAdjunctMorphismWithGivenInternalCoHom}(b, c, g, i)$.

1.14.108 AddTensorProductToInternalCoHomRightAdjunctMorphism (for IsCapCategory, IsFunction)

▷ AddTensorProductToInternalCoHomRightAdjunctMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToInternalCoHomRightAdjunctMorphism. $F : (b, c, g) \mapsto \text{TensorProductToInternalCoHomRightAdjunctMorphism}(b, c, g)$.

1.14.109 AddTensorProductToInternalCoHomRightAdjunctMorphismWithGivenInternalCoHom (for IsCapCategory, IsFunction)

▷ AddTensorProductToInternalCoHomRightAdjunctMorphismWithGivenInternalCoHom(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToInternalCoHomRightAdjunctMorphismWithGivenInternalCoHom. $F : (b, c, g, i) \mapsto \text{TensorProductToInternalCoHomRightAdjunctMorphismWithGivenInternalCoHom}(b, c, g, i)$.

1.14.110 AddUniversalPropertyOfCoDual (for IsCapCategory, IsFunction)

▷ AddUniversalPropertyOfCoDual(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation UniversalPropertyOfCoDual. $F : (t, a, \alpha) \mapsto \text{UniversalPropertyOfCoDual}(t, a, \alpha)$.

1.14.111 AddIsomorphismFromLeftDualObjectToLeftInternalHomIntoTensorUnit (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromLeftDualObjectToLeftInternalHomIntoTensorUnit(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation IsomorphismFromLeftDualObjectToLeftInternalHomIntoTensorUnit. $F : (a) \mapsto \text{IsomorphismFromLeftDualObjectToLeftInternalHomIntoTensorUnit}(a)$.

1.14.112 AddIsomorphismFromLeftInternalHomIntoTensorUnitToLeftDualObject (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromLeftInternalHomIntoTensorUnitToLeftDualObject(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation IsomorphismFromLeftInternalHomIntoTensorUnitToLeftDualObject. $F : (a) \mapsto \text{IsomorphismFromLeftInternalHomIntoTensorUnitToLeftDualObject}(a)$.

1.14.113 AddIsomorphismFromLeftInternalHomToObject (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromLeftInternalHomToObject(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation IsomorphismFromLeftInternalHomToObject. $F : (a) \mapsto \text{IsomorphismFromLeftInternalHomToObject}(a)$.

1.14.114 AddIsomorphismFromLeftInternalHomToObjectWithGivenLeftInternalHom (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromLeftInternalHomToObjectWithGivenLeftInternalHom(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation IsomorphismFromLeftInternalHomToObjectWithGivenLeftInternalHom. $F : (a, s) \mapsto \text{IsomorphismFromLeftInternalHomToObjectWithGivenLeftInternalHom}(a, s)$.

1.14.115 AddIsomorphismFromObjectToLeftInternalHom (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromObjectToLeftInternalHom(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation IsomorphismFromObjectToLeftInternalHom. $F : (a) \mapsto \text{IsomorphismFromObjectToLeftInternalHom}(a)$.

1.14.116 AddIsomorphismFromObjectToLeftInternalHomWithGivenLeftInternalHom (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromObjectToLeftInternalHomWithGivenLeftInternalHom(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation IsomorphismFromObjectToLeftInternalHomWithGivenLeftInternalHom. $F : (a, r) \mapsto \text{IsomorphismFromObjectToLeftInternalHomWithGivenLeftInternalHom}(a, r)$.

1.14.117 AddLeftClosedMonoidalCoevaluationMorphism (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalCoevaluationMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalCoevaluationMorphism. $F : (a, b) \mapsto \text{LeftClosedMonoidalCoevaluationMorphism}(a, b)$.

1.14.118 AddLeftClosedMonoidalCoevaluationMorphismWithGivenRange (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalCoevaluationMorphismWithGivenRange(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalCoevaluationMorphismWithGivenRange. $F : (a, b, r) \mapsto \text{LeftClosedMonoidalCoevaluationMorphismWithGivenRange}(a, b, r)$.

1.14.119 AddLeftClosedMonoidalEvaluationForLeftDual (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalEvaluationForLeftDual(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalEvaluationForLeftDual. $F : (a) \mapsto \text{LeftClosedMonoidalEvaluationForLeftDual}(a)$.

1.14.120 AddLeftClosedMonoidalEvaluationForLeftDualWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalEvaluationForLeftDualWithGivenTensorProduct(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalEvaluationForLeftDualWithGivenTensorProduct. $F : (s, a, r) \mapsto \text{LeftClosedMonoidalEvaluationForLeftDualWithGivenTensorProduct}(s, a, r)$.

1.14.121 AddLeftClosedMonoidalEvaluationMorphism (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalEvaluationMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalEvaluationMorphism. $F : (a, b) \mapsto \text{LeftClosedMonoidalEvaluationMorphism}(a, b)$.

1.14.122 AddLeftClosedMonoidalEvaluationMorphismWithGivenSource (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalEvaluationMorphismWithGivenSource(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalEvaluationMorphismWithGivenSource. $F : (a, b, s) \mapsto \text{LeftClosedMonoidalEvaluationMorphismWithGivenSource}(a, b, s)$.

1.14.123 AddLeftClosedMonoidalLambdaElimination (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalLambdaElimination(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalLambdaElimination. $F : (a, b, \alpha) \mapsto \text{LeftClosedMonoidalLambdaElimination}(a, b, \alpha)$.

1.14.124 AddLeftClosedMonoidalLambdaIntroduction (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalLambdaIntroduction(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalLambdaIntroduction. $F : (\alpha) \mapsto \text{LeftClosedMonoidalLambdaIntroduction}(\alpha)$.

1.14.125 AddLeftClosedMonoidalPostComposeMorphism (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalPostComposeMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalPostComposeMorphism. $F : (a, b, c) \mapsto \text{LeftClosedMonoidalPostComposeMorphism}(a, b, c)$.

1.14.126 AddLeftClosedMonoidalPostComposeMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalPostComposeMorphismWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalPostComposeMorphismWithGivenObjects. $F : (s, a, b, c, r) \mapsto \text{LeftClosedMonoidalPostComposeMorphismWithGivenObjects}(s, a, b, c, r)$.

1.14.127 AddLeftClosedMonoidalPreComposeMorphism (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalPreComposeMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalPreComposeMorphism. $F : (a, b, c) \mapsto \text{LeftClosedMonoidalPreComposeMorphism}(a, b, c)$.

1.14.128 AddLeftClosedMonoidalPreComposeMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddLeftClosedMonoidalPreComposeMorphismWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftClosedMonoidalPreComposeMorphismWithGivenObjects. $F : (s, a, b, c, r) \mapsto \text{LeftClosedMonoidalPreComposeMorphismWithGivenObjects}(s, a, b, c, r)$.

1.14.129 AddLeftDualOnMorphisms (for IsCapCategory, IsFunction)

▷ AddLeftDualOnMorphisms(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftDualOnMorphisms. $F : (alpha) \mapsto \text{LeftDualOnMorphisms}(alpha)$.

1.14.130 AddLeftDualOnMorphismsWithGivenLeftDUALS (for IsCapCategory, IsFunction)

▷ AddLeftDualOnMorphismsWithGivenLeftDUALS(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftDualOnMorphismsWithGivenLeftDUALS. $F : (s, alpha, r) \mapsto \text{LeftDualOnMorphismsWithGivenLeftDUALS}(s, alpha, r)$.

1.14.131 AddLeftDualOnObjects (for IsCapCategory, IsFunction)

▷ AddLeftDualOnObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftDualOnObjects. $F : (a) \mapsto \text{LeftDualOnObjects}(a)$.

1.14.132 AddLeftInternalHomOnMorphisms (for IsCapCategory, IsFunction)

▷ AddLeftInternalHomOnMorphisms(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftInternalHomOnMorphisms. $F : (alpha, beta) \mapsto \text{LeftInternalHomOnMorphisms}(alpha, beta)$.

1.14.133 AddLeftInternalHomOnMorphismsWithGivenLeftInternalHoms (for IsCapCategory, IsFunction)

▷ AddLeftInternalHomOnMorphismsWithGivenLeftInternalHoms(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation

`LeftInternalHomOnMorphismsWithGivenLeftInternalHoms`. $F : (s, \alpha, \beta, r) \mapsto \text{LeftInternalHomOnMorphismsWithGivenLeftInternalHoms}(s, \alpha, \beta, r)$.

1.14.134 `AddLeftInternalHomOnObjects` (for `IsCapCategory`, `IsFunction`)

▷ `AddLeftInternalHomOnObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftInternalHomOnObjects`. $F : (a, b) \mapsto \text{LeftInternalHomOnObjects}(a, b)$.

1.14.135 `AddLeftInternalHomToTensorProductAdjunctMorphism` (for `IsCapCategory`, `IsFunction`)

▷ `AddLeftInternalHomToTensorProductAdjunctMorphism(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftInternalHomToTensorProductAdjunctMorphism`. $F : (b, c, g) \mapsto \text{LeftInternalHomToTensorProductAdjunctMorphism}(b, c, g)$.

1.14.136 `AddLeftInternalHomToTensorProductAdjunctMorphismWithGivenTensorProduct` (for `IsCapCategory`, `IsFunction`)

▷ `AddLeftInternalHomToTensorProductAdjunctMorphismWithGivenTensorProduct(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftInternalHomToTensorProductAdjunctMorphismWithGivenTensorProduct`. $F : (b, c, g, t) \mapsto \text{LeftInternalHomToTensorProductAdjunctMorphismWithGivenTensorProduct}(b, c, g, t)$.

1.14.137 `AddMorphismFromTensorProductToLeftInternalHom` (for `IsCapCategory`, `IsFunction`)

▷ `AddMorphismFromTensorProductToLeftInternalHom(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `MorphismFromTensorProductToLeftInternalHom`. $F : (a, b) \mapsto \text{MorphismFromTensorProductToLeftInternalHom}(a, b)$.

1.14.138 `AddMorphismFromTensorProductToLeftInternalHomWithGivenObjects` (for `IsCapCategory`, `IsFunction`)

▷ `AddMorphismFromTensorProductToLeftInternalHomWithGivenObjects(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation

MorphismFromTensorProductToLeftInternalHomWithGivenObjects. $F : (s, a, b, r) \mapsto$
MorphismFromTensorProductToLeftInternalHomWithGivenObjects(s, a, b, r).

1.14.139 AddMorphismToLeftBidual (for IsCapCategory, IsFunction)

▷ AddMorphismToLeftBidual(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismToLeftBidual. $F : (a) \mapsto$ MorphismToLeftBidual(a).

1.14.140 AddMorphismToLeftBidualWithGivenLeftBidual (for IsCapCategory, IsFunction)

▷ AddMorphismToLeftBidualWithGivenLeftBidual(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismToLeftBidualWithGivenLeftBidual. $F : (a, r) \mapsto$ MorphismToLeftBidualWithGivenLeftBidual(a, r).

1.14.141 AddTensorProductLeftDualityCompatibilityMorphism (for IsCapCategory, IsFunction)

▷ AddTensorProductLeftDualityCompatibilityMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductLeftDualityCompatibilityMorphism. $F : (a, b) \mapsto$ TensorProductLeftDualityCompatibilityMorphism(a, b).

1.14.142 AddTensorProductLeftDualityCompatibilityMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddTensorProductLeftDualityCompatibilityMorphismWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductLeftDualityCompatibilityMorphismWithGivenObjects. $F : (s, a, b, r) \mapsto$ TensorProductLeftDualityCompatibilityMorphismWithGivenObjects(s, a, b, r).

1.14.143 AddTensorProductLeftInternalHomCompatibilityMorphism (for IsCapCategory, IsFunction)

▷ AddTensorProductLeftInternalHomCompatibilityMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductLeftInternalHomCompatibilityMorphism. $F : (list) \mapsto$ TensorProductLeftInternalHomCompatibilityMorphism($list$).

1.14.144 AddTensorProductLeftInternalHomCompatibilityMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddTensorProductLeftInternalHomCompatibilityMorphismWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductLeftInternalHomCompatibilityMorphismWithGivenObjects. $F : (source, list, range) \mapsto \text{TensorProductLeftInternalHomCompatibilityMorphismWithGivenObjects}(source, list, range)$.

1.14.145 AddTensorProductToLeftInternalHomAdjunctMorphism (for IsCapCategory, IsFunction)

▷ AddTensorProductToLeftInternalHomAdjunctMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToLeftInternalHomAdjunctMorphism. $F : (a, b, f) \mapsto \text{TensorProductToLeftInternalHomAdjunctMorphism}(a, b, f)$.

1.14.146 AddTensorProductToLeftInternalHomAdjunctMorphismWithGivenLeftInternalHom (for IsCapCategory, IsFunction)

▷ AddTensorProductToLeftInternalHomAdjunctMorphismWithGivenLeftInternalHom(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToLeftInternalHomAdjunctMorphismWithGivenLeftInternalHom. $F : (a, b, f, i) \mapsto \text{TensorProductToLeftInternalHomAdjunctMorphismWithGivenLeftInternalHom}(a, b, f, i)$.

1.14.147 AddUniversalPropertyOfLeftDual (for IsCapCategory, IsFunction)

▷ AddUniversalPropertyOfLeftDual(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation UniversalPropertyOfLeftDual. $F : (t, a, alpha) \mapsto \text{UniversalPropertyOfLeftDual}(t, a, alpha)$.

1.14.148 AddIsomorphismFromLeftCoDualObjectToLeftInternalCoHomFromTensorUnit (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromLeftCoDualObjectToLeftInternalCoHomFromTensorUnit(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromLeftCoDualObjectToLeftInternalCoHomFromTensorUnit`. $F : (a) \mapsto \text{IsomorphismFromLeftCoDualObjectToLeftInternalCoHomFromTensorUnit}(a)$.

1.14.149 AddIsomorphismFromLeftInternalCoHomFromTensorUnitToLeftCoDualObject (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromLeftInternalCoHomFromTensorUnitToLeftCoDualObject(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromLeftInternalCoHomFromTensorUnitToLeftCoDualObject`. $F : (a) \mapsto \text{IsomorphismFromLeftInternalCoHomFromTensorUnitToLeftCoDualObject}(a)$.

1.14.150 AddIsomorphismFromLeftInternalCoHomToObject (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromLeftInternalCoHomToObject(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromLeftInternalCoHomToObject`. $F : (a) \mapsto \text{IsomorphismFromLeftInternalCoHomToObject}(a)$.

