Transformations de graphes décorés

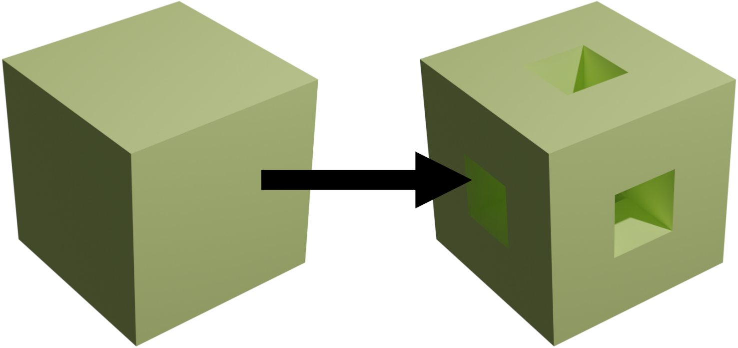
Application aux opérations de modélisation géométrique

Pascale Le Gall[1] Romain Pascual[1]
Hakim Belhaouari[2] Agnès Arnould[2]

[1] Laboratoire MICS, CentraleSupélec, Université Paris-Saclay
[2] Laboratoire XLIM, Université de Poitiers

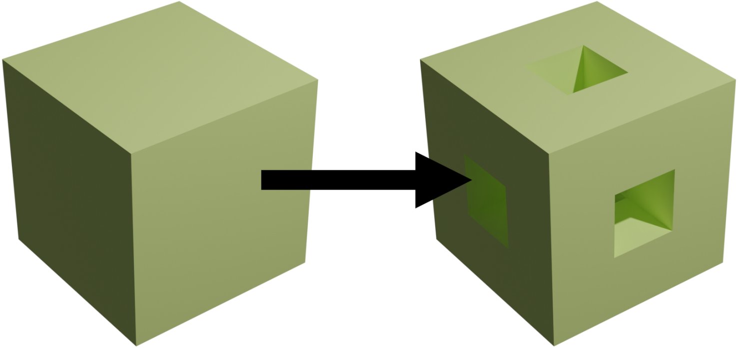
22 juin 2023

Le contexte : comment spécifier et prototyper une opération de modélisation géométrique ?



Opération de Menger

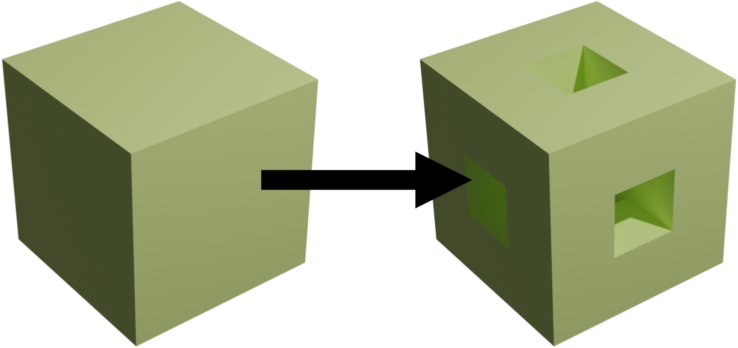
Le contexte : comment spécifier et prototyper une opération de modélisation géométrique ?



Opération de Menger

Structure d'un objet = modèle combinatoire = graphe

Le contexte : comment spécifier et prototyper une opération de modélisation géométrique ?



Opération de Menger

Structure d'un objet = modèle combinatoire = graphe
Opération = transformation d'objets = réécriture de graphes

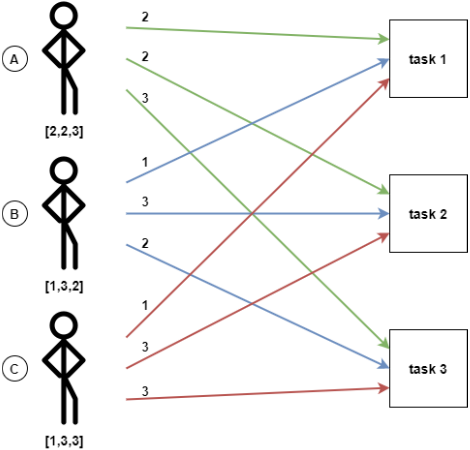
Un graphe ?

Un graphe ?

