TriangulatedCategoriesForCAP

framework for triangulated categories

1.0

14 April 2018

Kamal Saleh

Kamal Saleh
Email: kamal.saleh@uni-siegen.de
Homepage: Kamal.saleh@uni-siegen.de
Address: Siegen

Contents

Chapter 1

Introduction

1.1 What is this package

bla bla

1.2 Definition of triangulated categories

A triangulated category consists of the following data.

Chapter 2

Triangulated Categories

2.1 GAP Categories

2.1.1 IsCapCategoryTriangle (for IsCapCategoryObject)

▷ `IsCapCategoryTriangle(obj)` (filter)

Returns: `true` or `false`

The GAP category of triangles in a category. Let $\mathscr{C}$ be an additive category and $\Sigma : \mathscr{C} \to \mathscr{C}$ an additive automorphism. A triangle in $\mathscr{C}$ (w.r.t. Σ) is a diagram of the form

$$X \xrightarrow{f} Y \xrightarrow{g} Z \xrightarrow{h} \Sigma X$$

such that the compositions of fg, gh and $h\Sigma f$ are zero. Such a triangle will be denoted by $\mathrm{Tr}(f,g,h)$.

2.1.2 IsCapCategoryTrianglesMorphism (for IsCapCategoryMorphism)

▷ `IsCapCategoryTrianglesMorphism(mor)` (filter)

Returns: `true` or `false`

The GAP category of morphisms of triangles. Let T_1, T_2 be two triangles in the additive category $\mathscr{C}$. A morphism of triangles is a commutative diagram

$$\begin{array}{ccccccc}
X_1 & \xrightarrow{f_1} & Y_1 & \xrightarrow{g} & Z_1 & \xrightarrow{h} & \Sigma X_1 \\
{\scriptstyle u}\downarrow & & \downarrow{\scriptstyle v} & & \downarrow{\scriptstyle w} & & \downarrow{\scriptstyle \Sigma u} \\
X_2 & \xrightarrow[f_2]{} & Y_2 & \xrightarrow[g_2]{} & Z_2 & \xrightarrow[h_2]{} & \Sigma X_2
\end{array}$$

The triangles and their morphisms define with obvious composition and identities an additive category. We denote this category by $\mathrm{Triangles}(\mathscr{C})$ and called the category of triangles in $\mathscr{C}$.

2.1.3 IsCapCategoryExactTriangle (for IsCapCategoryTriangle)

▷ `IsCapCategoryExactTriangle(obj)` (filter)

Returns: `true` or `false`

The GAP category of exact triangles. An exact triangle in a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ is a triangle in $\mathscr{C}$ that belongs to the class of exact triangles in $\mathscr{C}$. I.e., a triangle that is isomorphismic to some standard exact triangle in $\mathscr{C}$.

2.1.4 IsCapCategoryStandardExactTriangle (for IsCapCategoryExactTriangle)

▷ `IsCapCategoryStandardExactTriangle(obj)` (filter)

Returns: `true` or `false`

The GAP category of standard exact triangles. A standard exact triangle in a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ is a triangle in $\mathscr{C}$ that belongs to the class of standard exact triangles the defines the triangulated structure of $\mathscr{C}$.

2.2 Constructors

2.2.1 CreateTriangle (for IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `CreateTriangle(f, g, h)` (operation)

The arguments are three morphisms f,g,h in a triangulated category $\mathscr{C}$ such that $\mathrm{Range}(f) = \mathrm{Source}(g), \mathrm{Range}(g) = \mathrm{Source}(h), \mathrm{Range}(h) = \Sigma \mathrm{Source}(f)$. The output is the triangle $\mathrm{Tr}(f,g,h)$ as an object in $\mathrm{Triangles}(\mathscr{C})$.

2.2.2 CreateExactTriangle (for IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `CreateExactTriangle(f, g, h)` (operation)

The arguments are three morphisms f,g,h in a triangulated category $\mathscr{C}$ such that $\mathrm{Range}(f) = \mathrm{Source}(g), \mathrm{Range}(g) = \mathrm{Source}(h), \mathrm{Range}(h) = \Sigma \mathrm{Source}(f)$. The output is the exact triangle $\mathrm{Tr}(f,g,h)$ as an object in $\mathrm{Triangles}(\mathscr{C})$.

2.2.3 CreateStandardExactTriangle (for IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `CreateStandardExactTriangle(f, g, h)` (operation)

The arguments are three morphisms f,g,h in a triangulated category $\mathscr{C}$ such that $\mathrm{Range}(f) = \mathrm{Source}(g), \mathrm{Range}(g) = \mathrm{Source}(h), \mathrm{Range}(h) = \Sigma \mathrm{Source}(f)$. The output is the standard exact triangle $\mathrm{Tr}(f,g,h)$ as an object in $\mathrm{Triangles}(\mathscr{C})$.