1.14.151 AddIsomorphismFromLeftInternalCoHomToObjectWithGivenLeftInternalCoHom (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromLeftInternalCoHomToObjectWithGivenLeftInternalCoHom(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromLeftInternalCoHomToObjectWithGivenLeftInternalCoHom`. $F : (a, s) \mapsto \text{IsomorphismFromLeftInternalCoHomToObjectWithGivenLeftInternalCoHom}(a, s)$.

1.14.152 AddIsomorphismFromObjectToLeftInternalCoHom (for IsCapCategory, IsFunction)

▷ `AddIsomorphismFromObjectToLeftInternalCoHom(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `IsomorphismFromObjectToLeftInternalCoHom`. $F : (a) \mapsto \text{IsomorphismFromObjectToLeftInternalCoHom}(a)$.

1.14.153 AddIsomorphismFromObjectToLeftInternalCoHomWithGivenLeftInternalCoHom (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromObjectToLeftInternalCoHomWithGivenLeftInternalCoHom(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation IsomorphismFromObjectToLeftInternalCoHomWithGivenLeftInternalCoHom. $F : (a, r) \mapsto \text{IsomorphismFromObjectToLeftInternalCoHomWithGivenLeftInternalCoHom}(a, r)$.

1.14.154 AddLeftCoDualOnMorphisms (for IsCapCategory, IsFunction)

▷ AddLeftCoDualOnMorphisms(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftCoDualOnMorphisms. $F : (\alpha) \mapsto \text{LeftCoDualOnMorphisms}(\alpha)$.

1.14.155 AddLeftCoDualOnMorphismsWithGivenLeftCoDuals (for IsCapCategory, IsFunction)

▷ AddLeftCoDualOnMorphismsWithGivenLeftCoDuals(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftCoDualOnMorphismsWithGivenLeftCoDuals. $F : (s, \alpha, r) \mapsto \text{LeftCoDualOnMorphismsWithGivenLeftCoDuals}(s, \alpha, r)$.

1.14.156 AddLeftCoDualOnObjects (for IsCapCategory, IsFunction)

▷ AddLeftCoDualOnObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftCoDualOnObjects. $F : (a) \mapsto \text{LeftCoDualOnObjects}(a)$.

1.14.157 AddLeftCoDualityTensorProductCompatibilityMorphism (for IsCapCategory, IsFunction)

▷ AddLeftCoDualityTensorProductCompatibilityMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftCoDualityTensorProductCompatibilityMorphism. $F : (a, b) \mapsto \text{LeftCoDualityTensorProductCompatibilityMorphism}(a, b)$.

1.14.158 AddLeftCoDualityTensorProductCompatibilityMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddLeftCoDualityTensorProductCompatibilityMorphismWithGivenObjects(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftCoDualityTensorProductCompatibilityMorphismWithGivenObjects. $F : (s, a, b, r) \mapsto \text{LeftCoDualityTensorProductCompatibilityMorphismWithGivenObjects}(s, a, b, r)$.

1.14.159 AddLeftCoclosedMonoidalCoevaluationMorphism (for IsCapCategory, IsFunction)

▷ AddLeftCoclosedMonoidalCoevaluationMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftCoclosedMonoidalCoevaluationMorphism. $F : (a, b) \mapsto \text{LeftCoclosedMonoidalCoevaluationMorphism}(a, b)$.

1.14.160 AddLeftCoclosedMonoidalCoevaluationMorphismWithGivenSource (for IsCapCategory, IsFunction)

▷ AddLeftCoclosedMonoidalCoevaluationMorphismWithGivenSource(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftCoclosedMonoidalCoevaluationMorphismWithGivenSource. $F : (a, b, s) \mapsto \text{LeftCoclosedMonoidalCoevaluationMorphismWithGivenSource}(a, b, s)$.

1.14.161 AddLeftCoclosedMonoidalEvaluationForLeftCoDual (for IsCapCategory, IsFunction)

▷ AddLeftCoclosedMonoidalEvaluationForLeftCoDual(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftCoclosedMonoidalEvaluationForLeftCoDual. $F : (a) \mapsto \text{LeftCoclosedMonoidalEvaluationForLeftCoDual}(a)$.

1.14.162 AddLeftCoclosedMonoidalEvaluationForLeftCoDualWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddLeftCoclosedMonoidalEvaluationForLeftCoDualWithGivenTensorProduct(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation

`LeftCoclosedMonoidalEvaluationForLeftCoDualWithGivenTensorProduct`. $F : (s, a, r) \mapsto \text{LeftCoclosedMonoidalEvaluationForLeftCoDualWithGivenTensorProduct}(s, a, r)$.

1.14.163 **AddLeftCoclosedMonoidalEvaluationMorphism (for IsCapCategory, IsFunction)**

▷ `AddLeftCoclosedMonoidalEvaluationMorphism(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftCoclosedMonoidalEvaluationMorphism`. $F : (a, b) \mapsto \text{LeftCoclosedMonoidalEvaluationMorphism}(a, b)$.

1.14.164 **AddLeftCoclosedMonoidalEvaluationMorphismWithGivenRange (for IsCapCategory, IsFunction)**

▷ `AddLeftCoclosedMonoidalEvaluationMorphismWithGivenRange(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftCoclosedMonoidalEvaluationMorphismWithGivenRange`. $F : (a, b, r) \mapsto \text{LeftCoclosedMonoidalEvaluationMorphismWithGivenRange}(a, b, r)$.

1.14.165 **AddLeftCoclosedMonoidalLambdaElimination (for IsCapCategory, IsFunction)**

▷ `AddLeftCoclosedMonoidalLambdaElimination(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftCoclosedMonoidalLambdaElimination`. $F : (a, b, \alpha) \mapsto \text{LeftCoclosedMonoidalLambdaElimination}(a, b, \alpha)$.

1.14.166 **AddLeftCoclosedMonoidalLambdaIntroduction (for IsCapCategory, IsFunction)**

▷ `AddLeftCoclosedMonoidalLambdaIntroduction(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftCoclosedMonoidalLambdaIntroduction`. $F : (\alpha) \mapsto \text{LeftCoclosedMonoidalLambdaIntroduction}(\alpha)$.

1.14.167 **AddLeftCoclosedMonoidalPostCoComposeMorphism (for IsCapCategory, IsFunction)**

▷ `AddLeftCoclosedMonoidalPostCoComposeMorphism(C, F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftCoclosedMonoidalPostCoComposeMorphism`. $F : (a, b, c) \mapsto \text{LeftCoclosedMonoidalPostCoComposeMorphism}(a, b, c)$.

1.14.168 AddLeftCoclosedMonoidalPostCoComposeMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddLeftCoclosedMonoidalPostCoComposeMorphismWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftCoclosedMonoidalPostCoComposeMorphismWithGivenObjects. $F : (s, a, b, c, r) \mapsto \text{LeftCoclosedMonoidalPostCoComposeMorphismWithGivenObjects}(s, a, b, c, r)$.

1.14.169 AddLeftCoclosedMonoidalPreCoComposeMorphism (for IsCapCategory, IsFunction)

▷ AddLeftCoclosedMonoidalPreCoComposeMorphism(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftCoclosedMonoidalPreCoComposeMorphism. $F : (a, b, c) \mapsto \text{LeftCoclosedMonoidalPreCoComposeMorphism}(a, b, c)$.

1.14.170 AddLeftCoclosedMonoidalPreCoComposeMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddLeftCoclosedMonoidalPreCoComposeMorphismWithGivenObjects(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftCoclosedMonoidalPreCoComposeMorphismWithGivenObjects. $F : (s, a, b, c, r) \mapsto \text{LeftCoclosedMonoidalPreCoComposeMorphismWithGivenObjects}(s, a, b, c, r)$.

1.14.171 AddLeftInternalCoHomOnMorphisms (for IsCapCategory, IsFunction)

▷ AddLeftInternalCoHomOnMorphisms(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftInternalCoHomOnMorphisms. $F : (\alpha, \beta) \mapsto \text{LeftInternalCoHomOnMorphisms}(\alpha, \beta)$.

1.14.172 AddLeftInternalCoHomOnMorphismsWithGivenLeftInternalCoHoms (for IsCapCategory, IsFunction)

▷ AddLeftInternalCoHomOnMorphismsWithGivenLeftInternalCoHoms(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftInternalCoHomOnMorphismsWithGivenLeftInternalCoHoms. $F : (s, \alpha, \beta, r) \mapsto \text{LeftInternalCoHomOnMorphismsWithGivenLeftInternalCoHoms}(s, \alpha, \beta, r)$.

1.14.173 AddLeftInternalCoHomOnObjects (for IsCapCategory, IsFunction)

▷ `AddLeftInternalCoHomOnObjects(C , F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftInternalCoHomOnObjects`. $F : (a, b) \mapsto \text{LeftInternalCoHomOnObjects}(a, b)$.

1.14.174 AddLeftInternalCoHomTensorProductCompatibilityMorphism (for IsCapCategory, IsFunction)

▷ `AddLeftInternalCoHomTensorProductCompatibilityMorphism(C , F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftInternalCoHomTensorProductCompatibilityMorphism`. $F : (list) \mapsto \text{LeftInternalCoHomTensorProductCompatibilityMorphism}(list)$.

1.14.175 AddLeftInternalCoHomTensorProductCompatibilityMorphismWithGivenObjects (for IsCapCategory, IsFunction)

▷ `AddLeftInternalCoHomTensorProductCompatibilityMorphismWithGivenObjects(C , F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftInternalCoHomTensorProductCompatibilityMorphismWithGivenObjects`. $F : (source, list, range) \mapsto \text{LeftInternalCoHomTensorProductCompatibilityMorphismWithGivenObjects}(source, list, range)$.

1.14.176 AddLeftInternalCoHomToTensorProductAdjunctMorphism (for IsCapCategory, IsFunction)

▷ `AddLeftInternalCoHomToTensorProductAdjunctMorphism(C , F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftInternalCoHomToTensorProductAdjunctMorphism`. $F : (a, c, f) \mapsto \text{LeftInternalCoHomToTensorProductAdjunctMorphism}(a, c, f)$.

1.14.177 AddLeftInternalCoHomToTensorProductAdjunctMorphismWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ `AddLeftInternalCoHomToTensorProductAdjunctMorphismWithGivenTensorProduct(C , F)` (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation `LeftInternalCoHomToTensorProductAdjunctMorphismWithGivenTensorProduct`. $F :$

$(a, c, f, t) \mapsto \text{LeftInternalCoHomToTensorProductAdjunctMorphismWithGivenTensorProduct}(a, c, f, t).$

1.14.178 AddMorphismFromLeftCoBidual (for IsCapCategory, IsFunction)

▷ AddMorphismFromLeftCoBidual(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromLeftCoBidual. $F : (a) \mapsto \text{MorphismFromLeftCoBidual}(a).$

1.14.179 AddMorphismFromLeftCoBidualWithGivenLeftCoBidual (for IsCapCategory, IsFunction)

▷ AddMorphismFromLeftCoBidualWithGivenLeftCoBidual(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromLeftCoBidualWithGivenLeftCoBidual. $F : (a, s) \mapsto \text{MorphismFromLeftCoBidualWithGivenLeftCoBidual}(a, s).$

1.14.180 AddMorphismFromLeftInternalCoHomToTensorProduct (for IsCapCategory, IsFunction)

▷ AddMorphismFromLeftInternalCoHomToTensorProduct(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromLeftInternalCoHomToTensorProduct. $F : (a, b) \mapsto \text{MorphismFromLeftInternalCoHomToTensorProduct}(a, b).$

1.14.181 AddMorphismFromLeftInternalCoHomToTensorProductWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddMorphismFromLeftInternalCoHomToTensorProductWithGivenObjects(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromLeftInternalCoHomToTensorProductWithGivenObjects. $F : (s, a, b, r) \mapsto \text{MorphismFromLeftInternalCoHomToTensorProductWithGivenObjects}(s, a, b, r).$

1.14.182 AddTensorProductToLeftInternalCoHomAdjunctMorphism (for IsCapCategory, IsFunction)

▷ AddTensorProductToLeftInternalCoHomAdjunctMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToLeftInternalCoHomAdjunctMorphism. $F : (b, c, g) \mapsto \text{TensorProductToLeftInternalCoHomAdjunctMorphism}(b, c, g).$

1.14.183 AddTensorProductToLeftInternalCoHomAdjunctMorphismWithGivenLeftInternalCoHom (for IsCapCategory, IsFunction)

▷ AddTensorProductToLeftInternalCoHomAdjunctMorphismWithGivenLeftInternalCoHom(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductToLeftInternalCoHomAdjunctMorphismWithGivenLeftInternalCoHom. $F : (b, c, g, i) \mapsto \text{TensorProductToLeftInternalCoHomAdjunctMorphismWithGivenLeftInternalCoHom}(b, c, g, i)$.

1.14.184 AddUniversalPropertyOfLeftCoDual (for IsCapCategory, IsFunction)

▷ AddUniversalPropertyOfLeftCoDual(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation UniversalPropertyOfLeftCoDual. $F : (t, a, \alpha) \mapsto \text{UniversalPropertyOfLeftCoDual}(t, a, \alpha)$.

1.14.185 AddAssociatorLeftToRight (for IsCapCategory, IsFunction)

▷ AddAssociatorLeftToRight(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation AssociatorLeftToRight. $F : (a, b, c) \mapsto \text{AssociatorLeftToRight}(a, b, c)$.

1.14.186 AddAssociatorLeftToRightWithGivenTensorProducts (for IsCapCategory, IsFunction)

▷ AddAssociatorLeftToRightWithGivenTensorProducts(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation AssociatorLeftToRightWithGivenTensorProducts. $F : (s, a, b, c, r) \mapsto \text{AssociatorLeftToRightWithGivenTensorProducts}(s, a, b, c, r)$.

1.14.187 AddAssociatorRightToLeft (for IsCapCategory, IsFunction)

▷ AddAssociatorRightToLeft(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation AssociatorRightToLeft. $F : (a, b, c) \mapsto \text{AssociatorRightToLeft}(a, b, c)$.

1.14.188 AddAssociatorRightToLeftWithGivenTensorProducts (for IsCapCategory, IsFunction)

▷ AddAssociatorRightToLeftWithGivenTensorProducts(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation AssociatorRightToLeftWithGivenTensorProducts. $F : (s, a, b, c, r) \mapsto \text{AssociatorRightToLeftWithGivenTensorProducts}(s, a, b, c, r)$.