Un ensemble d'objets ou d'éléments et des relations

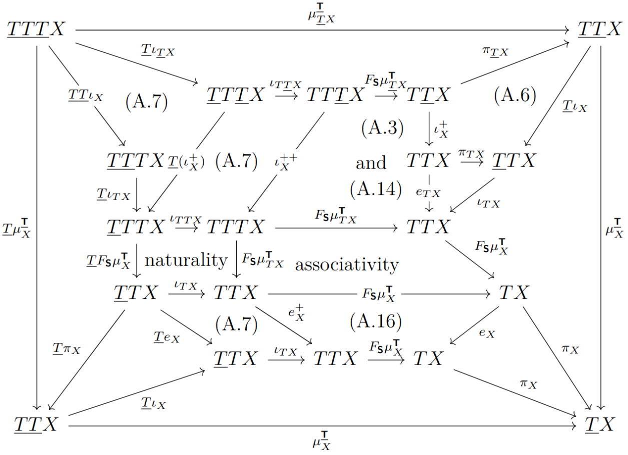
Un graphe ?

Un ensemble d'objets ou d'éléments et des relations



Un graphe ?

Un ensemble d’objets ou d’éléments et des relations



Réécriture (ou transformation)

Réécriture (ou transformation)

Réécriture de mots :

- un alphabet Σ
- ensemble de règles $u \to v$ avec u et v mots de Σ^*

Réécriture (ou transformation)

Réécriture de mots :

- un alphabet Σ
- ensemble de règles $u \to v$ avec u et v mots de Σ^*

orant $\to$ eur

Réécriture (ou transformation)

Réécriture de mots :

- un alphabet Σ
- ensemble de règles $u \to v$ avec u et v mots de Σ^*

orant $\to$ eur

Vous␣êtes␣doctorants␣!

Réécriture (ou transformation)

Réécriture de mots :

- un alphabet Σ
- ensemble de règles $u \to v$ avec u et v mots de Σ^*

orant $\to$ eur

Vous␣êtes␣doctorants␣! $\to$ Vous␣êtes␣docteurs␣!

Comment réécrire un graphe ?

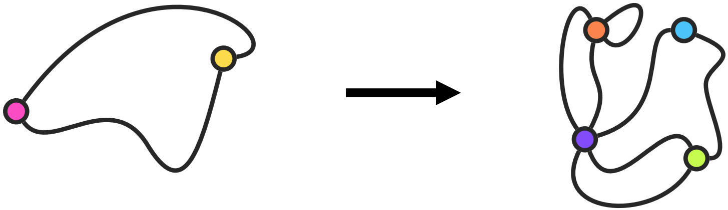
Comment réécrire un graphe ? [1]

$$\bullet \overset{*}{-} {\color{blue}\bullet} \overset{o}{-} \bullet \overset{r}{-} \bullet \overset{a}{-} \bullet \overset{n}{-} \bullet \overset{t}{-} {\color{blue}\bullet} \overset{*}{-} \bullet \quad \longrightarrow \quad \bullet \overset{*}{-} {\color{blue}\bullet} \overset{e}{-} \bullet \overset{u}{-} \bullet \overset{r}{-} {\color{blue}\bullet} \overset{*}{-} \bullet$$

1. inspiré de EHRIG 1979

Comment réécrire un graphe ? [1]

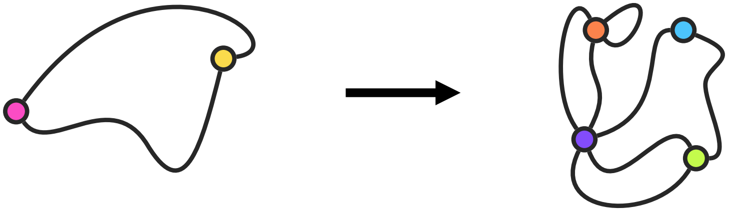
$$\bullet \overset{*}{-} {\color{blue}\bullet} \overset{o}{-} \bullet \overset{r}{-} \bullet \overset{a}{-} \bullet \overset{n}{-} \bullet \overset{t}{-} {\color{blue}\bullet} \overset{*}{-} \bullet \quad \rightarrow \quad \bullet \overset{*}{-} {\color{blue}\bullet} \overset{e}{-} \bullet \overset{u}{-} \bullet \overset{r}{-} {\color{blue}\bullet} \overset{*}{-} \bullet$$