2.2.4 CreateTrianglesMorphism (for IsCapCategoryTriangle, IsCapCategoryTriangle, IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `CreateTrianglesMorphism(T1, T2, u, v, w)` (operation)

The arguments are two triangles T_1, T_2 and three morphisms u,v and w. The output the triangles morphism $T_1 \to T_2$ given by these morphisms.

2.3 Categorical Operations

2.3.1 LiftColift (for IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `LiftColift(alpha, beta, gamma, delta)` (operation)

Returns: a morphism in $\mathrm{Hom}(Y,W)$ + `{fail}`

The arguments are four morphisms $\alpha : X \to Y, \beta : X \to W, \gamma : Y \to Z$ and $\delta : W \to Z$. Such that $\gamma \circ \alpha \sim_{X,Z} \delta \circ \beta$. The output is a morphism $\lambda : Y \to W$ that is a colift of β along α and is a lift of γ along δ. I.e., $\lambda \circ \alpha \sim_{X,W} \beta$ and $\delta \circ \lambda \sim_{Y,Z} \gamma$; or fail if such a morphism doesn't exist.

$$\begin{tikzcd}
& Y \arrow[dd, dashed, "\lambda"] \arrow[dr, "\gamma"] & \\
X \arrow[ur, "\alpha"] \arrow[dr, "\beta"'] & & Z \\
& W \arrow[ur, "\delta"'] &
\end{tikzcd}$$

2.3.2 AddLiftColift (for IsCapCategory, IsFunction)

▷ `AddLiftColift(C, F)` (operation)

Returns: nothing

The arguments are a category $\mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation LiftColift. The function F maps a quadruple $\alpha, \beta, \gamma, \delta$ to a morphism λ as described above if it exists or to fail otherwise.

2.3.3 ShiftOfObject (for IsCapCategoryObject)

▷ `ShiftOfObject(X)` (operation)

Returns: an object

The argument is an object X in a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$. The output is ΣX.

2.3.4 AddShiftOfObject (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation ShiftOfObject. The function F maps an object X to ΣX.

2.3.5 ShiftOfMorphism (for IsCapCategoryMorphism)

▷ `ShiftOfMorphism(f)` (operation)

Returns: a morphism

The argument is a morphism f in a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$. The output is Σf.

2.3.6 AddShiftOfMorphism (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation ShiftOfMorphism. The function F maps a morphism f to Σf.

2.3.7 ReverseShiftOfObject (for IsCapCategoryObject)

▷ `ReverseShiftOfObject(X)` (operation)

Returns: an object

The argument is an object X in a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$. The output is $\Sigma^{-1}X$.

2.3.8 AddReverseShiftOfObject (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation ReverseShiftOfObject. The function F maps an object X to $\Sigma^{-1}X$.

2.3.9 ReverseShiftOfMorphism (for IsCapCategoryMorphism)

▷ `ReverseShiftOfMorphism(f)` (operation)

Returns: a morphism

The argument is a morphism f in a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$. The output is $\Sigma^{-1}f$.

2.3.10 AddReverseShiftOfMorphism (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation ShiftOfMorphism. The function F maps a morphism f to $\Sigma^{-1}f$.

2.3.11 ShiftExpandingIsomorphismWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject)

▷ `ShiftExpandingIsomorphismWithGivenObjects(X, L, Y)` (operation)

Returns: a morphism

The arguments are list $L = [A_1, \dots, A_n]$ and two objects $X = \Sigma \bigoplus_i A_i, Y = \bigoplus_i \Sigma A_i$. The output is the isomorphism $X \to Y$ associated to Σ.

2.3.12 AddShiftExpandingIsomorphismWithGivenObjects (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation ShiftExpandingIsomorphismWithGivenObjects.

2.3.13 ShiftExpandingIsomorphism (for IsList)

▷ `ShiftExpandingIsomorphism(L)` (operation)

Returns: a morphism

The argument is a list $L = [A_1, \dots, A_n]$. The output is the isomorphism $X \to Y$ associated to Σ, where $X = \Sigma \bigoplus_i A_i$ and $Y = \bigoplus_i \Sigma A_i$

2.3.14 ShiftFactoringIsomorphismWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject)

▷ `ShiftFactoringIsomorphismWithGivenObjects(Y, L, X)` (operation)

Returns: a morphism

The arguments are list $L = [A_1, \dots, A_n]$ and two objects $Y = \bigoplus_i \Sigma A_i, X = \Sigma \bigoplus_i A_i$. The output is the isomorphism $Y \to X$ associated to Σ.