1.14.189 AddLeftUnitor (for IsCapCategory, IsFunction)

▷ AddLeftUnitor(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftUnitor. $F : (a) \mapsto \text{LeftUnitor}(a)$.

1.14.190 AddLeftUnitorInverse (for IsCapCategory, IsFunction)

▷ AddLeftUnitorInverse(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftUnitorInverse. $F : (a) \mapsto \text{LeftUnitorInverse}(a)$.

1.14.191 AddLeftUnitorInverseWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddLeftUnitorInverseWithGivenTensorProduct(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftUnitorInverseWithGivenTensorProduct. $F : (a, r) \mapsto \text{LeftUnitorInverseWithGivenTensorProduct}(a, r)$.

1.14.192 AddLeftUnitorWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddLeftUnitorWithGivenTensorProduct(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation LeftUnitorWithGivenTensorProduct. $F : (a, s) \mapsto \text{LeftUnitorWithGivenTensorProduct}(a, s)$.

1.14.193 AddRightUnitor (for IsCapCategory, IsFunction)

▷ AddRightUnitor(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation RightUnitor. $F : (a) \mapsto \text{RightUnitor}(a)$.

1.14.194 AddRightUnitorInverse (for IsCapCategory, IsFunction)

▷ AddRightUnitorInverse(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation RightUnitorInverse. $F : (a) \mapsto \text{RightUnitorInverse}(a)$.

1.14.195 AddRightUnitorInverseWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddRightUnitorInverseWithGivenTensorProduct(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation RightUnitorInverseWithGivenTensorProduct. $F : (a, r) \mapsto \text{RightUnitorInverseWithGivenTensorProduct}(a, r)$.

1.14.196 AddRightUnitorWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddRightUnitorWithGivenTensorProduct(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation RightUnitorWithGivenTensorProduct. $F : (a, s) \mapsto \text{RightUnitorWithGivenTensorProduct}(a, s)$.

1.14.197 AddTensorProductOnMorphisms (for IsCapCategory, IsFunction)

▷ AddTensorProductOnMorphisms(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductOnMorphisms. $F : (\alpha, \beta) \mapsto \text{TensorProductOnMorphisms}(\alpha, \beta)$.

1.14.198 AddTensorProductOnMorphismsWithGivenTensorProducts (for IsCapCategory, IsFunction)

▷ AddTensorProductOnMorphismsWithGivenTensorProducts(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductOnMorphismsWithGivenTensorProducts. $F : (s, \alpha, \beta, r) \mapsto \text{TensorProductOnMorphismsWithGivenTensorProducts}(s, \alpha, \beta, r)$.

1.14.199 AddCoevaluationForDual (for IsCapCategory, IsFunction)

▷ AddCoevaluationForDual(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoevaluationForDual. $F : (a) \mapsto \text{CoevaluationForDual}(a)$.

1.14.200 AddCoevaluationForDualWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddCoevaluationForDualWithGivenTensorProduct(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoevaluationForDualWithGivenTensorProduct. $F : (s, a, r) \mapsto \text{CoevaluationForDualWithGivenTensorProduct}(s, a, r)$.

1.14.201 AddIsomorphismFromInternalHomToTensorProductWithDualObject (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromInternalHomToTensorProductWithDualObject(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation IsomorphismFromInternalHomToTensorProductWithDualObject. $F : (a, b) \mapsto \text{IsomorphismFromInternalHomToTensorProductWithDualObject}(a, b)$.

1.14.202 AddIsomorphismFromTensorProductWithDualObjectToInternalHom (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromTensorProductWithDualObjectToInternalHom(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation IsomorphismFromTensorProductWithDualObjectToInternalHom. $F : (a, b) \mapsto \text{IsomorphismFromTensorProductWithDualObjectToInternalHom}(a, b)$.

1.14.203 AddMorphismFromBidual (for IsCapCategory, IsFunction)

▷ AddMorphismFromBidual(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromBidual. $F : (a) \mapsto \text{MorphismFromBidual}(a)$.

1.14.204 AddMorphismFromBidualWithGivenBidual (for IsCapCategory, IsFunction)

▷ AddMorphismFromBidualWithGivenBidual(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromBidualWithGivenBidual. $F : (a, s) \mapsto \text{MorphismFromBidualWithGivenBidual}(a, s)$.

1.14.205 AddMorphismFromInternalHomToTensorProduct (for IsCapCategory, IsFunction)

▷ AddMorphismFromInternalHomToTensorProduct(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromInternalHomToTensorProduct. $F : (a, b) \mapsto \text{MorphismFromInternalHomToTensorProduct}(a, b)$.

1.14.206 AddMorphismFromInternalHomToTensorProductWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddMorphismFromInternalHomToTensorProductWithGivenObjects(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromInternalHomToTensorProductWithGivenObjects. $F : (s, a, b, r) \mapsto \text{MorphismFromInternalHomToTensorProductWithGivenObjects}(s, a, b, r)$.

1.14.207 AddRankMorphism (for IsCapCategory, IsFunction)

▷ AddRankMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation RankMorphism. $F : (a) \mapsto \text{RankMorphism}(a)$.

1.14.208 AddTensorProductInternalHomCompatibilityMorphismInverse (for IsCapCategory, IsFunction)

▷ AddTensorProductInternalHomCompatibilityMorphismInverse(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductInternalHomCompatibilityMorphismInverse. $F : (list) \mapsto \text{TensorProductInternalHomCompatibilityMorphismInverse}(list)$.

1.14.209 AddTensorProductInternalHomCompatibilityMorphismInverseWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddTensorProductInternalHomCompatibilityMorphismInverseWithGivenObjects(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TensorProductInternalHomCompatibilityMorphismInverseWithGivenObjects. $F : (source, list, range) \mapsto \text{TensorProductInternalHomCompatibilityMorphismInverseWithGivenObjects}(source, list, range)$.

1.14.210 AddTraceMap (for IsCapCategory, IsFunction)

▷ AddTraceMap(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation TraceMap. $F : (alpha) \mapsto \text{TraceMap}(alpha)$.

1.14.211 AddCoRankMorphism (for IsCapCategory, IsFunction)

▷ AddCoRankMorphism(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoRankMorphism. $F : (a) \mapsto \text{CoRankMorphism}(a)$.

1.14.212 AddCoTraceMap (for IsCapCategory, IsFunction)

▷ AddCoTraceMap(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoTraceMap. $F : (alpha) \mapsto \text{CoTraceMap}(alpha)$.

1.14.213 AddCoclosedCoevaluationForCoDual (for IsCapCategory, IsFunction)

▷ AddCoclosedCoevaluationForCoDual(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedCoevaluationForCoDual. $F : (a) \mapsto \text{CoclosedCoevaluationForCoDual}(a)$.

1.14.214 AddCoclosedCoevaluationForCoDualWithGivenTensorProduct (for IsCapCategory, IsFunction)

▷ AddCoclosedCoevaluationForCoDualWithGivenTensorProduct(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation CoclosedCoevaluationForCoDualWithGivenTensorProduct. $F : (s, a, r) \mapsto \text{CoclosedCoevaluationForCoDualWithGivenTensorProduct}(s, a, r)$.

1.14.215 AddInternalCoHomTensorProductCompatibilityMorphismInverse (for IsCapCategory, IsFunction)

▷ AddInternalCoHomTensorProductCompatibilityMorphismInverse(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalCoHomTensorProductCompatibilityMorphismInverse. $F : (list) \mapsto \text{InternalCoHomTensorProductCompatibilityMorphismInverse}(list)$.

1.14.216 AddInternalCoHomTensorProductCompatibilityMorphismInverseWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddInternalCoHomTensorProductCompatibilityMorphismInverseWithGivenObjects(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation InternalCoHomTensorProductCompatibilityMorphismInverseWithGivenObjects. $F : (source, list, range) \mapsto \text{InternalCoHomTensorProductCompatibilityMorphismInverseWithGivenObjects}(source, list, range)$

1.14.217 AddIsomorphismFromInternalCoHomToTensorProductWithCoDualObject (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromInternalCoHomToTensorProductWithCoDualObject(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation IsomorphismFromInternalCoHomToTensorProductWithCoDualObject. $F : (a, b) \mapsto \text{IsomorphismFromInternalCoHomToTensorProductWithCoDualObject}(a, b)$.

1.14.218 AddIsomorphismFromTensorProductWithCoDualObjectToInternalCoHom (for IsCapCategory, IsFunction)

▷ AddIsomorphismFromTensorProductWithCoDualObjectToInternalCoHom(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation IsomorphismFromTensorProductWithCoDualObjectToInternalCoHom. $F : (a, b) \mapsto \text{IsomorphismFromTensorProductWithCoDualObjectToInternalCoHom}(a, b)$.

1.14.219 AddMorphismFromTensorProductToInternalCoHom (for IsCapCategory, IsFunction)

▷ AddMorphismFromTensorProductToInternalCoHom(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismFromTensorProductToInternalCoHom. $F : (a, b) \mapsto \text{MorphismFromTensorProductToInternalCoHom}(a, b)$.

1.14.220 AddMorphismFromTensorProductToInternalCoHomWithGivenObjects (for IsCapCategory, IsFunction)

▷ AddMorphismFromTensorProductToInternalCoHomWithGivenObjects(C , F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation

MorphismFromTensorProductToInternalCoHomWithGivenObjects. $F : (s, a, b, r) \mapsto$
MorphismFromTensorProductToInternalCoHomWithGivenObjects(s, a, b, r).

1.14.221 AddMorphismToCoBidual (for IsCapCategory, IsFunction)

▷ AddMorphismToCoBidual(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismToCoBidual. $F : (a) \mapsto \text{MorphismToCoBidual}(a)$.

1.14.222 AddMorphismToCoBidualWithGivenCoBidual (for IsCapCategory, IsFunction)

▷ AddMorphismToCoBidualWithGivenCoBidual(C, F) (operation)

Returns: nothing

The arguments are a category C and a function F . This operation adds the given function F to the category for the basic operation MorphismToCoBidualWithGivenCoBidual. $F : (a, r) \mapsto \text{MorphismToCoBidualWithGivenCoBidual}(a, r)$.

Chapter 2

Examples and Tests

2.1 Test functions

2.1.1 AdditiveMonoidalCategoriesTest

▷ `AdditiveMonoidalCategoriesTest(cat, a, L)` (function)

The arguments are

- a CAP category *cat*
- an object *a*
- a list *L* of objects

This function checks for every operation declared in `AdditiveMonoidalCategories.gd` if it is computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.2 BraidedMonoidalCategoriesTest

▷ `BraidedMonoidalCategoriesTest(cat, a, b)` (function)

The arguments are

- a CAP category *cat*
- objects *a, b*

This function checks for every operation declared in `BraidedMonoidalCategories.gd` if it is computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.3 ClosedMonoidalCategoriesTest

▷ `ClosedMonoidalCategoriesTest(cat, a, b, c, d, alpha, beta, gamma, delta, epsilon, zeta)` (function)

The arguments are

- a CAP category *cat*
- objects a, b, c, d
- a morphism $\alpha : a \rightarrow b$
- a morphism $\beta : c \rightarrow d$
- a morphism $\gamma : a \otimes b \rightarrow 1$
- a morphism $\delta : c \otimes d \rightarrow 1$
- a morphism $\varepsilon : 1 \rightarrow \text{Hom}(a, b)$
- a morphism $\zeta : 1 \rightarrow \text{Hom}(c, d)$

This function checks for every operation declared in `ClosedMonoidalCategories.gd` if it is computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.4 ClosedMonoidalCategoriesTestWithGiven

▷ `ClosedMonoidalCategoriesTestWithGiven(cat, a, b, c, d, alpha, beta)` (function)

The arguments are

- a CAP category *cat*
- objects a, b, c, d
- a morphism $\alpha : a \rightarrow b$

- a morphism $\beta : c \rightarrow d$

This function checks for some `*WithGiven` operations declared in `ClosedMonoidalCategories.gd` if they are computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.5 CoclosedMonoidalCategoriesTest

▷ `CoclosedMonoidalCategoriesTest(cat, a, b, c, d, alpha, beta, gamma, delta, epsilon, zeta)` (function)

The arguments are

- a CAP category *cat*
- objects *a, b, c, d*
- a morphism $\alpha : a \rightarrow b$
- a morphism $\beta : c \rightarrow d$
- a morphism $\gamma : 1 \rightarrow a \otimes b$
- a morphism $\delta : 1 \rightarrow c \otimes d$
- a morphism $\varepsilon : \text{coHom}(a, b) \rightarrow 1$
- a morphism $\zeta : \text{coHom}(c, d) \rightarrow 1$

This function checks for every operation declared in `CoclosedMonoidalCategories.gd` if it is computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.6 CoclosedMonoidalCategoriesTestWithGiven

▷ `CoclosedMonoidalCategoriesTestWithGiven(cat, a, b, c, d, alpha, beta, gamma, delta, epsilon, zeta)` (function)

The arguments are

- a CAP category *cat*
- objects a, b, c, d
- a morphism $\alpha : a \rightarrow b$
- a morphism $\beta : c \rightarrow d$

This function checks for some `*WithGiven` operations declared in `CoclosedMonoidalCategories.gd` if they are computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.7 LeftClosedMonoidalCategoriesTest

▷ `LeftClosedMonoidalCategoriesTest(cat, a, b, c, d, alpha, beta, gamma, delta, epsilon, zeta)` (function)

The arguments are

- a CAP category *cat*
- objects a, b, c, d
- a morphism $\alpha : a \rightarrow b$
- a morphism $\beta : c \rightarrow d$
- a morphism $\gamma : a \otimes b \rightarrow 1$
- a morphism $\delta : c \otimes d \rightarrow 1$
- a morphism $\varepsilon : 1 \rightarrow \text{Hom}(a, b)$
- a morphism $\zeta : 1 \rightarrow \text{Hom}(c, d)$

This function checks for every operation declared in `LeftClosedMonoidalCategories.gd` if it is computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.8 LeftClosedMonoidalCategoriesTestWithGiven

▷ `LeftClosedMonoidalCategoriesTestWithGiven(cat, a, b, c, d, alpha, beta, gamma, delta, epsilon, zeta)` (function)

The arguments are

- a CAP category *cat*
- objects *a, b, c, d*
- a morphism $\alpha : a \rightarrow b$
- a morphism $\beta : c \rightarrow d$

This function checks for some `*WithGiven` operationS declared in `LeftClosedMonoidalCategories.gd` if they are computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.9 LeftCoclosedMonoidalCategoriesTest

▷ `LeftCoclosedMonoidalCategoriesTest(cat, a, b, c, d, alpha, beta, gamma, delta, epsilon, zeta)` (function)