1. inspiré de Ehrig 1979

Comment réécrire un graphe ? [1]

$$\bullet \overset{*}{—} \bullet \overset{o}{—} \bullet \overset{r}{—} \bullet \overset{a}{—} \bullet \overset{n}{—} \bullet \overset{t}{—} \bullet \overset{*}{—} \bullet \quad \rightarrow \quad \bullet \overset{*}{—} \bullet \overset{e}{—} \bullet \overset{u}{—} \bullet \overset{r}{—} \bullet \overset{*}{—} \bullet$$

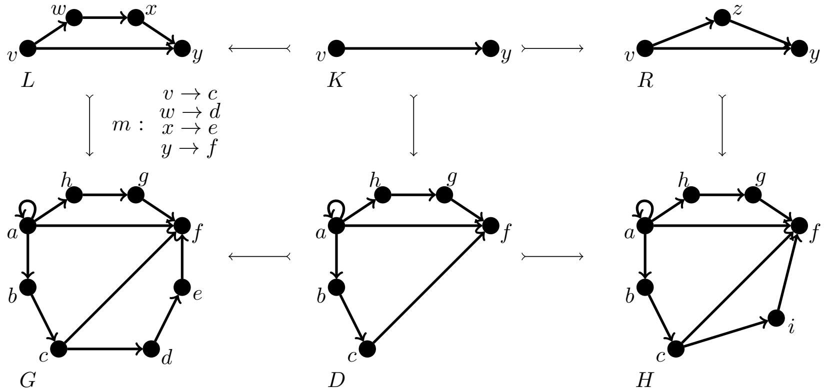


Pas de notion de début et de fin d'un graphe

$\Rightarrow$ Identifier des éléments de "recollement"

1. inspiré de EHRIG 1979

Opérations comme des transformations de graphes



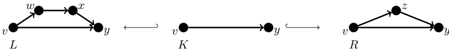
Approche dite par "double pushout" fondée sur la théorie des catégories

$$
\begin{tikzcd}
X_1 \arrow[rr, "f_{12}"] \arrow[dr, "f_{13}"] \arrow[dd, "h_1"'] & & X_2 \arrow[dr, "f_{23}"] \arrow[dd, "h_2"] & \\
& X_3 \arrow[rr, "f_{34}"] \arrow[dd, "h_3"'] & & X_4 \arrow[dd, "h_4"] \\
Y_1 \arrow[rr, "g_{12}"] \arrow[dr, "g_{13}"'] & & Y_2 \arrow[dr, "g_{24}"] & \\
& Y_3 \arrow[rr, "g_{34}"'] & & Y_4
\end{tikzcd}
$$

Théorie des catégories = Art d'écrire des diagrammes combinant des carrés bien agencés

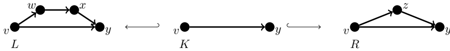
Organisation de la journée

- Session 1 : Théorie des catégories et transformations de graphes

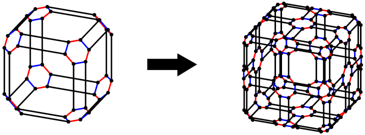


Organisation de la journée

- Session 1 : Théorie des catégories et transformations de graphes

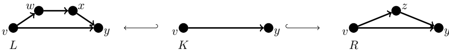


- Session 2 : Opérations de modélisation géométrique comme spécialisation des transformations de graphes

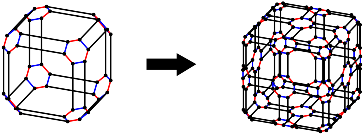


Organisation de la journée

- Session 1 : Théorie des catégories et transformations de graphes



- Session 2 : Opérations de modélisation géométrique comme spécialisation des transformations de graphes



- Session 3 : TP avec Jerboa une plate-forme dédiée à la conception d'opérations de modélisation géométrique

Plan de la session 1

Introduction

- Origine dans les années 1950

- Met l'accent sur les relations entre les entités étudiées, plutôt que les entités elles-mêmes