2.3.15 AddShiftFactoringIsomorphismWithGivenObjects (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation AddShiftFactoringIsomorphismWithGivenObjects.

2.3.16 ShiftFactoringIsomorphism (for IsList)

▷ `ShiftFactoringIsomorphism(L)` (operation)

Returns: a morphism

The argument is a list $L = [A_1, \dots, A_n]$. The output is the isomorphism $Y \to X$ associated to Σ, where $Y = \bigoplus_i \Sigma A_i$ and $X = \Sigma \bigoplus_i A_i$.

2.3.17 ReverseShiftExpandingIsomorphismWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject)

▷ `ReverseShiftExpandingIsomorphismWithGivenObjects(X, L, Y)` (operation)

Returns: a morphism

The arguments are list $L = [A_1, \dots, A_n]$ and two objects $X = \Sigma^{-1} \bigoplus_i A_i, Y = \bigoplus_i \Sigma^{-1} A_i$. The output is the isomorphism $X \to Y$ associated to Σ^{-1}.

2.3.18 AddReverseShiftExpandingIsomorphismWithGivenObjects (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation ReverseShiftExpandingIsomorphismWithGivenObjects.

2.3.19 ReverseShiftExpandingIsomorphism (for IsList)

▷ `ReverseShiftExpandingIsomorphism(L)` (operation)

Returns: a morphism

The argument is a list $L = [A_1, \dots, A_n]$. The output is the isomorphism $X \to Y$ associated to Σ, where $X = \Sigma \bigoplus_i A_i$ and $Y = \bigoplus_i \Sigma A_i$

2.3.20 ReverseShiftFactoringIsomorphismWithGivenObjects (for IsCapCategoryObject, IsList, IsCapCategoryObject)

▷ `ReverseShiftFactoringIsomorphismWithGivenObjects(Y, L, X)` (operation)

Returns: a morphism

The arguments are list $L = [A_1, \dots, A_n]$ and two objects $Y = \bigoplus_i \Sigma^{-1} A_i, X = \Sigma^{-1} \bigoplus_i A_i$. The output is the isomorphism $Y \to X$ associated to Σ^{-1}.

2.3.21 AddReverseShiftFactoringIsomorphismWithGivenObjects (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation ReverseShiftFactoringIsomorphismWithGivenObjects.

2.3.22 ReverseShiftFactoringIsomorphism (for IsList)

▷ `ReverseShiftFactoringIsomorphism(L)` (operation)

Returns: a morphism

The argument is a list $L = [A_1, \dots, A_n]$. The output is the isomorphism $Y \to X$ associated to Σ^{-1}, where $Y = \bigoplus_i \Sigma^{-1} A_i$ and $X = \Sigma^{-1} \bigoplus_i A_i$.

2.3.23 IsomorphismIntoShiftOfReverseShift (for IsCapCategoryObject)

▷ `IsomorphismIntoShiftOfReverseShift(X)` (operation)

Returns: a morphism

The argument is an object X. The output is the isomorphism $X \to (\Sigma \circ \Sigma^{-1})X$.

2.3.24 AddIsomorphismIntoShiftOfReverseShift (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation IsomorphismIntoShiftOfReverseShift.

2.3.25 IsomorphismIntoReverseShiftOfShift (for IsCapCategoryObject)

▷ `IsomorphismIntoReverseShiftOfShift(X)` (operation)

Returns: a morphism

The argument is an object X. The output is the isomorphism $X \to (\Sigma^{-1} \circ \Sigma)X$.

2.3.26 AddIsomorphismIntoReverseShiftOfShift (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation IsomorphismIntoReverseShiftOfShift.

2.3.27 IsomorphismFromShiftOfReverseShift (for IsCapCategoryObject)

▷ `IsomorphismFromShiftOfReverseShift(X)` (operation)

Returns: a morphism

The argument is an object X. The output is the isomorphism $(\Sigma \circ \Sigma^{-1})X \to X$.

2.3.28 AddIsomorphismFromShiftOfReverseShift (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation IsomorphismFromShiftOfReverseShift.

2.3.29 IsomorphismFromReverseShiftOfShift (for IsCapCategoryObject)

▷ `IsomorphismFromReverseShiftOfShift(X)` (operation)

Returns: a morphism

The argument is an object X. The output is the isomorphism $(\Sigma^{-1} \circ \Sigma)X \to X$.