The arguments are

- a CAP category *cat*
- objects *a, b, c, d*
- a morphism $\alpha : a \rightarrow b$
- a morphism $\beta : c \rightarrow d$
- a morphism $\gamma : 1 \rightarrow a \otimes b$
- a morphism $\delta : 1 \rightarrow c \otimes d$

- a morphism $\varepsilon : \text{coHom}(a, b) \rightarrow 1$
- a morphism $\zeta : \text{coHom}(c, d) \rightarrow 1$

This function checks for every operation declared in `LeftCoclosedMonoidalCategories.gd` if it is computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.10 LeftCoclosedMonoidalCategoriesTestWithGiven

▷ `LeftCoclosedMonoidalCategoriesTestWithGiven(cat, a, b, c, d, alpha, beta, gamma, delta, epsilon, zeta)` (function)

The arguments are

- a CAP category *cat*
- objects *a, b, c, d*
- a morphism $\alpha : a \rightarrow b$
- a morphism $\beta : c \rightarrow d$

This function checks for some `*WithGiven` operations declared in `LeftCoclosedMonoidalCategories.gd` if they are computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.11 MonoidalCategoriesTensorProductAndUnitTest

▷ `MonoidalCategoriesTensorProductAndUnitTest(cat, a, b)` (function)

The arguments are

- a CAP category *cat*
- objects *a, b*

This function checks for every operation declared in `MonoidalCategoriesTensorProductAndUnit.gd` if it is computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.12 MonoidalCategoriesTest

▷ `MonoidalCategoriesTest(cat, a, b, c, alpha, beta)` (function)

The arguments are

- a CAP category *cat*
- objects *a, b, c*
- a morphism $\alpha : a \rightarrow b$
- a morphism $\beta : c \rightarrow d$

This function checks for every operation declared in `MonoidalCategories.gd` if it is computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.13 RigidSymmetricClosedMonoidalCategoriesTest

▷ `RigidSymmetricClosedMonoidalCategoriesTest(cat, a, b, c, d, alpha)` (function)

The arguments are

- a CAP category *cat*
- objects *a, b, c, d*
- an endomorphism $\alpha : a \rightarrow a$

This function checks for every object and morphism declared in `RigidSymmetricClosedMonoidalCategories.gd` if it is computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.1.14 RigidSymmetricCoclosedMonoidalCategoriesTest

▷ `RigidSymmetricCoclosedMonoidalCategoriesTest(cat, a, b, c, d, alpha)` (function)

The arguments are

- a CAP category *cat*
- objects *a, b, c, d*
- an endomorphism $\alpha : a \rightarrow a$

This function checks for every object and morphism declared in `RigidSymmetricCoclosedMonoidalCategories.gd` if it is computable in the CAP category *cat*. If yes, then the operation is executed with the parameters given above and compared to the equivalent computation in the opposite category of *cat*. Pass the options

- `verbose := true` to output more information.
- `only_primitive_operations := true`, which is passed on to `Opposite()`, to only primitively install dual operations for primitively installed operations in *cat*. The advantage is, that more derivations might be tested. On the downside, this might test fewer `dual_pre/postprocessor_funcs`.

2.2 Basics

Example

```
gap> LoadPackage( "MonoidalCategories", false );
true
gap> LoadPackage( "LinearAlgebraForCAP", false );
true
gap> Q := HomalgFieldOfRationals();
Q
gap> a := VectorSpaceObject( 1, Q );
<A vector space object over Q of dimension 1>
gap> b := VectorSpaceObject( 2, Q );
<A vector space object over Q of dimension 2>
gap> c := VectorSpaceObject( 3, Q );
<A vector space object over Q of dimension 3>
gap> z := ZeroObject( CapCategory( a ) );
<A vector space object over Q of dimension 0>
gap> alpha := VectorSpaceMorphism( a, [ [ 1, 0 ] ], b );
<A morphism in Category of matrices over Q>
gap> beta := VectorSpaceMorphism( b,
> [ [ 1, 0, 0 ], [ 0, 1, 0 ] ], c );
```

```

<A morphism in Category of matrices over Q>
gap> gamma := VectorSpaceMorphism( c,
>      [ [ 0, 1, 1 ], [ 1, 0, 1 ], [ 1, 1, 0 ] ], c );
<A morphism in Category of matrices over Q>
gap> IsCongruentForMorphisms(
>      TensorProductOnMorphisms( alpha, beta ),
>      TensorProductOnMorphisms( beta, alpha ) );
false
gap> IsOne( AssociatorRightToLeft( a, b, c ) );
true
gap> IsCongruentForMorphisms(
>      gamma, LambdaElimination( c, c, LambdaIntroduction( gamma ) ) );
true
gap> IsZero( TraceMap( gamma ) );
true
gap> IsCongruentForMorphisms(
>      RankMorphism( DirectSum( a, b ) ), RankMorphism( c ) );
true
gap> IsOne( Braiding( b, c ) );
false
gap> IsOne( PreCompose( Braiding( b, c ), Braiding( c, b ) ) );
true

```

Chapter 3

Code Generation for Monoidal Categories

3.1 Monoidal Categories

3.1.1 WriteFileForMonoidalStructure

▷ WriteFileForMonoidalStructure(*key_val_rec*, *package_name*, *files_rec*) (function)

Returns: nothing

This functions uses the dictionary *key_val_rec* to create a new monoidal structure. It generates the necessary files in the package *package_name* using the file-correspondence table *files_rec*. See the implementation for details.

3.2 Closed Monoidal Categories

3.2.1 WriteFileForClosedMonoidalStructure

▷ WriteFileForClosedMonoidalStructure(*key_val_rec*, *package_name*, *files_rec*) (function)

Returns: nothing

This functions uses the dictionary *key_val_rec* to create a new closed monoidal structure. It generates the necessary files in the package *package_name* using the file-correspondence table *files_rec*. See the implementation for details.

3.2.2 WriteFileForLeftClosedMonoidalStructure

▷ WriteFileForLeftClosedMonoidalStructure(*key_val_rec*, *package_name*, *files_rec*) (function)

Returns: nothing

This functions uses the dictionary *key_val_rec* to create a new left closed monoidal structure. It generates the necessary files in the package *package_name* using the file-correspondence table *files_rec*. See the implementation for details.

3.3 Coclosed Monoidal Categories

3.3.1 WriteFileForCoclosedMonoidalStructure

▷ `WriteFileForCoclosedMonoidalStructure(key_val_rec, package_name, files_rec)`
(function)

Returns: nothing

This functions uses the dictionary `key_val_rec` to create a new coclosed monoidal structure. It generates the necessary files in the package `package_name` using the file-correspondence table `files_rec`. See the implementation for details.

3.3.2 WriteFileForLeftCoclosedMonoidalStructure

▷ `WriteFileForLeftCoclosedMonoidalStructure(key_val_rec, package_name, files_rec)`
(function)

Returns: nothing

This functions uses the dictionary `key_val_rec` to create a new left coclosed monoidal structure. It generates the necessary files in the package `package_name` using the file-correspondence table `files_rec`. See the implementation for details.

Chapter 4

The terminal category with multiple objects

This is an example of a category which is created using `CategoryConstructor` out of no input.

This category “lies” in all doctrines and can hence be used (in conjunction with `LazyCategory`) in order to check the type-correctness of the various derived methods provided by `CAP` or any `CAP`-based package.

4.1 Constructors

4.2 `GAP` Categories

Chapter 5

Legacy Operations and Synonyms

5.1 Legacy operations

5.1.1 `CoclosedCoevaluationMorphism` (for `IsCapCategoryObject`, `IsCapCategoryObject`)

▷ `CoclosedCoevaluationMorphism(a, b)` (operation)

This is a legacy operation for `CoclosedMonoidalLeftCoevaluationMorphism(b, a)`, i.e., with the first and second argument interchanged.

5.1.2 `CoclosedCoevaluationMorphismWithGivenSource` (for `IsCapCategoryObject`, `IsCapCategoryObject`, `IsCapCategoryObject`)

▷ `CoclosedCoevaluationMorphismWithGivenSource(a, b, s)` (operation)

This is a legacy operation for `CoclosedMonoidalLeftCoevaluationMorphismWithGivenSource(b, a, s)`, i.e., with the first and second argument interchanged.

5.1.3 `CoclosedEvaluationMorphism` (for `IsCapCategoryObject`, `IsCapCategoryObject`)

▷ `CoclosedEvaluationMorphism(a, b)` (operation)

This is a legacy operation for `CoclosedMonoidalLeftEvaluationMorphism(b, a)`, i.e., with the first and second argument interchanged.

5.1.4 `CoclosedEvaluationMorphismWithGivenRange` (for `IsCapCategoryObject`, `IsCapCategoryObject`, `IsCapCategoryObject`)

▷ `CoclosedEvaluationMorphismWithGivenRange(a, b, r)` (operation)

This is a legacy operation for `CoclosedMonoidalLeftEvaluationMorphismWithGivenRange(b, a, r)`, i.e., with the first and second argument interchanged.

5.1.5 CoevaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ `CoevaluationMorphism(a, b)` (operation)

This is a legacy operation for `ClosedMonoidalLeftCoevaluationMorphism(b, a)`, i.e., with the first and second argument interchanged.

5.1.6 CoevaluationMorphismWithGivenRange (for IsCapCategoryObject, IsCapCategoryObject)

▷ `CoevaluationMorphismWithGivenRange(a, b, r)` (operation)

This is a legacy operation for `ClosedMonoidalLeftCoevaluationMorphismWithGivenRange(b, a, r)`, i.e., with the first and second argument interchanged.

5.2 Synonyms for legacy operations

5.2.1 EvaluationMorphism (for IsCapCategoryObject, IsCapCategoryObject)

▷ `EvaluationMorphism(arg1, arg2)` (operation)

This is a synonym for `ClosedMonoidalLeftEvaluationMorphism`.

5.2.2 EvaluationMorphismWithGivenSource (for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject)

▷ `EvaluationMorphismWithGivenSource(arg1, arg2, arg3)` (operation)

This is a synonym for `ClosedMonoidalLeftEvaluationMorphismWithGivenSource`.

5.2.3 InternalCoHomToTensorProductAdjunctionMap (for IsObject)

▷ `InternalCoHomToTensorProductAdjunctionMap(arg)` (operation)

This is a synonym for `InternalCoHomToTensorProductLeftAdjunctMorphism`.

5.2.4 InternalCoHomToTensorProductAdjunctionMapWithGivenTensorProduct (for IsObject)

▷ `InternalCoHomToTensorProductAdjunctionMapWithGivenTensorProduct(arg)` (operation)

This is a synonym for `InternalCoHomToTensorProductLeftAdjunctionMapWithGivenTensorProduct`.

5.2.5 InternalHomToTensorProductAdjunctionMap (for IsObject)

▷ InternalHomToTensorProductAdjunctionMap(*arg*) (operation)

This is a synonym for InternalHomToTensorProductLeftAdjunctMorphism.

5.2.6 InternalHomToTensorProductAdjunctionMapWithGivenTensorProduct (for IsObject)

▷ InternalHomToTensorProductAdjunctionMapWithGivenTensorProduct(*arg*) (operation)

This is a synonym for InternalHomToTensorProductLeftAdjunctionMapWithGivenTensorProduct.

5.2.7 TensorProductToInternalCoHomAdjunctionMap (for IsObject)

▷ TensorProductToInternalCoHomAdjunctionMap(*arg*) (operation)

This is a synonym for TensorProductToInternalCoHomLeftAdjunctMorphism.

5.2.8 TensorProductToInternalCoHomAdjunctionMapWithGivenInternalCoHom (for IsObject)

▷ TensorProductToInternalCoHomAdjunctionMapWithGivenInternalCoHom(*arg*) (operation)

This is a synonym for TensorProductToInternalCoHomLeftAdjunctMorphismWithGivenInternalCoHom.

5.2.9 TensorProductToInternalHomAdjunctionMap (for IsObject)

▷ TensorProductToInternalHomAdjunctionMap(*arg*) (operation)

This is a synonym for TensorProductToInternalHomLeftAdjunctMorphism.

5.2.10 TensorProductToInternalHomAdjunctionMapWithGivenInternalHom (for IsObject)

▷ TensorProductToInternalHomAdjunctionMapWithGivenInternalHom(*arg*) (operation)

This is a synonym for TensorProductToInternalHomLeftAdjunctMorphismWithGivenInternalHom.

5.2.11 InternalCoHomToTensorProductLeftAdjunctionMap (for IsObject)

▷ InternalCoHomToTensorProductLeftAdjunctionMap(*arg*) (operation)

This is a synonym for InternalCoHomToTensorProductLeftAdjunctMorphism.

5.2.12 InternalHomToTensorProductLeftAdjunctionMap (for IsObject)

▷ `InternalHomToTensorProductLeftAdjunctionMap(arg)` (operation)

This is a synonym for `InternalHomToTensorProductLeftAdjunctMorphism`.

5.2.13 TensorProductToInternalCoHomLeftAdjunctionMap (for IsObject)

▷ `TensorProductToInternalCoHomLeftAdjunctionMap(arg)` (operation)

This is a synonym for `TensorProductToInternalCoHomLeftAdjunctMorphism`.

5.2.14 TensorProductToInternalCoHomLeftAdjunctionMapWithGivenInternalCoHom (for IsObject)

▷ `TensorProductToInternalCoHomLeftAdjunctionMapWithGivenInternalCoHom(arg)` (operation)

This is a synonym for `TensorProductToInternalCoHomLeftAdjunctMorphismWithGivenInternalCoHom`.

5.2.15 TensorProductToInternalHomLeftAdjunctionMap (for IsObject)

▷ `TensorProductToInternalHomLeftAdjunctionMap(arg)` (operation)

This is a synonym for `TensorProductToInternalHomLeftAdjunctMorphism`.

5.2.16 TensorProductToInternalHomLeftAdjunctionMapWithGivenInternalHom (for IsObject)

▷ `TensorProductToInternalHomLeftAdjunctionMapWithGivenInternalHom(arg)` (operation)

This is a synonym for `TensorProductToInternalHomLeftAdjunctMorphismWithGivenInternalHom`.

Chapter 6

MonoidalCategories automatic generated documentation

6.1 MonoidalCategories automatic generated documentation of properties

6.1.1 IsBraidedMonoidalCategory (for IsCapCategory)

▷ `IsBraidedMonoidalCategory(\mathcal{C})` (property)
Returns: true or false
The property of the category \mathcal{C} being braided monoidal.