  $\Rightarrow$ *abstraction* des détails relatifs aux entités elles-mêmes

- très utilisée en informatique,

  ... ici pour formaliser la réécriture de graphes

Les catégories pour les nuls

Des objets
Des relations réflexives et transitives

Et c'est tout

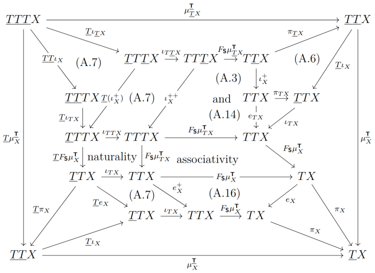
Les catégories pour les nuls

Des objets
Des relations réflexives et transitives

Et c'est tout

Enfin, pas tout à fait

Des diagrammes
Et des compositions de concepts



Catégories

Une catégorie $\mathcal{A}$ consiste en :

- une collection $\mathcal{O}(\mathcal{A})$ d'objets ;
- pour tout $A, B \in \mathcal{O}(\mathcal{A})$, une collection $\mathcal{A}(A, B)$ de morphismes de A vers B.
- pour tout $A, B, C \in \mathcal{O}(\mathcal{A})$, une fonction

  $$\left\{ \begin{array}{ccc} \mathcal{A}(B,C) \times \mathcal{A}(A,B) & \to & \mathcal{A}(A,C) \\ (g,f) & \mapsto & g \circ f \end{array} \right.$$

  appelée composition, satisfaisant :
  pour tout $f \in \mathcal{A}(A,B)$, $g \in \mathcal{A}(B,C)$ et $h \in \mathcal{A}(C,D)$,

  $$(h \circ g) \circ f = h \circ (g \circ f)$$

- pour tout $A \in \mathcal{O}(\mathcal{A})$, il existe un élément 1_A de $\mathcal{A}(A,A)$, appelé l'identité sur A, tel queue :
  pour tout $f \in \mathcal{A}(A,B)$, $f \circ 1_A = f = 1_B \circ f$

Commentaires

- Collection = expression pour désigner n'importe quel tas d'objets comme une classe, un ensemble, un type, etc.

  Une catégorie est dite petite si ses collections d'objets et de morphismes sont des ensembles.

- Une catégorie peut être vue comme un *graphe orienté* particulier
  - boucle sur chaque nœud/objet (identité)
  - et composition des arcs/morphismes (associativité)

- Notations :
  - $A \in \mathcal{A}$ pour $A \in \mathcal{O}(\mathcal{A})$
  - $f \colon A \to B$ ou $A \xrightarrow{f} B$ pour $f \in \mathcal{A}(A, B)$.
  - $\mathrm{Hom}_{\mathcal{A}}(A, B)$ ou $\mathrm{Mor}_{\mathcal{A}}(A, B)$ pour $\mathcal{A}(A, B)$

Catégories particulières

- Catégorie vide $\emptyset$: aucun objet et aucun morphisme.

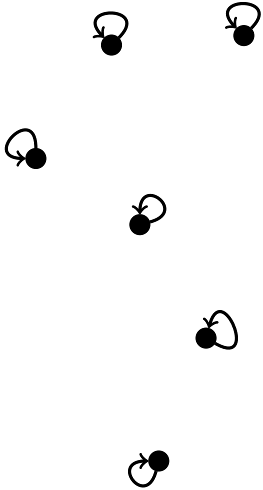
- Catégorie triviale $*$: un seul objet $*$ avec un seul morphisme identité 1_*.

- Catégorie produit $\mathcal{A} \times \mathcal{B}$ avec pour objets les couples (A, B) dans $\mathcal{A} \times \mathcal{B}$ et les morphismes les couples de morphismes $(f : A \to A', g : B \to B')$

- $\mathcal{A}^{op}$ la catégorie opposée avec pour objets $\mathcal{O}(\mathcal{A})$ et pour morphismes $f^{op} : B \to A \in \mathcal{A}^{op}(B, A)$ pour $f \in \mathcal{A}(A, B)$
Observons que $f^{op} \circ g^{op} = (g \circ f)^{op}$ et $1_A^{op} = 1_A$.

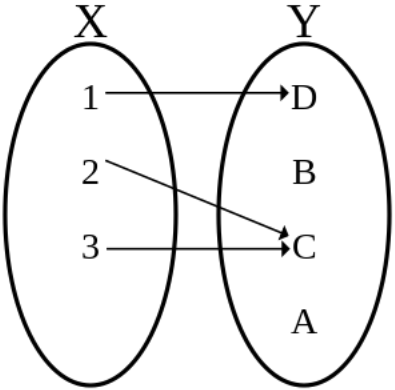
Catégories discrètes

Les seuls morphismes sont les morphismes identités associés aux éléments d'un ensemble.