2.3.30 AddIsomorphismFromReverseShiftOfShift (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation IsomorphismFromReverseShiftOfShift.

2.3.31 IsStandardExactTriangle (for IsCapCategoryTriangle)

▷ `IsStandardExactTriangle(T)` (property)

Returns: `true` or `false`

The argument is a triangle $T \in \text{Triangles}(\mathscr{C})$, where $\mathscr{C}$ a triangulated category $\mathscr{C}$. The output is true if T is standard exact triangle, otherwise the output is false.

2.3.32 AddIsStandardExactTriangle (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation IsStandardExactTriangle.

2.3.33 IsExactTriangle (for IsCapCategoryTriangle)

▷ `IsExactTriangle(T)` (property)

Returns: `true` or `false`

The argument is a triangle $T \in \text{Triangles}(\mathscr{C})$, where $\mathscr{C}$ a triangulated category $\mathscr{C}$. The output is true if T is exact triangle, otherwise the output is false.

2.3.34 AddIsExactTriangle (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation IsExactTriangle.

2.3.35 CompleteMorphismToStandardExactTriangle (for IsCapCategoryMorphism)

▷ `CompleteMorphismToStandardExactTriangle(T)` (operation)

The argument is morphism f in a triangulated category $\mathscr{C}$. The output is a standard exact triangle which exists by the axioms of triangulated structure. We denote this standard exact triangle by $\text{Tr}(f)$.

$$X \xrightarrow{f} Y \xrightarrow{\alpha(f)} C(f) \xrightarrow{\beta(f)} \Sigma X$$

2.3.36 AddCompleteMorphismToStandardExactTriangle (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation CompleteMorphismToStandardExactTriangle.

2.3.37 IsomorphismIntoStandardExactTriangle (for IsCapCategoryExactTriangle)

▷ `IsomorphismIntoStandardExactTriangle(T)` (attribute)

Returns: morphism in $\text{Hom}(T, \text{Tr}(f))$

The argument is an exact triangle $T = \text{Tr}(f, g, h) \in \text{Triangles}(\mathscr{C})$, where $\mathscr{C}$ a triangulated category. The output is a triangles isomorphism into the standard exact triangle $\text{Tr}(f)$. The first two morphisms of the output should be identity morphisms.

$$\begin{tikzcd} T: & X \arrow[r, "f"] \arrow[d, "\mathrm{id}_X"'] & Y \arrow[r, "g"] \arrow[d, "\mathrm{id}_Y"] & Z \arrow[r, "h"] \arrow[d, dashed, "\lambda"] & \Sigma X \arrow[d, "\mathrm{id}_{\Sigma X}"] \\ \mathrm{Tr}(f): & X \arrow[r, "f"'] & Y \arrow[r, "\alpha(f)"'] & C(f) \arrow[r, "\beta(f)"'] & \Sigma X \end{tikzcd}$$

2.3.38 AddIsomorphismIntoStandardExactTriangle (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation IsomorphismIntoStandardExactTriangle.

2.3.39 IsomorphismFromStandardExactTriangle (for IsCapCategoryExactTriangle)

▷ `IsomorphismFromStandardExactTriangle(T)` (attribute)

Returns: morphism in $\mathrm{Hom}(\mathrm{Tr}(f), T)$

The argument is an exact triangle $T = \mathrm{Tr}(f, g, h) \in \mathrm{Triangles}(\mathscr{C})$, where $\mathscr{C}$ a triangulated category. The output is a triangles isomorphism from the standard exact triangle $\mathrm{Tr}(f)$. The first two morphisms of the output should be identity morphisms.

$$\begin{tikzcd} T: & X \arrow[r, "f"] & Y \arrow[r, "g"] & Z \arrow[r, "h"] & \Sigma X \\ \mathrm{Tr}(f): & X \arrow[r, "f"'] \arrow[u, "\mathrm{id}_X"] & Y \arrow[r, "\alpha(f)"'] \arrow[u, "\mathrm{id}_Y"'] & C(f) \arrow[r, "\beta(f)"'] \arrow[u, dashed, "\lambda"'] & \Sigma X \arrow[u, "\mathrm{id}_{\Sigma X}"'] \end{tikzcd}$$

2.3.40 AddIsomorphismFromStandardExactTriangle (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation IsomorphismFromStandardExactTriangle.

2.3.41 RotationOfStandardExactTriangle (for IsCapCategoryStandardExactTriangle)

▷ `RotationOfStandardExactTriangle(T)` (attribute)

Returns: an exact triangle

The argument is a standard exact triangle $T = \mathrm{Tr}(f, \alpha(f), \beta(f)) \in \mathrm{Triangles}(\mathscr{C})$, where $\mathscr{C}$ a triangulated category $\mathscr{C}$. The output is the exact triangle $\mathrm{Tr}(\alpha(f), \beta(f), -\Sigma f)$. If no methods for IsomorphismFromStandardExactTriangle and IsomorphismIntoStandardExactTriangle are installed for the category, then the two attributes IsomorphismFromStandardExactTriangle and IsomorphismIntoStandardExactTriangle should be set for the output $\mathrm{Tr}(\alpha(f), \beta(f), -\Sigma f)$.

2.3.42 AddRotationOfStandardExactTriangle (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation RotationOfStandardExactTriangle.

2.3.43 RotationOfExactTriangle (for IsCapCategoryExactTriangle)

▷ `RotationOfExactTriangle(T)` (attribute)

Returns: an exact triangle

The argument is a exact triangle $T = \mathrm{Tr}(f,g,h) \in \mathrm{Triangles}(\mathscr{C})$, where $\mathscr{C}$ a triangulated category $\mathscr{C}$. The output is the exact triangle $\mathrm{Tr}(g,h,-\Sigma f)$.

2.3.44 AddRotationOfExactTriangle (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation RotationOfExactTriangle.

2.3.45 CompleteToMorphismOfStandardExactTriangles (for IsCapCategoryStandardExactTriangle, IsCapCategoryStandardExactTriangle, IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `CompleteToMorphismOfStandardExactTriangles(T)` (operation)

Returns: a triangles morphism

The arguments are two standard exact triangles $T_1 = \mathrm{Tr}(f_1), T_2 = \mathrm{Tr}(f_2) \in \mathrm{Triangles}(\mathscr{C})$ and two morphisms u, v in a triangulated category $\mathscr{C}$ with $v \circ f_1 \sim_{X_1,Y_2} f_2 \circ u$. The output is a triangles morphism $T_1 \to T_2$

$$\begin{tikzcd}
X_1 \arrow[r, "f_1"] \arrow[d, "u"'] & Y_1 \arrow[r, "\alpha(f_1)"] \arrow[d, "v"] & C(f_1) \arrow[r, "\beta(f_1)"] \arrow[d, dashed, "w"] & \Sigma X_1 \arrow[d, "\Sigma u"] \\
X_2 \arrow[r, "f_2"'] & Y_2 \arrow[r, "\alpha(f_2)"'] & C(f_2) \arrow[r, "\beta(f_2)"'] & \Sigma X_2
\end{tikzcd}$$

2.3.46 AddCompleteToMorphismOfStandardExactTriangles (for IsCapCategory, IsFunction)

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

Returns: nothing

The arguments are a triangulated category $\mathscr{C}$ w.r.t. $\Sigma : \mathscr{C} \to \mathscr{C}$ and a function F. This operation adds the given function F to the category for the basic operation CompleteToMorphismOfStandardExactTriangles.

2.3.47 OctahedralAxiom (for IsCapCategoryMorphism, IsCapCategoryMorphism)

▷ `OctahedralAxiom(T)` (operation)

Returns: a triangle

The arguments are morphisms $f : X \to Y, g : Y \to Z$ in the triangulated category $\mathscr{C}$. The output is an exact triangle $T = \mathrm{Tr}(u, v, w)$ such that the following diagram is commutative.

$$
\begin{tikzcd}
X \arrow[r, "f"] \arrow[dr, bend right, "h:=g\circ f"'] & Y \arrow[r, "\alpha(f)"] \arrow[d, "g"] & C(f) \arrow[r, "\beta(f)"] \arrow[d, dashed, "u"] & \Sigma X \arrow[d, "\mathrm{id}_{\Sigma X}"] \\
 & Z \arrow[r, "\alpha(h)"] \arrow[d, "\alpha(g)"'] & C(h) \arrow[r, "\beta(h)"] \arrow[d, dashed, "v"] & \Sigma X \arrow[d, "\Sigma f"] \\
 & C(g) \arrow[r, "\mathrm{id}_{C(g)}"] \arrow[d, "\beta(g)"'] & C(g) \arrow[r, "\beta(g)"] \arrow[d, dashed, "w"] & \Sigma Y \\
 & \Sigma Y \arrow[r, "\Sigma\alpha(f)"'] & \Sigma C(f) &
\end{tikzcd}
$$

If no methods for IsomorphismFromStandardExactTriangle and IsomorphismIntoStandardExactTriangle are installed for the category, then the two attributes IsomorphismFromStandardExactTriangle and IsomorphismIntoStandardExactTriangle should be set for the output T.

Index