6.1.2 IsClosedMonoidalCategory (for IsCapCategory)

▷ `IsClosedMonoidalCategory(\mathcal{C})` (property)
Returns: true or false
The property of the category \mathcal{C} being (bi)closed monoidal.

6.1.3 IsCoclosedMonoidalCategory (for IsCapCategory)

▷ `IsCoclosedMonoidalCategory(\mathcal{C})` (property)
Returns: true or false
The property of the category \mathcal{C} being (bi)coclosed monoidal.

6.1.4 IsLeftClosedMonoidalCategory (for IsCapCategory)

▷ `IsLeftClosedMonoidalCategory(\mathcal{C})` (property)
Returns: true or false
The property of the category \mathcal{C} being left closed monoidal.

6.1.5 IsLeftCoclosedMonoidalCategory (for IsCapCategory)

▷ `IsLeftCoclosedMonoidalCategory(\mathcal{C})` (property)
Returns: true or false
The property of the category \mathcal{C} being coclosed monoidal.

6.1.6 IsMonoidalCategory (for IsCapCategory)

- ▷ IsMonoidalCategory(C) (property)
Returns: true or false
 The property of the category C being monoidal.

6.1.7 IsStrictMonoidalCategory (for IsCapCategory)

- ▷ IsStrictMonoidalCategory(C) (property)
Returns: true or false
 The property of the category C being strict monoidal.

6.1.8 IsRigidSymmetricClosedMonoidalCategory (for IsCapCategory)

- ▷ IsRigidSymmetricClosedMonoidalCategory(C) (property)
Returns: true or false
 The property of the category C being rigid symmetric closed monoidal.

6.1.9 IsRigidSymmetricCoclosedMonoidalCategory (for IsCapCategory)

- ▷ IsRigidSymmetricCoclosedMonoidalCategory(C) (property)
Returns: true or false
 The property of the category C being rigid symmetric coclosed monoidal.

6.1.10 IsSymmetricClosedMonoidalCategory (for IsCapCategory)

- ▷ IsSymmetricClosedMonoidalCategory(C) (property)
Returns: true or false
 The property of the category C being symmetric closed monoidal.

6.1.11 IsSymmetricCoclosedMonoidalCategory (for IsCapCategory)

- ▷ IsSymmetricCoclosedMonoidalCategory(C) (property)
Returns: true or false
 The property of the category C being symmetric coclosed monoidal.

6.1.12 IsSymmetricMonoidalCategory (for IsCapCategory)

- ▷ IsSymmetricMonoidalCategory(C) (property)
Returns: true or false
 The property of the category C being symmetric monoidal.

Index

AddAssociatorLeftToRight
 for IsCapCategory, IsFunction, [82](#)
 AddAssociatorLeftToRightWithGivenTensorProducts
 for IsCapCategory, IsFunction, [82](#)
 AddAssociatorRightToLeft
 for IsCapCategory, IsFunction, [82](#)
 AddAssociatorRightToLeftWithGivenTensorProducts
 for IsCapCategory, IsFunction, [83](#)
 AddBraiding
 for IsCapCategory, IsFunction, [47](#)
 AddBraidingInverse
 for IsCapCategory, IsFunction, [47](#)
 AddBraidingInverseWithGivenTensorProducts
 for IsCapCategory, IsFunction, [47](#)
 AddBraidingWithGivenTensorProducts
 for IsCapCategory, IsFunction, [48](#)
 AddClosedMonoidalLeftCoevaluationMorphism
 for IsCapCategory, IsFunction, [48](#)
 AddClosedMonoidalLeftCoevaluationMorphismWithGivenRange
 for IsCapCategory, IsFunction, [48](#)
 AddClosedMonoidalLeftEvaluationMorphism
 for IsCapCategory, IsFunction, [48](#)
 AddClosedMonoidalLeftEvaluationMorphismWithGivenSource
 for IsCapCategory, IsFunction, [48](#)
 AddClosedMonoidalRightCoevaluationMorphism
 for IsCapCategory, IsFunction, [49](#)
 AddClosedMonoidalRightCoevaluationMorphismWithGivenRange
 for IsCapCategory, IsFunction, [49](#)
 AddClosedMonoidalRightEvaluationMorphism

Morphism
 for IsCapCategory, IsFunction, [49](#)
 AddClosedMonoidalRightEvaluation-
 MorphismWithGivenSource
 for IsCapCategory, IsFunction, [49](#)
 AddCoclosedCoevaluationForCoDual
 for IsCapCategory, IsFunction, [87](#)
 AddCoclosedCoevaluationForCoDualWith-
 GivenTensorProduct
 for IsCapCategory, IsFunction, [87](#)
 AddCoclosedEvaluationForCoDual
 for IsCapCategory, IsFunction, [60](#)
 AddCoclosedEvaluationForCoDualWith-
 GivenTensorProduct
 for IsCapCategory, IsFunction, [60](#)
 AddCoclosedMonoidalLeftCoevaluation-
 Morphism
 for IsCapCategory, IsFunction, [60](#)
 AddCoclosedMonoidalLeftCoevaluation-
 MorphismWithGivenSource
 for IsCapCategory, IsFunction, [60](#)
 AddCoclosedMonoidalLeftEvaluation-
 Morphism
 for IsCapCategory, IsFunction, [60](#)
 AddCoclosedMonoidalLeftEvaluation-
 MorphismWithGivenRange
 for IsCapCategory, IsFunction, [61](#)
 AddCoclosedMonoidalRightCoevaluation-
 Morphism
 for IsCapCategory, IsFunction, [61](#)
 AddCoclosedMonoidalRightCoevaluation-
 MorphismWithGivenSource
 for IsCapCategory, IsFunction, [61](#)
 AddCoclosedMonoidalRightEvaluation-
 Morphism
 for IsCapCategory, IsFunction, [61](#)
 AddCoclosedMonoidalRightEvaluation-
 MorphismWithGivenRange

- for IsCapCategory, IsFunction, 61
- AddCoDualityTensorProduct-
 - CompatibilityMorphism
 - for IsCapCategory, IsFunction, 59
- AddCoDualityTensorProduct-
 - CompatibilityMorphismWith-
 - GivenObjects
 - for IsCapCategory, IsFunction, 59
- AddCoDualOnMorphisms
 - for IsCapCategory, IsFunction, 58
- AddCoDualOnMorphismsWithGivenCoDuals
 - for IsCapCategory, IsFunction, 58
- AddCoDualOnObjects
 - for IsCapCategory, IsFunction, 59
- AddCoevaluationForDual
 - for IsCapCategory, IsFunction, 84
- AddCoevaluationForDualWithGivenTensor-
 - Product
 - for IsCapCategory, IsFunction, 85
- AddCoLambdaElimination
 - for IsCapCategory, IsFunction, 59
- AddCoLambdaIntroduction
 - for IsCapCategory, IsFunction, 59
- AddCoRankMorphism
 - for IsCapCategory, IsFunction, 87
- AddCoTraceMap
 - for IsCapCategory, IsFunction, 87
- AddDualOnMorphisms
 - for IsCapCategory, IsFunction, 49
- AddDualOnMorphismsWithGivenDuals
 - for IsCapCategory, IsFunction, 49
- AddDualOnObjects
 - for IsCapCategory, IsFunction, 50
- AddEvaluationForDual
 - for IsCapCategory, IsFunction, 50
- AddEvaluationForDualWithGivenTensor-
 - Product
 - for IsCapCategory, IsFunction, 50
- AddInternalCoHomOnMorphisms
 - for IsCapCategory, IsFunction, 62
- AddInternalCoHomOnMorphismsWithGiven-
 - InternalCoHoms
 - for IsCapCategory, IsFunction, 62
- AddInternalCoHomTensorProduct-
 - CompatibilityMorphism
 - for IsCapCategory, IsFunction, 62
- AddInternalCoHomTensorProduct-
 - CompatibilityMorphismInverse
 - for IsCapCategory, IsFunction, 87
- AddInternalCoHomTensorProduct-
 - CompatibilityMorphismInverse-
 - WithGivenObjects
 - for IsCapCategory, IsFunction, 88
- AddInternalCoHomTensorProduct-
 - CompatibilityMorphismWith-
 - GivenObjects
 - for IsCapCategory, IsFunction, 62
- AddInternalCoHomToTensorProductLeft-
 - AdjunctMorphism
 - for IsCapCategory, IsFunction, 63
- AddInternalCoHomToTensorProductLeft-
 - AdjunctMorphismWithGivenTensor-
 - Product
 - for IsCapCategory, IsFunction, 63
- AddInternalCoHomToTensorProductRight-
 - AdjunctMorphism
 - for IsCapCategory, IsFunction, 63
- AddInternalCoHomToTensorProductRight-
 - AdjunctMorphismWithGivenTensor-
 - Product
 - for IsCapCategory, IsFunction, 63
- AddInternalHomOnMorphisms
 - for IsCapCategory, IsFunction, 50
- AddInternalHomOnMorphismsWithGiven-
 - InternalHoms
 - for IsCapCategory, IsFunction, 50
- AddInternalHomOnObjects
 - for IsCapCategory, IsFunction, 50
- AddInternalHomToTensorProductLeft-
 - AdjunctionIsomorphism
 - for IsCapCategory, IsFunction, 51
- AddInternalHomToTensorProductLeft-
 - AdjunctionIsomorphismWithGiven-
 - Objects
 - for IsCapCategory, IsFunction, 51
- AddInternalHomToTensorProductLeft-
 - AdjunctMorphism
 - for IsCapCategory, IsFunction, 51
- AddInternalHomToTensorProductLeft-
 - AdjunctMorphismWithGivenTensor-

Product
 for IsCapCategory, IsFunction, 51
 AddInternalHomToTensorProductRight-
 AdjunctionIsomorphism
 for IsCapCategory, IsFunction, 52
 AddInternalHomToTensorProductRight-
 AdjunctionIsomorphismWithGiven-
 Objects
 for IsCapCategory, IsFunction, 52
 AddInternalHomToTensorProductRight-
 AdjunctMorphism
 for IsCapCategory, IsFunction, 52
 AddInternalHomToTensorProductRight-
 AdjunctMorphismWithGivenTensor-
 Product
 for IsCapCategory, IsFunction, 52
 AddIsomorphismFromCoDualObjectTo-
 InternalCoHomFromTensorUnit
 for IsCapCategory, IsFunction, 63
 AddIsomorphismFromDualObjectTo-
 InternalHomIntoTensorUnit
 for IsCapCategory, IsFunction, 52
 AddIsomorphismFromInternalCoHomFrom-
 TensorUnitToCoDualObject
 for IsCapCategory, IsFunction, 64
 AddIsomorphismFromInternalCoHomTo-
 Object
 for IsCapCategory, IsFunction, 64
 AddIsomorphismFromInternalCoHomTo-
 ObjectWithGivenInternalCoHom
 for IsCapCategory, IsFunction, 64
 AddIsomorphismFromInternalCoHomTo-
 TensorProductWithCoDualObject
 for IsCapCategory, IsFunction, 88
 AddIsomorphismFromInternalHomInto-
 TensorUnitToDualObject
 for IsCapCategory, IsFunction, 53
 AddIsomorphismFromInternalHomToObject
 for IsCapCategory, IsFunction, 53
 AddIsomorphismFromInternalHomToObject-
 WithGivenInternalHom
 for IsCapCategory, IsFunction, 53
 AddIsomorphismFromInternalHomToTensor-
 ProductWithDualObject
 for IsCapCategory, IsFunction, 85
 AddIsomorphismFromLeftCoDualObjectTo-
 LeftInternalCoHomFromTensorUnit
 for IsCapCategory, IsFunction, 74
 AddIsomorphismFromLeftDualObjectTo-
 LeftInternalHomIntoTensorUnit
 for IsCapCategory, IsFunction, 67
 AddIsomorphismFromLeftInternalCoHom-
 FromTensorUnitToLeftCoDual-
 Object
 for IsCapCategory, IsFunction, 75
 AddIsomorphismFromLeftInternalCoHomTo-
 Object
 for IsCapCategory, IsFunction, 75
 AddIsomorphismFromLeftInternalCoHom-
 ToObjectWithGivenLeftInternal-
 CoHom
 for IsCapCategory, IsFunction, 75
 AddIsomorphismFromLeftInternalHomInto-
 TensorUnitToLeftDualObject
 for IsCapCategory, IsFunction, 67
 AddIsomorphismFromLeftInternalHomTo-
 Object
 for IsCapCategory, IsFunction, 68
 AddIsomorphismFromLeftInternalHomTo-
 ObjectWithGivenLeftInternalHom
 for IsCapCategory, IsFunction, 68
 AddIsomorphismFromObjectToInternal-
 CoHom
 for IsCapCategory, IsFunction, 64
 AddIsomorphismFromObjectToInternalCo-
 HomWithGivenInternalCoHom
 for IsCapCategory, IsFunction, 64
 AddIsomorphismFromObjectToInternalHom
 for IsCapCategory, IsFunction, 53
 AddIsomorphismFromObjectToInternalHom-
 WithGivenInternalHom
 for IsCapCategory, IsFunction, 53
 AddIsomorphismFromObjectToLeft-
 InternalCoHom
 for IsCapCategory, IsFunction, 75
 AddIsomorphismFromObjectToLeft-
 InternalCoHomWithGivenLeft-
 InternalCoHom
 for IsCapCategory, IsFunction, 76
 AddIsomorphismFromObjectToLeft-
 InternalHom
 for IsCapCategory, IsFunction, 68