Catégorie des ensembles $\mathbf{Set}$

- Objets = ensembles
- Morphismes = fonctions entre les ensembles

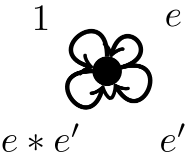


Monoïdes vus comme des catégories

Un monoïde $(E, *, 1)$ est un ensemble E muni d'une opération interne $*$ associative et d'un élément neutre 1 pour cette opération

Monoïdes vus comme des catégories

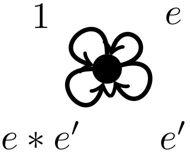
Un monoïde $(E, *, 1)$ est un ensemble E muni d'une opération interne $*$ associative et d'un élément neutre 1 pour cette opération



Monoïde = catégorie à un seul élément

Monoïdes vus comme des catégories

Un monoïde $(E, *, 1)$ est un ensemble E muni d'une opération interne $*$ associative et d'un élément neutre 1 pour cette opération



Monoïde = catégorie à un seul élément

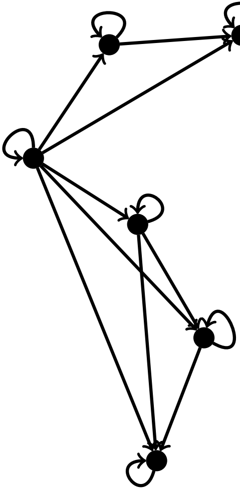
Une autre idée ?

Préordres vus comme des catégories

Un préordre $(E, \leq)$ est un ensemble E muni d'une relation binaire $\leq$ réflexive et transitive.

Préordres vus comme des catégories

Un préordre $(E, \leq)$ est un ensemble E muni d'une relation binaire $\leq$ réflexive et transitive.



Catégorie *fine* = au plus un morphisme entre deux éléments

Tout est catégorie

La plupart des structures mathématiques (groupes, anneaux, espaces topologiques, etc) peuvent être vues comme des catégories

y compris les différentes classes de graphes

Isomorphismes, monomorphismes et épimorphismes

- $f\colon A \to B$ dans $\mathcal{A}$ est un isomorphisme s'il existe $g\colon B \to A$ dans $\mathcal{A}$ vérifiant
$$g \circ f = 1_A \text{ et } f \circ g = 1_B$$
g est alors appelé l'inverse de f.

- $f\colon A \to B$ dans $\mathcal{A}$ est un monomorphisme ou un mono, $f\colon A \hookrightarrow B$, s'il est simplifiable à gauche i.e., pour tout objet $X \in \mathcal{A}$ et pour toute paire de morphismes $g, g'\colon X \to A$,
$$\text{si } f \circ g = f \circ g' \text{ alors } g = g'$$

- $f\colon A \to B$ dans $\mathcal{A}$ est un épimorphisme, ou un épi, s'il est simplifiable à droite i.e., pour tout objet $X \in \mathcal{A}$ et pour toute paire de morphismes $g, g'\colon B \to X$,
$$\text{si } g \circ f = g' \circ f \text{ alors } g = g'$$

Objets initiaux et terminaux

Un objet $\varnothing_{\mathcal{A}}$ d'une catégorie $\mathcal{A}$ est dit initial si pour tout objet $A \in \mathcal{A}$, il existe un unique morphisme $\varnothing_{\mathcal{A}} \to A$.

Par dualité, un objet $1_{\mathcal{A}}$ d'une catégorie $\mathcal{A}$ est dit terminal si pour tout objet $A \in \mathcal{A}$, il existe un unique morphisme $A \to 1_{\mathcal{A}}$.

Les objets initiaux et terminaux d'une catégorie $\mathcal{A}$, *lorsqu'ils existent*, sont uniques à isomorphisme près.

Notations

- L'objet initial de $\mathcal{A}$ est noté $\varnothing_{\mathcal{A}}$.
- $!_A$ dénote l'unique morphisme $A \to 1_{\mathcal{A}}$, pour A objet de $\mathcal{A}$.

Exemples

- L'unique objet de la catégorie triviale $*$ est à la fois initial et terminal.

- Dans $\mathbf{Set}$, l'ensemble vide $\emptyset$ est initial et tous les singletons sont terminaux.

Diagrammes commutatifs

Un diagramme exprime des équations entre morphismes de mêmes source et cible.

$$\begin{tikzcd}
B \arrow[d, "g"'] & A \arrow[l, "f"'] \arrow[d, "i"] \\
C \arrow[r, "h"'] & D
\end{tikzcd}$$

exprime l'égalité

$$h \circ g \circ f = i$$

Un diagramme est dit commutatif si tous les couples de chemins de même source et même cible définissent deux morphismes égaux par composition des morphismes le long des chemins.

Diagrammes remarquables

- carrés

$$\begin{tikzcd} A \arrow[r, "f"] \arrow[d, "f'"'] & B \arrow[d, "g"] \\ C \arrow[r, "g'"'] & D \end{tikzcd}$$

- spans

$$A \xleftarrow{g} C \xrightarrow{h} B$$

- cospans

$$A \xrightarrow{g} C \xleftarrow{h} B$$

Première construction universelle : produit

Le produit de A et B deux objets d'une catégorie $\mathcal{A}$ est la donnée d'un span

$$A \xleftarrow{p_A} P \xrightarrow{p_B} B$$

dans $\mathcal{A}$ vérifiant que pour tout autre span de la forme $A \xleftarrow{f_A} X \xrightarrow{f_B} B$ dans $\mathcal{A}$, il existe un unique morphisme $f : X \to P$ assurant que le diagramme ci-dessous commute :

$$\begin{tikzcd}
 & X \arrow[dl, "f_A"'] \arrow[d, dotted, "\exists! f"] \arrow[dr, "f_B"] & \\
A & P \arrow[l, "p_A"] \arrow[r, "p_B"'] & B
\end{tikzcd}$$

Les morphismes p_A and p_B sont appelés projections.

Produit : commentaires

- L'existence et l'unicité d'un morphisme $f\colon X \to P$, est noté dans les diagrammes par :

$$A \overset{\exists! f}{\dashrightarrow} B$$

- Si A et B admettent deux produits $A \xleftarrow{p_A} P \xrightarrow{p_B} B$ et $A \xleftarrow{p'_A} P' \xrightarrow{p'_B} B$, alors P et P' sont isormorphes.

- Par abus, le produit de A et B est décrit par P, dénoté par $A \times B$, laissant implicites les projections p_A et p_B.

Exemple dans $\mathbf{Set}$

Le produit de deux ensembles A et B dans $\mathbf{Set}$ est le produit cartésien $A \times B$ usuel muni des projections sur chacune des composantes.

Produits fibrés (pullbacks)

Le produit fibré d'un cospan $A \xrightarrow{g} C \xleftarrow{h} B$ est un span $A \xleftarrow{p_A} P \xrightarrow{p_B} B$ vérifiant que le carré

$$\begin{tikzcd} P \arrow[r, "p_A"] \arrow[d, "p_B"'] & A \arrow[d, "g"] \\ B \arrow[r, "h"'] & C \end{tikzcd} \tag{1}$$

est commutatif et que pour tout span $A \xleftarrow{f_A} X \xrightarrow{f_B} B$ formant un carré communtatif avec $A \xrightarrow{g} C \xleftarrow{h} B$, on a :

$$\begin{tikzcd} X \arrow[drr, bend left, "f_A"] \arrow[ddr, bend right, "f_B"'] \arrow[dr, dotted, "\exists! f"] & & \\ & P \arrow[r, "p_A"] \arrow[d, "p_B"'] & A \arrow[d, "g"] \\ & B \arrow[r, "h"'] & C \end{tikzcd}$$

Exemple

Le produit fibré de $A \xrightarrow{g} C \xleftarrow{h} B$ dans $\mathbf{Set}$ est l'ensemble

$$P = \{(a, b) \in A \times B \mid g(a) = h(b)\}$$

muni des projections

$$p_A(a, b) = a \ \text{ et } p_B(a, b) = b$$

Somme amalgamée (dual du produit fibré)

La somme amalgamée (pushout) d'un span $A \xleftarrow{g} C \xrightarrow{h} B$ est le cospan $A \xrightarrow{p_A} P \xleftarrow{p_B} B$ vérifiant que le carré

$$\begin{tikzcd} C \arrow[r, "g"] \arrow[d, "h"'] & A \arrow[d, "p_A"] \\ B \arrow[r, "p_B"'] & P \end{tikzcd}$$

commute et est universel : pour tout autre carré commutatif, il existe un unique morphisme assurant la commutatitivité de :

$$\begin{tikzcd} C \arrow[r, "g"] \arrow[d, "h"'] & A \arrow[d, "p_A"] \arrow[ddr, bend left, "f_A"] & \\ B \arrow[r, "p_B"'] \arrow[drr, bend right, "f_B"'] & P \arrow[dr, dotted, "\exists! f"] & \\ & & X \end{tikzcd}$$

p_A et p_B les coprojections de la somme amalgamée.