AddIsomorphismFromObjectToLeft-
 InternalHomWithGivenLeft-
 InternalHom
 for IsCapCategory, IsFunction, 68
 AddIsomorphismFromTensorProductWithCo-
 DualObjectToInternalCoHom
 for IsCapCategory, IsFunction, 88
 AddIsomorphismFromTensorProductWith-
 DualObjectToInternalHom
 for IsCapCategory, IsFunction, 85
 AdditiveMonoidalCategoriesTest, 90
 AddLambdaElimination
 for IsCapCategory, IsFunction, 54
 AddLambdaIntroduction
 for IsCapCategory, IsFunction, 54
 AddLeftClosedMonoidalCoevaluation-
 Morphism
 for IsCapCategory, IsFunction, 68
 AddLeftClosedMonoidalCoevaluation-
 MorphismWithGivenRange
 for IsCapCategory, IsFunction, 69
 AddLeftClosedMonoidalEvaluationFor-
 LeftDual
 for IsCapCategory, IsFunction, 69
 AddLeftClosedMonoidalEvaluationFor-
 LeftDualWithGivenTensorProduct
 for IsCapCategory, IsFunction, 69
 AddLeftClosedMonoidalEvaluation-
 Morphism
 for IsCapCategory, IsFunction, 69
 AddLeftClosedMonoidalEvaluation-
 MorphismWithGivenSource
 for IsCapCategory, IsFunction, 69
 AddLeftClosedMonoidalLambdaElimination
 for IsCapCategory, IsFunction, 70
 AddLeftClosedMonoidalLambda-
 Introduction
 for IsCapCategory, IsFunction, 70
 AddLeftClosedMonoidalPostCompose-
 Morphism
 for IsCapCategory, IsFunction, 70
 AddLeftClosedMonoidalPostCompose-
 MorphismWithGivenObjects
 for IsCapCategory, IsFunction, 70
 AddLeftClosedMonoidalPreCompose-
 Morphism
 for IsCapCategory, IsFunction, 70
 AddLeftClosedMonoidalPreCompose-
 MorphismWithGivenObjects
 for IsCapCategory, IsFunction, 71
 AddLeftCoclosedMonoidalCoevaluation-
 Morphism
 for IsCapCategory, IsFunction, 77
 AddLeftCoclosedMonoidalCoevaluation-
 MorphismWithGivenSource
 for IsCapCategory, IsFunction, 77
 AddLeftCoclosedMonoidalEvaluationFor-
 LeftCoDual
 for IsCapCategory, IsFunction, 77
 AddLeftCoclosedMonoidalEvaluation-
 ForLeftCoDualWithGivenTensor-
 Product
 for IsCapCategory, IsFunction, 77
 AddLeftCoclosedMonoidalEvaluation-
 Morphism
 for IsCapCategory, IsFunction, 78
 AddLeftCoclosedMonoidalEvaluation-
 MorphismWithGivenRange
 for IsCapCategory, IsFunction, 78
 AddLeftCoclosedMonoidalLambda-
 Elimination
 for IsCapCategory, IsFunction, 78
 AddLeftCoclosedMonoidalLambda-
 Introduction
 for IsCapCategory, IsFunction, 78
 AddLeftCoclosedMonoidalPostCoCompose-
 Morphism
 for IsCapCategory, IsFunction, 78
 AddLeftCoclosedMonoidalPostCoCompose-
 MorphismWithGivenObjects
 for IsCapCategory, IsFunction, 79
 AddLeftCoclosedMonoidalPreCoCompose-
 Morphism
 for IsCapCategory, IsFunction, 79
 AddLeftCoclosedMonoidalPreCoCompose-
 MorphismWithGivenObjects
 for IsCapCategory, IsFunction, 79
 AddLeftCoDualityTensorProduct-
 CompatibilityMorphism
 for IsCapCategory, IsFunction, 76
 AddLeftCoDualityTensorProduct-
 CompatibilityMorphismWith-

GivenObjects
 for IsCapCategory, IsFunction, 77
 AddLeftCoDualOnMorphisms
 for IsCapCategory, IsFunction, 76
 AddLeftCoDualOnMorphismsWithGivenLeft-
 CoDuals
 for IsCapCategory, IsFunction, 76
 AddLeftCoDualOnObjects
 for IsCapCategory, IsFunction, 76
 AddLeftDistributivityExpanding
 for IsCapCategory, IsFunction, 46
 AddLeftDistributivityExpandingWith-
 GivenObjects
 for IsCapCategory, IsFunction, 46
 AddLeftDistributivityFactoring
 for IsCapCategory, IsFunction, 46
 AddLeftDistributivityFactoringWith-
 GivenObjects
 for IsCapCategory, IsFunction, 46
 AddLeftDualOnMorphisms
 for IsCapCategory, IsFunction, 71
 AddLeftDualOnMorphismsWithGivenLeft-
 Duals
 for IsCapCategory, IsFunction, 71
 AddLeftDualOnObjects
 for IsCapCategory, IsFunction, 71
 AddLeftInternalCoHomOnMorphisms
 for IsCapCategory, IsFunction, 79
 AddLeftInternalCoHomOnMorphismsWith-
 GivenLeftInternalCoHoms
 for IsCapCategory, IsFunction, 79
 AddLeftInternalCoHomOnObjects
 for IsCapCategory, IsFunction, 80
 AddLeftInternalCoHomTensorProduct-
 CompatibilityMorphism
 for IsCapCategory, IsFunction, 80
 AddLeftInternalCoHomTensorProduct-
 CompatibilityMorphismWithGiven-
 Objects
 for IsCapCategory, IsFunction, 80
 AddLeftInternalCoHomToTensorProduct-
 AdjunctMorphism
 for IsCapCategory, IsFunction, 80
 AddLeftInternalCoHomToTensorProduct-
 AdjunctMorphismWithGivenTensor-
 Product
 for IsCapCategory, IsFunction, 80
 AddLeftInternalHomOnMorphisms
 for IsCapCategory, IsFunction, 71
 AddLeftInternalHomOnMorphismsWith-
 GivenLeftInternalHoms
 for IsCapCategory, IsFunction, 71
 AddLeftInternalHomOnObjects
 for IsCapCategory, IsFunction, 72
 AddLeftInternalHomToTensorProduct-
 AdjunctMorphism
 for IsCapCategory, IsFunction, 72
 AddLeftInternalHomToTensorProduct-
 AdjunctMorphismWithGivenTensor-
 Product
 for IsCapCategory, IsFunction, 72
 AddLeftUnitor
 for IsCapCategory, IsFunction, 83
 AddLeftUnitorInverse
 for IsCapCategory, IsFunction, 83
 AddLeftUnitorInverseWithGivenTensor-
 Product
 for IsCapCategory, IsFunction, 83
 AddLeftUnitorWithGivenTensorProduct
 for IsCapCategory, IsFunction, 83
 AddMonoidalPostComposeMorphism
 for IsCapCategory, IsFunction, 65
 AddMonoidalPostComposeMorphismWith-
 GivenObjects
 for IsCapCategory, IsFunction, 65
 AddMonoidalPostComposeMorphism
 for IsCapCategory, IsFunction, 54
 AddMonoidalPostComposeMorphismWith-
 GivenObjects
 for IsCapCategory, IsFunction, 54
 AddMonoidalPreComposeMorphism
 for IsCapCategory, IsFunction, 65
 AddMonoidalPreComposeMorphismWith-
 GivenObjects
 for IsCapCategory, IsFunction, 65
 AddMonoidalPreComposeMorphism
 for IsCapCategory, IsFunction, 54
 AddMonoidalPreComposeMorphismWith-
 GivenObjects
 for IsCapCategory, IsFunction, 55
 AddMorphismFromBidual
 for IsCapCategory, IsFunction, 85

AddMorphismFromBidualWithGivenBidual
 for IsCapCategory, IsFunction, 85
 AddMorphismFromCoBidual
 for IsCapCategory, IsFunction, 65
 AddMorphismFromCoBidualWithGivenCo-
 Bidual
 for IsCapCategory, IsFunction, 66
 AddMorphismFromInternalCoHomToTensor-
 Product
 for IsCapCategory, IsFunction, 66
 AddMorphismFromInternalCoHomToTensor-
 ProductWithGivenObjects
 for IsCapCategory, IsFunction, 66
 AddMorphismFromInternalHomToTensor-
 Product
 for IsCapCategory, IsFunction, 86
 AddMorphismFromInternalHomToTensor-
 ProductWithGivenObjects
 for IsCapCategory, IsFunction, 86
 AddMorphismFromLeftCoBidual
 for IsCapCategory, IsFunction, 81
 AddMorphismFromLeftCoBidualWithGiven-
 LeftCoBidual
 for IsCapCategory, IsFunction, 81
 AddMorphismFromLeftInternalCoHomTo-
 TensorProduct
 for IsCapCategory, IsFunction, 81
 AddMorphismFromLeftInternalCoHomTo-
 TensorProductWithGivenObjects
 for IsCapCategory, IsFunction, 81
 AddMorphismFromTensorProductTo-
 InternalCoHom
 for IsCapCategory, IsFunction, 88
 AddMorphismFromTensorProductTo-
 InternalCoHomWithGivenObjects
 for IsCapCategory, IsFunction, 88
 AddMorphismFromTensorProductTo-
 InternalHom
 for IsCapCategory, IsFunction, 55
 AddMorphismFromTensorProductTo-
 InternalHomWithGivenObjects
 for IsCapCategory, IsFunction, 55
 AddMorphismFromTensorProductToLeft-
 InternalHom
 for IsCapCategory, IsFunction, 72
 AddMorphismFromTensorProductToLeft-
 InternalHomWithGivenObjects
 for IsCapCategory, IsFunction, 72
 AddMorphismToBidual
 for IsCapCategory, IsFunction, 55
 AddMorphismToBidualWithGivenBidual
 for IsCapCategory, IsFunction, 55
 AddMorphismToCoBidual
 for IsCapCategory, IsFunction, 89
 AddMorphismToCoBidualWithGivenCoBidual
 for IsCapCategory, IsFunction, 89
 AddMorphismToLeftBidual
 for IsCapCategory, IsFunction, 73
 AddMorphismToLeftBidualWithGivenLeft-
 Bidual
 for IsCapCategory, IsFunction, 73
 AddRankMorphism
 for IsCapCategory, IsFunction, 86
 AddRightDistributivityExpanding
 for IsCapCategory, IsFunction, 46
 AddRightDistributivityExpandingWith-
 GivenObjects
 for IsCapCategory, IsFunction, 47
 AddRightDistributivityFactoring
 for IsCapCategory, IsFunction, 47
 AddRightDistributivityFactoringWith-
 GivenObjects
 for IsCapCategory, IsFunction, 47
 AddRightUnitor
 for IsCapCategory, IsFunction, 83
 AddRightUnitorInverse
 for IsCapCategory, IsFunction, 84
 AddRightUnitorInverseWithGivenTensor-
 Product
 for IsCapCategory, IsFunction, 84
 AddRightUnitorWithGivenTensorProduct
 for IsCapCategory, IsFunction, 84
 AddTensorProductDualityCompatibility-
 Morphism
 for IsCapCategory, IsFunction, 55
 AddTensorProductDualityCompatibility-
 MorphismWithGivenObjects
 for IsCapCategory, IsFunction, 56
 AddTensorProductInternalHom-
 CompatibilityMorphism
 for IsCapCategory, IsFunction, 56
 AddTensorProductInternalHom-

CompatibilityMorphismInverse
 for IsCapCategory, IsFunction, 86
 AddTensorProductInternalHom-
 CompatibilityMorphismInverse-
 WithGivenObjects
 for IsCapCategory, IsFunction, 86
 AddTensorProductInternalHom-
 CompatibilityMorphismWith-
 GivenObjects
 for IsCapCategory, IsFunction, 56
 AddTensorProductLeftDuality-
 CompatibilityMorphism
 for IsCapCategory, IsFunction, 73
 AddTensorProductLeftDuality-
 CompatibilityMorphismWith-
 GivenObjects
 for IsCapCategory, IsFunction, 73
 AddTensorProductLeftInternalHom-
 CompatibilityMorphism
 for IsCapCategory, IsFunction, 73
 AddTensorProductLeftInternalHom-
 CompatibilityMorphismWith-
 GivenObjects
 for IsCapCategory, IsFunction, 74
 AddTensorProductOnMorphisms
 for IsCapCategory, IsFunction, 84
 AddTensorProductOnMorphismsWithGiven-
 TensorProducts
 for IsCapCategory, IsFunction, 84
 AddTensorProductOnObjects
 for IsCapCategory, IsFunction, 6
 AddTensorProductToInternalCoHomLeft-
 AdjunctMorphism
 for IsCapCategory, IsFunction, 66
 AddTensorProductToInternalCoHomLeft-
 AdjunctMorphismWithGiven-
 InternalCoHom
 for IsCapCategory, IsFunction, 66
 AddTensorProductToInternalCoHomRight-
 AdjunctMorphism
 for IsCapCategory, IsFunction, 67
 AddTensorProductToInternalCoHomRight-
 AdjunctMorphismWithGiven-
 InternalCoHom
 for IsCapCategory, IsFunction, 67
 AddTensorProductToInternalHomLeft-

AdjunctionIsomorphism
 for IsCapCategory, IsFunction, 57
 AddTensorProductToInternalHomLeft-
 AdjunctionIsomorphismWithGiven-
 Objects
 for IsCapCategory, IsFunction, 57
 AddTensorProductToInternalHomLeft-
 AdjunctMorphism
 for IsCapCategory, IsFunction, 56
 AddTensorProductToInternalHomLeft-
 AdjunctMorphismWithGiven-
 InternalHom
 for IsCapCategory, IsFunction, 57
 AddTensorProductToInternalHomRight-
 AdjunctionIsomorphism
 for IsCapCategory, IsFunction, 58
 AddTensorProductToInternalHomRight-
 AdjunctionIsomorphismWithGiven-
 Objects
 for IsCapCategory, IsFunction, 58
 AddTensorProductToInternalHomRight-
 AdjunctMorphism
 for IsCapCategory, IsFunction, 57
 AddTensorProductToInternalHomRight-
 AdjunctMorphismWithGiven-
 InternalHom
 for IsCapCategory, IsFunction, 57
 AddTensorProductToLeftInternalCoHom-
 AdjunctMorphism
 for IsCapCategory, IsFunction, 81
 AddTensorProductToLeftInternalCoHom-
 AdjunctMorphismWithGivenLeft-
 InternalCoHom
 for IsCapCategory, IsFunction, 82
 AddTensorProductToLeftInternalHom-
 AdjunctMorphism
 for IsCapCategory, IsFunction, 74
 AddTensorProductToLeftInternalHom-
 AdjunctMorphismWithGivenLeft-
 InternalHom
 for IsCapCategory, IsFunction, 74
 AddTensorUnit
 for IsCapCategory, IsFunction, 6
 AddTraceMap
 for IsCapCategory, IsFunction, 87
 AddUniversalPropertyOfCoDual

for IsCapCategory, IsFunction, 67
 AddUniversalPropertyOfDual
 for IsCapCategory, IsFunction, 58
 AddUniversalPropertyOfLeftCoDual
 for IsCapCategory, IsFunction, 82
 AddUniversalPropertyOfLeftDual
 for IsCapCategory, IsFunction, 74
 AssociatorLeftToRight
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 4
 AssociatorLeftToRightWithGivenTensor-
 Products
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, IsCapCategoryObject,
 4
 AssociatorRightToLeft
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 4
 AssociatorRightToLeftWithGivenTensor-
 Products
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, IsCapCategoryObject,
 4
 BraidedMonoidalCategoriesTest, 90
 Braiding
 for IsCapCategoryObject, IsCapCategory-
 Object, 8
 BraidingInverse
 for IsCapCategoryObject, IsCapCategory-
 Object, 8
 BraidingInverseWithGivenTensorProducts
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, 8
 BraidingWithGivenTensorProducts
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, 8
 ClosedMonoidalCategoriesTest, 91
 ClosedMonoidalCategoriesTestWithGiven,
 91
 ClosedMonoidalLeftCoevaluationMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, 18
 ClosedMonoidalLeftCoevaluation-
 MorphismWithGivenRange
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 18
 ClosedMonoidalLeftEvaluationMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, 18
 ClosedMonoidalLeftEvaluationMorphism-
 WithGivenSource
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 18
 ClosedMonoidalRightCoevaluation-
 Morphism
 for IsCapCategoryObject, IsCapCategory-
 Object, 16
 ClosedMonoidalRightCoevaluation-
 MorphismWithGivenRange
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 16
 ClosedMonoidalRightEvaluationMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, 16
 ClosedMonoidalRightEvaluationMorphism-
 WithGivenSource
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 16
 CoclosedCoevaluationForCoDual
 for IsCapCategoryObject, 44
 CoclosedCoevaluationForCoDualWith-
 GivenTensorProduct
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 44
 CoclosedCoevaluationMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, 102
 CoclosedCoevaluationMorphismWithGiven-
 Source
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 102
 CoclosedEvaluationForCoDual
 for IsCapCategoryObject, 37
 CoclosedEvaluationForCoDualWithGiven-
 TensorProduct
 for IsCapCategoryObject, IsCapCategory-

- Object, IsCapCategoryObject, [37](#)
- CoclosedEvaluationMorphism
 - for IsCapCategoryObject, IsCapCategory-Object, [102](#)
- CoclosedEvaluationMorphismWithGiven-Range
 - for IsCapCategoryObject, IsCapCategory-Object, IsCapCategoryObject, [102](#)
- CoclosedMonoidalCategoriesTest, [92](#)
- CoclosedMonoidalCategoriesTestWith-Given, [93](#)
- CoclosedMonoidalLeftCoevaluation-Morphism
 - for IsCapCategoryObject, IsCapCategory-Object, [34](#)
- CoclosedMonoidalLeftCoevaluation-MorphismWithGivenSource
 - for IsCapCategoryObject, IsCapCategory-Object, IsCapCategoryObject, [34](#)
- CoclosedMonoidalLeftEvaluationMorphism
 - for IsCapCategoryObject, IsCapCategory-Object, [34](#)
- CoclosedMonoidalLeftEvaluation-MorphismWithGivenRange
 - for IsCapCategoryObject, IsCapCategory-Object, IsCapCategoryObject, [34](#)
- CoclosedMonoidalRightCoevaluation-Morphism
 - for IsCapCategoryObject, IsCapCategory-Object, [33](#)
- CoclosedMonoidalRightCoevaluation-MorphismWithGivenSource
 - for IsCapCategoryObject, IsCapCategory-Object, IsCapCategoryObject, [33](#)
- CoclosedMonoidalRightEvaluation-Morphism
 - for IsCapCategoryObject, IsCapCategory-Object, [32](#)
- CoclosedMonoidalRightEvaluation-MorphismWithGivenRange
 - for IsCapCategoryObject, IsCapCategory-Object, IsCapCategoryObject, [32](#)
- CoDualityTensorProductCompatibility-Morphism
 - for IsCapCategoryObject, IsCapCategory-Object, [38](#)
- CoDualityTensorProductCompatibility-MorphismWithGivenObjects
 - for IsCapCategoryObject, IsCapCategory-Object, IsCapCategoryObject, IsCap-CategoryObject, [38](#)
- CoDualOnMorphisms
 - for IsCapCategoryMorphism, [36](#)
- CoDualOnMorphismsWithGivenCoDUALs
 - for IsCapCategoryObject, IsCapCategory-Morphism, IsCapCategoryObject, [36](#)
- CoDualOnObjects
 - for IsCapCategoryObject, [36](#)
- CoevaluationForDual
 - for IsCapCategoryObject, [42](#)
- CoevaluationForDualWithGivenTensor-Product
 - for IsCapCategoryObject, IsCapCategory-Object, IsCapCategoryObject, [42](#)
- CoevaluationMorphism
 - for IsCapCategoryObject, IsCapCategory-Object, [103](#)
- CoevaluationMorphismWithGivenRange
 - for IsCapCategoryObject, IsCapCategory-Object, IsCapCategoryObject, [103](#)
- CoLambdaElimination
 - for IsCapCategoryObject, IsCapCategory-Object, IsCapCategoryMorphism, [39](#)
- CoLambdaIntroduction
 - for IsCapCategoryMorphism, [39](#)
- CoRankMorphism
 - for IsCapCategoryObject, [44](#)
- CoTraceMap
 - for IsCapCategoryMorphism, [44](#)
- DualOnMorphisms
 - for IsCapCategoryMorphism, [21](#)
- DualOnMorphismsWithGivenDUALs
 - for IsCapCategoryObject, IsCapCategory-Morphism, IsCapCategoryObject, [21](#)
- DualOnObjects
 - for IsCapCategoryObject, [21](#)
- EvaluationForDual
 - for IsCapCategoryObject, [21](#)
- EvaluationForDualWithGivenTensor-Product

for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 21
 EvaluationMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, 103
 EvaluationMorphismWithGivenSource
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 103
 InternalCoHom
 for IsCapCategoryCell, IsCapCategoryCell,
 45
 InternalCoHomOnMorphisms
 for IsCapCategoryMorphism, IsCapCatego-
 ryMorphism, 32
 InternalCoHomOnMorphismsWithGiven-
 InternalCoHoms
 for IsCapCategoryObject, IsCapCategory-
 Morphism, IsCapCategoryMorphism,
 IsCapCategoryObject, 32
 InternalCoHomOnObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, 32
 InternalCoHomTensorProduct-
 CompatibilityMorphism
 for IsList, 37
 InternalCoHomTensorProduct-
 CompatibilityMorphismInverse
 for IsList, 43
 InternalCoHomTensorProduct-
 CompatibilityMorphismInverse-
 WithGivenObjects
 for IsCapCategoryObject, IsList, IsCapCate-
 goryObject, 44
 InternalCoHomTensorProduct-
 CompatibilityMorphismWith-
 GivenObjects
 for IsCapCategoryObject, IsList, IsCapCate-
 goryObject, 37
 InternalCoHomToTensorProduct-
 AdjunctionMap
 for IsObject, 103
 InternalCoHomToTensorProduct-
 AdjunctionMapWithGivenTensor-
 Product
 for IsObject, 103
 InternalCoHomToTensorProductLeft-
 AdjunctionMap
 for IsObject, 104
 InternalCoHomToTensorProductLeft-
 AdjunctMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryMorphism, 35
 InternalCoHomToTensorProductLeft-
 AdjunctMorphismWithGivenTensor-
 Product
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryMorphism, Is-
 CapCategoryObject, 35
 InternalCoHomToTensorProductRight-
 AdjunctMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryMorphism, 33
 InternalCoHomToTensorProductRight-
 AdjunctMorphismWithGivenTensor-
 Product
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryMorphism, Is-
 CapCategoryObject, 34
 InternalHom
 for IsCapCategoryCell, IsCapCategoryCell,
 45
 InternalHomOnMorphisms
 for IsCapCategoryMorphism, IsCapCatego-
 ryMorphism, 15
 InternalHomOnMorphismsWithGiven-
 InternalHoms
 for IsCapCategoryObject, IsCapCategory-
 Morphism, IsCapCategoryMorphism,
 IsCapCategoryObject, 16
 InternalHomOnObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, 15
 InternalHomToTensorProduct-
 AdjunctionMap
 for IsObject, 104
 InternalHomToTensorProductAdjunction-
 MapWithGivenTensorProduct
 for IsObject, 104
 InternalHomToTensorProductLeft-
 AdjunctionIsomorphism
 for IsCapCategoryObject, IsCapCategory-

Object, IsCapCategoryObject, 20
 InternalHomToTensorProductLeft-
 AdjunctionIsomorphismWith-
 GivenObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, IsCapCategoryObject,
 20
 InternalHomToTensorProductLeft-
 AdjunctionMap
 for IsObject, 105
 InternalHomToTensorProductLeftAdjunct-
 Morphism
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryMorphism, 19
 InternalHomToTensorProductLeftAdjunct-
 MorphismWithGivenTensorProduct
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryMorphism, Is-
 CapCategoryObject, 20
 InternalHomToTensorProductRight-
 AdjunctionIsomorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 18
 InternalHomToTensorProductRight-
 AdjunctionIsomorphismWith-
 GivenObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, IsCapCategoryObject,
 18
 InternalHomToTensorProductRight-
 AdjunctMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryMorphism, 17
 InternalHomToTensorProductRight-
 AdjunctMorphismWithGivenTensor-
 Product
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryMorphism, Is-
 CapCategoryObject, 17
 IsBraidedMonoidalCategory
 for IsCapCategory, 106
 IsClosedMonoidalCategory
 for IsCapCategory, 106
 IsCoclosedMonoidalCategory
 for IsCapCategory, 106
 IsLeftClosedMonoidalCategory
 for IsCapCategory, 106
 IsLeftCoclosedMonoidalCategory
 for IsCapCategory, 106
 IsMonoidalCategory
 for IsCapCategory, 107
 IsomorphismFromCoDualObjectToInternal-
 CoHomFromTensorUnit
 for IsCapCategoryObject, 38
 IsomorphismFromDualObjectToInternal-
 HomIntoTensorUnit
 for IsCapCategoryObject, 23
 IsomorphismFromInternalCoHomFrom-
 TensorUnitToCoDualObject
 for IsCapCategoryObject, 39
 IsomorphismFromInternalCoHomToObject
 for IsCapCategoryObject, 40
 IsomorphismFromInternalCoHomToObject-
 WithGivenInternalCoHom
 for IsCapCategoryObject, IsCapCategory-
 Object, 40
 IsomorphismFromInternalCoHomToTensor-
 ProductWithCoDualObject
 for IsCapCategoryObject, IsCapCategory-
 Object, 43
 IsomorphismFromInternalHomIntoTensor-
 UnitToDualObject
 for IsCapCategoryObject, 23
 IsomorphismFromInternalHomToObject
 for IsCapCategoryObject, 24
 IsomorphismFromInternalHomToObject-
 WithGivenInternalHom
 for IsCapCategoryObject, IsCapCategory-
 Object, 24
 IsomorphismFromInternalHomToTensor-
 ProductWithDualObject
 for IsCapCategoryObject, IsCapCategory-
 Object, 41
 IsomorphismFromLeftCoDualObjectToLeft-
 InternalCoHomFromTensorUnit
 for IsCapCategoryObject, 30
 IsomorphismFromLeftDualObjectToLeft-
 InternalHomIntoTensorUnit
 for IsCapCategoryObject, 14
 IsomorphismFromLeftInternalCoHomFrom-

TensorUnitToLeftCoDualObject
 for IsCapCategoryObject, [30](#)
 IsomorphismFromLeftInternalCoHomTo-
 Object
 for IsCapCategoryObject, [31](#)
 IsomorphismFromLeftInternalCoHomTo-
 ObjectWithGivenLeftInternal-
 CoHom
 for IsCapCategoryObject, IsCapCategory-
 Object, [31](#)
 IsomorphismFromLeftInternalHomInto-
 TensorUnitToLeftDualObject
 for IsCapCategoryObject, [14](#)
 IsomorphismFromLeftInternalHomToObject
 for IsCapCategoryObject, [15](#)
 IsomorphismFromLeftInternalHomTo-
 ObjectWithGivenLeftInternalHom
 for IsCapCategoryObject, IsCapCategory-
 Object, [15](#)
 IsomorphismFromObjectToInternalCoHom
 for IsCapCategoryObject, [39](#)
 IsomorphismFromObjectToInternalCoHom-
 WithGivenInternalCoHom
 for IsCapCategoryObject, IsCapCategory-
 Object, [39](#)
 IsomorphismFromObjectToInternalHom
 for IsCapCategoryObject, [24](#)
 IsomorphismFromObjectToInternalHom-
 WithGivenInternalHom
 for IsCapCategoryObject, IsCapCategory-
 Object, [24](#)
 IsomorphismFromObjectToLeftInternal-
 CoHom
 for IsCapCategoryObject, [31](#)
 IsomorphismFromObjectToLeftInternalCo-
 HomWithGivenLeftInternalCoHom
 for IsCapCategoryObject, IsCapCategory-
 Object, [31](#)
 IsomorphismFromObjectToLeftInternalHom
 for IsCapCategoryObject, [14](#)
 IsomorphismFromObjectToLeftInternal-
 HomWithGivenLeftInternalHom
 for IsCapCategoryObject, IsCapCategory-
 Object, [15](#)
 IsomorphismFromTensorProductWithCo-
 DualObjectToInternalCoHom
 for IsCapCategoryObject, IsCapCategory-
 Object, [43](#)
 IsomorphismFromTensorProductWithDual-
 ObjectToInternalHom
 for IsCapCategoryObject, IsCapCategory-
 Object, [40](#)
 IsRigidSymmetricClosedMonoidalCategory
 for IsCapCategory, [107](#)
 IsRigidSymmetricCoclosedMonoidal-
 Category
 for IsCapCategory, [107](#)
 IsStrictMonoidalCategory
 for IsCapCategory, [107](#)
 IsSymmetricClosedMonoidalCategory
 for IsCapCategory, [107](#)
 IsSymmetricCoclosedMonoidalCategory
 for IsCapCategory, [107](#)
 IsSymmetricMonoidalCategory
 for IsCapCategory, [107](#)
 LambdaElimination
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryMorphism, [24](#)
 LambdaIntroduction
 for IsCapCategoryMorphism, [24](#)
 LeftClosedMonoidalCategoriesTest, [93](#)
 LeftClosedMonoidalCategoriesTestWith-
 Given, [94](#)
 LeftClosedMonoidalCoevaluationMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, [9](#)
 LeftClosedMonoidalCoevaluation-
 MorphismWithGivenRange
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, [10](#)
 LeftClosedMonoidalEvaluationForLeft-
 Dual
 for IsCapCategoryObject, [12](#)
 LeftClosedMonoidalEvaluationForLeft-
 DualWithGivenTensorProduct
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, [12](#)
 LeftClosedMonoidalEvaluationMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, [9](#)
 LeftClosedMonoidalEvaluationMorphism-
 WithGivenSource