Somme amalgamée dans $\mathbf{Set}$

La somme amalgamée $A \xleftarrow{g} C \xrightarrow{h} B$ dans $\mathbf{Set}$ est

$$P = (A \sqcup B)/\sim$$

- $A \sqcup B$ est l'union disjointe de A et B, isomorphe à l'ensemble $(\{0\} \times A) \cup (\{1\} \times B)$
- et $\sim$ est la relation d'équivalence définie sur C par :
  $g(c) \sim h(c)$ pour tout $c \in C$

Les coprojections $p_A \colon A \to P$ et $p_B \colon B \to P$ envoient les éléments $a \in A$ et $b \in B$ sur leur classe d'équivalence dans P.

Abus de notation : $A +_C B$ pour $A \xleftarrow{g} C \xrightarrow{h} B$

Foncteurs

Un foncteur $F\colon \mathcal{A} \to \mathcal{B}$ entre deux catégories est défini par :

- une fonction $F : \mathcal{O}(\mathcal{A}) \to \mathcal{O}(\mathcal{B})$,
- et pour tout couple $A, A' \in \mathcal{A}$, une fonction $F : \mathcal{A}(A, A') \to \mathcal{B}(F(A), F(A'))$,

satisfaisant les propriétés suivantes :

- $F(g \circ f) = F(g) \circ F(f)$ pour tous les morphismes $f\colon A \to B$ et $g\colon B \to C$ dans $\mathcal{A}$;
- $F(1_A) = 1_{F(A)}$ pour tout objet $A \in \mathcal{A}$.

Catégorie $\mathbf{Cat}$

La catégorie des catégories $\mathbf{Cat}$ est définie par :

- objets = petites catégories.
- morphismes = foncteurs entre deux catégories.

Nota bene : $\mathbf{Cat}$ est bien définie car

- $1_{\mathcal{A}}$ est le foncteur dont les composantes sont les identités sur les objets et les morphismes de $\mathcal{A}$
- la composée de deux foncteurs se définit par la composée standard élément (objet et morphisme) par élément

Exemples de foncteur

- $P : \mathbf{Set} \to \mathbf{Set}$ défini par :
  - pour un ensemble A,
$$A \mapsto \mathcal{P}(A) = \{X \mid X \subseteq A\}$$
  - pour une application $f : A \to B$,
$$f \mapsto \quad P(f) : \begin{array}{ccc} \mathcal{P}(A) & \to & \mathcal{P}(B) \\ X & \mapsto & \{f(x) \mid x \in X\} \end{array}$$

  est un foncteur.

- Soient $(E, \leq)$ et $(E', \preccurlyeq)$ deux préordres.

  Les foncteurs $F : (E, \leq) \to (E', \preccurlyeq)$ sont les fonctions monotones, i.e. vérifiant $F(a) \preccurlyeq F(b)$ pour a et b de E avec $a \leq b$.

Catégorie slice $\mathcal{A}/A$ (ou catégorie des objets de $\mathcal{A}$ au dessus de A)

Soit A un objet d'une catégorie $\mathcal{A}$.

La catégorie slice de $\mathcal{A}$ sur A, notée $\mathcal{A}/A$, est définie par :

- les objets de $\mathcal{O}(\mathcal{A}/A)$ sont les morphismes $X \to A$ dans $\mathcal{A}$,
- pour deux objets $f\colon X \to A$ et $f'\colon X' \to A$ dans $\mathcal{A}/A$, l'ensemble $\mathcal{A}/A(f, f')$ des morphismes de f vers f' sont les morphismes $g\colon X \to X'$ assurant le diagramme commutatif suivant :

$$\begin{tikzcd} X \arrow[rr, "g"] \arrow[dr, "f"'] & & X' \arrow[dl, "f'"] \\ & A & \end{tikzcd}$$

Remarque : le morphisme 1_A est terminal dans $\mathcal{A}/A$.