for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 9
 LeftClosedMonoidalLambdaElimination
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryMorphism, 14
 LeftClosedMonoidalLambdaIntroduction
 for IsCapCategoryMorphism, 14
 LeftClosedMonoidalPostComposeMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 11
 LeftClosedMonoidalPostComposeMorphism-
 WithGivenObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, IsCapCategoryObject,
 11
 LeftClosedMonoidalPreComposeMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 11
 LeftClosedMonoidalPreComposeMorphism-
 WithGivenObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, IsCapCategoryObject,
 11
 LeftCoclosedMonoidalCategoriesTest, 94
 LeftCoclosedMonoidalCategoriesTest-
 WithGiven, 95
 LeftCoclosedMonoidalCoevaluation-
 Morphism
 for IsCapCategoryObject, IsCapCategory-
 Object, 26
 LeftCoclosedMonoidalCoevaluation-
 MorphismWithGivenSource
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 26
 LeftCoclosedMonoidalEvaluationForLeft-
 CoDual
 for IsCapCategoryObject, 28
 LeftCoclosedMonoidalEvaluationForLeft-
 CoDualWithGivenTensorProduct
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 28
 LeftCoclosedMonoidalEvaluationMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, 25
 LeftCoclosedMonoidalEvaluation-
 MorphismWithGivenRange
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 26
 LeftCoclosedMonoidalLambdaElimination
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryMorphism, 31
 LeftCoclosedMonoidalLambdaIntroduction
 for IsCapCategoryMorphism, 31
 LeftCoclosedMonoidalPostCoCompose-
 Morphism
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 27
 LeftCoclosedMonoidalPostCoCompose-
 MorphismWithGivenObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, IsCapCategoryObject,
 28
 LeftCoclosedMonoidalPreCoCompose-
 Morphism
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, 27
 LeftCoclosedMonoidalPreCoCompose-
 MorphismWithGivenObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, IsCapCategoryObject,
 27
 LeftCoDualityTensorProduct-
 CompatibilityMorphism
 for IsCapCategoryObject, IsCapCategory-
 Object, 29
 LeftCoDualityTensorProduct-
 CompatibilityMorphismWith-
 GivenObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, 29
 LeftCoDualOnMorphisms
 for IsCapCategoryMorphism, 28
 LeftCoDualOnMorphismsWithGivenLeftCo-
 Duals
 for IsCapCategoryObject, IsCapCategory-
 Morphism, IsCapCategoryObject, 28
 LeftCoDualOnObjects

- for IsCapCategoryObject, 28
- LeftDistributivityExpanding
 - for IsCapCategoryObject, IsList, 6
- LeftDistributivityExpandingWithGivenObjects
 - for IsCapCategoryObject, IsCapCategoryObject, IsList, IsCapCategoryObject, 6
- LeftDistributivityFactoring
 - for IsCapCategoryObject, IsList, 6
- LeftDistributivityFactoringWithGivenObjects
 - for IsCapCategoryObject, IsCapCategoryObject, IsList, IsCapCategoryObject, 7
- LeftDualOnMorphisms
 - for IsCapCategoryMorphism, 11
- LeftDualOnMorphismsWithGivenLeftDuals
 - for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject, 12
- LeftDualOnObjects
 - for IsCapCategoryObject, 11
- LeftInternalCoHom
 - for IsCapCategoryCell, IsCapCategoryCell, 45
- LeftInternalCoHomOnMorphisms
 - for IsCapCategoryMorphism, IsCapCategoryMorphism, 25
- LeftInternalCoHomOnMorphismsWithGivenLeftInternalCoHoms
 - for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryObject, 25
- LeftInternalCoHomOnObjects
 - for IsCapCategoryObject, IsCapCategoryObject, 25
- LeftInternalCoHomTensorProductCompatibilityMorphism
 - for IsList, 29
- LeftInternalCoHomTensorProductCompatibilityMorphismWithGivenObjects
 - for IsCapCategoryObject, IsList, IsCapCategoryObject, 29
- LeftInternalCoHomToTensorProductAdjunctMorphism
 - for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, 26
- LeftInternalCoHomToTensorProductAdjunctMorphismWithGivenTensorProduct
 - for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject, 27
- LeftInternalHom
 - for IsCapCategoryCell, IsCapCategoryCell, 45
- LeftInternalHomOnMorphisms
 - for IsCapCategoryMorphism, IsCapCategoryMorphism, 9
- LeftInternalHomOnMorphismsWithGivenLeftInternalHoms
 - for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryObject, 9
- LeftInternalHomOnObjects
 - for IsCapCategoryObject, IsCapCategoryObject, 9
- LeftInternalHomToTensorProductAdjunctMorphism
 - for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, 10
- LeftInternalHomToTensorProductAdjunctMorphismWithGivenTensorProduct
 - for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject, 10
- LeftUnitor
 - for IsCapCategoryObject, 4
- LeftUnitorInverse
 - for IsCapCategoryObject, 5
- LeftUnitorInverseWithGivenTensorProduct
 - for IsCapCategoryObject, IsCapCategoryObject, 5
- LeftUnitorWithGivenTensorProduct
 - for IsCapCategoryObject, IsCapCategoryObject, 5
- MonoidalCategoriesTensorProductAndUnitTest, 95
- MonoidalCategoriesTest, 96
- MonoidalPostComposeMorphism
 - for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 36

MonoidalPostCoComposeMorphismWithGivenObjects for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 36	Object, 38
MonoidalPostComposeMorphism for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 21	MorphismFromInternalCoHomToTensorProductWithGivenObjects for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 38
MonoidalPostComposeMorphismWithGivenObjects for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 21	MorphismFromInternalHomToTensorProduct for IsCapCategoryObject, IsCapCategoryObject, 41
MonoidalPreCoComposeMorphism for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 35	MorphismFromInternalHomToTensorProductWithGivenObjects for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 41
MonoidalPreCoComposeMorphismWithGivenObjects for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 36	MorphismFromLeftCoBidual for IsCapCategoryObject, 29
MonoidalPreComposeMorphism for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 20	MorphismFromLeftCoBidualWithGivenLeftCoBidual for IsCapCategoryObject, IsCapCategoryObject, 29
MonoidalPreComposeMorphismWithGivenObjects for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 20	MorphismFromLeftInternalCoHomToTensorProduct for IsCapCategoryObject, IsCapCategoryObject, 30
MorphismFromBidual for IsCapCategoryObject, 42	MorphismFromLeftInternalCoHomToTensorProductWithGivenObjects for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 30
MorphismFromBidualWithGivenBidual for IsCapCategoryObject, IsCapCategoryObject, 42	MorphismFromTensorProductToInternalCoHom for IsCapCategoryObject, IsCapCategoryObject, 43
MorphismFromCoBidual for IsCapCategoryObject, 37	MorphismFromTensorProductToInternalCoHomWithGivenObjects for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 43
MorphismFromCoBidualWithGivenCoBidual for IsCapCategoryObject, IsCapCategoryObject, 37	MorphismFromTensorProductToInternalHom for IsCapCategoryObject, IsCapCategoryObject, 23
MorphismFromInternalCoHomToTensorProduct for IsCapCategoryObject, IsCapCategoryObject, 38	MorphismFromTensorProductToInternalHomWithGivenObjects for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 23

MorphismFromTensorProductToLeft-
 InternalHom
 for IsCapCategoryObject, IsCapCategory-
 Object, [13](#)
 MorphismFromTensorProductToLeft-
 InternalHomWithGivenObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, [13](#)
 MorphismToBidual
 for IsCapCategoryObject, [22](#)
 MorphismToBidualWithGivenBidual
 for IsCapCategoryObject, IsCapCategory-
 Object, [22](#)
 MorphismToCoBidual
 for IsCapCategoryObject, [44](#)
 MorphismToCoBidualWithGivenCoBidual
 for IsCapCategoryObject, IsCapCategory-
 Object, [45](#)
 MorphismToLeftBidual
 for IsCapCategoryObject, [12](#)
 MorphismToLeftBidualWithGivenLeft-
 Bidual
 for IsCapCategoryObject, IsCapCategory-
 Object, [12](#)

 RankMorphism
 for IsCapCategoryObject, [42](#)
 RightDistributivityExpanding
 for IsList, IsCapCategoryObject, [7](#)
 RightDistributivityExpandingWithGiven-
 Objects
 for IsCapCategoryObject, IsList, IsCapCate-
 goryObject, IsCapCategoryObject, [7](#)
 RightDistributivityFactoring
 for IsList, IsCapCategoryObject, [7](#)
 RightDistributivityFactoringWithGiven-
 Objects
 for IsCapCategoryObject, IsList, IsCapCate-
 goryObject, IsCapCategoryObject, [7](#)
 RightUnitor
 for IsCapCategoryObject, [5](#)
 RightUnitorInverse
 for IsCapCategoryObject, [5](#)
 RightUnitorInverseWithGivenTensor-
 Product
 for IsCapCategoryObject, IsCapCategory-
 Object, [5](#)
 RightUnitorWithGivenTensorProduct
 for IsCapCategoryObject, IsCapCategory-
 Object, [5](#)
 RigidSymmetricClosedMonoidal-
 CategoriesTest, [96](#)
 RigidSymmetricCoclosedMonoidal-
 CategoriesTest, [97](#)

 TensorProductDualityCompatibility-
 Morphism
 for IsCapCategoryObject, IsCapCategory-
 Object, [22](#)
 TensorProductDualityCompatibility-
 MorphismWithGivenObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, [23](#)
 TensorProductInternalHomCompatibility-
 Morphism
 for IsList, [22](#)
 TensorProductInternalHomCompatibility-
 MorphismInverse
 for IsList, [41](#)
 TensorProductInternalHomCompatibility-
 MorphismInverseWithGivenObjects
 for IsCapCategoryObject, IsList, IsCapCate-
 goryObject, [41](#)
 TensorProductInternalHomCompatibility-
 MorphismWithGivenObjects
 for IsCapCategoryObject, IsList, IsCapCate-
 goryObject, [22](#)
 TensorProductLeftDualityCompatibility-
 Morphism
 for IsCapCategoryObject, IsCapCategory-
 Object, [13](#)
 TensorProductLeftDualityCompatibility-
 MorphismWithGivenObjects
 for IsCapCategoryObject, IsCapCategory-
 Object, IsCapCategoryObject, IsCap-
 CategoryObject, [13](#)
 TensorProductLeftInternalHom-
 CompatibilityMorphism
 for IsList, [12](#)
 TensorProductLeftInternalHom-
 CompatibilityMorphismWith-

GivenObjects
 for IsCapCategoryObject, IsList, IsCapCategoryObject, 13
 TensorProductOnMorphisms
 for IsCapCategoryMorphism, IsCapCategoryMorphism, 3
 TensorProductOnMorphismsWithGivenTensorProducts
 for IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryObject, 4
 TensorProductOnObjects
 for IsCapCategoryObject, IsCapCategoryObject, 6
 TensorProductToInternalCoHomAdjunctionMap
 for IsObject, 104
 TensorProductToInternalCoHomAdjunctionMapWithGivenInternalCoHom
 for IsObject, 104
 TensorProductToInternalCoHomLeftAdjunctionMap
 for IsObject, 105
 TensorProductToInternalCoHomLeftAdjunctionMapWithGivenInternalCoHom
 for IsObject, 105
 TensorProductToInternalCoHomLeftAdjunctMorphism
 for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, 35
 TensorProductToInternalCoHomLeftAdjunctMorphismWithGivenInternalCoHom
 for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject, 35
 TensorProductToInternalCoHomRightAdjunctMorphism
 for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, 33
 TensorProductToInternalCoHomRightAdjunctMorphismWithGivenInternalCoHom
 for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject, 33
 TensorProductToInternalHomAdjunctionMap
 for IsObject, 104
 TensorProductToInternalHomAdjunctionMapWithGivenInternalHom
 for IsObject, 104
 TensorProductToInternalHomLeftAdjunctionIsomorphism
 for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 19
 TensorProductToInternalHomLeftAdjunctionIsomorphismWithGivenObjects
 for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 19
 TensorProductToInternalHomLeftAdjunctionMap
 for IsObject, 105
 TensorProductToInternalHomLeftAdjunctionMapWithGivenInternalHom
 for IsObject, 105
 TensorProductToInternalHomLeftAdjunctMorphism
 for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, 19
 TensorProductToInternalHomLeftAdjunctMorphismWithGivenInternalHom
 for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryMorphism, IsCapCategoryObject, 19
 TensorProductToInternalHomRightAdjunctionIsomorphism
 for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 17
 TensorProductToInternalHomRightAdjunctionIsomorphismWithGivenObjects
 for IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, IsCapCategoryObject, 17

TensorProductToInternalHomRight- AdjunctMorphism for IsCapCategoryObject, IsCapCategory- Object, IsCapCategoryMorphism, 16	WriteFileForClosedMonoidalStructure, 99 WriteFileForCoclosedMonoidalStructure, 100 WriteFileForLeftClosedMonoidal- Structure, 99 WriteFileForLeftCoclosedMonoidal- Structure, 100 WriteFileForMonoidalStructure, 99
TensorProductToInternalHomRight- AdjunctMorphismWithGiven- InternalHom for IsCapCategoryObject, IsCapCategory- Object, IsCapCategoryMorphism, Is- CapCategoryObject, 17	
TensorProductToLeftInternalCoHom- AdjunctMorphism for IsCapCategoryObject, IsCapCategory- Object, IsCapCategoryMorphism, 26	
TensorProductToLeftInternalCoHom- AdjunctMorphismWithGivenLeft- InternalCoHom for IsCapCategoryObject, IsCapCategory- Object, IsCapCategoryMorphism, Is- CapCategoryObject, 26	
TensorProductToLeftInternalHomAdjunct- Morphism for IsCapCategoryObject, IsCapCategory- Object, IsCapCategoryMorphism, 10	
TensorProductToLeftInternalHom- AdjunctMorphismWithGivenLeft- InternalHom for IsCapCategoryObject, IsCapCategory- Object, IsCapCategoryMorphism, Is- CapCategoryObject, 10	
TensorUnit for IsCapCategory, 6	
TraceMap for IsCapCategoryMorphism, 42	
UniversalPropertyOfCoDual for IsCapCategoryObject, IsCapCategory- Object, IsCapCategoryMorphism, 39	
UniversalPropertyOfDual for IsCapCategoryObject, IsCapCategory- Object, IsCapCategoryMorphism, 24	
UniversalPropertyOfLeftCoDual for IsCapCategoryObject, IsCapCategory- Object, IsCapCategoryMorphism, 30	
UniversalPropertyOfLeftDual for IsCapCategoryObject, IsCapCategory- Object, IsCapCategoryMorphism, 14	