Catégories pour les structures de données

- **Syntaxe**
  Une signature décrit l'interface d'un type abstrait à l'aide des identificateurs des types de données et de leurs opérations.

  Les signatures (interfaces des types abstraits) peuvent être organisées au sein d'une catégorie.

- **Sémantique**
  Pour une signature donnée, une algèbre fournit un ensemble de valeurs pour chacun des types et des interprétations fonctionnelles des opérations.

  Les algèbres définies pour une signature constituent les objets d'une catégorie.

Signatures

Une signature $\Omega = (S, F)$ d'un type de données est définie par

- un ensemble S de noms de types
- et un ensemble F de noms d'opérations f munis d'un profil $(s_1, \ldots s_m, s$ dans $S^* \times S$ noté sous la forme

$$f : s_1 \times \ldots \times s_m \to s$$

Exemple :

$$\begin{array}{rrrl} \Omega_1 = (& \{bool, int\}, & & \\ & \{true, false : & \to & bool, \\ & zero : & \to & int, \\ & succ : & int \to & int, \\ & plus : & int \times int \to & int, \\ & even : & int \to & bool, \quad \ldots\}) \end{array}$$

Catégorie des signatures

Un morphisme de signatures $\sigma : \Omega_1 = (S_1, F_1) \to \Omega_2 = (S_2, F_2)$ est défini par

- une application $\sigma_S : S_1 \to S_2$
- une application $\sigma_F : F_1 \to F_2$ préservant les profils, i.e. vérifiant que pour f dans F_1 de profil $s_1 \times \ldots \times s_m \to s$, $\sigma_F(f)$ est de profil $\sigma_S(s_1) \times \ldots \times \sigma_S(s_m) \to \sigma_S(s)$

$$
\begin{array}{rrcll}
\Omega_1 = (& \{bool, int\}, & & & \\
 & \{true, false : & & \rightarrow & bool, \\
 & zero : & & \rightarrow & int, \\
 & succ : & int & \rightarrow & int, \\
 & plus : & int \times int & \rightarrow & int, \\
 & even : & int & \rightarrow & bool, \quad \ldots\})
\end{array}
$$

$$
\begin{array}{rrcll}
\Omega_2 = (& \{Truth, Nat, List\}, & & & \\
 & \{\top, \bot : & & \rightarrow & Truth, \\
 & 0 : & & \rightarrow & Nat, \\
 & +1 : & Nat & \rightarrow & Nat, \\
 & + : & Nat \times Nat & \rightarrow & Nat, \\
 & pair : & Nat & \rightarrow & Truth, \\
 & [] : & & \rightarrow & List, \qquad \ldots\})
\end{array}
$$

$\sigma = (\sigma_S, \sigma_F) : \Omega_1 \rightarrow \Omega_2$ est un morphisme de signatures avec :

$$
\begin{array}{rl}
\sigma_S : & bool \mapsto Truth, int \mapsto Nat \\
\sigma_F : & true \mapsto \top, false \mapsto \bot, zero \mapsto 0, \\
 & succ \mapsto +1, plus \mapsto +, even \mapsto +
\end{array}
$$

Catégorie des Ω-algèbres $\mathbf{Alg}(\Omega)$

Étant donnée une signature Ω, une Ω-algèbre est définie par

- une famille d'ensembles de valeurs $(\mathcal{A}_s)_{s \in S}$
- et par une fonction $f^{\mathcal{A}} \colon \mathcal{A}_{s_1} \times \ldots \times \mathcal{A}_{s_m} \to \mathcal{A}_s$ pour tout nom d'opération $f \colon s_1 \times \ldots \times s_m \to s$ de Ω.

Un morphisme de Ω-algèbres $g \colon \mathcal{A} \to \mathcal{B}$ est une famille d'applications $(g_s \colon \mathcal{A}_s \to \mathcal{B}_s)_{s \in S}$ vérifiant que pour tout $f \colon s_1 \times \ldots \times s_m \to s \in F$ et pour $a_1 \in \mathcal{A}_{s_1}$, …, $a_m \in \mathcal{A}_{s_m}$,

$$g_s(f^{\mathcal{A}}(a_1, \ldots, a_m)) = f^{\mathcal{B}}(g_{s_1}(a_1), \ldots, g_{s_m}(a_m)).$$

Les Ω-algèbres et leurs morphismes forment la catégorie $\mathbf{Alg}(\Omega)$.

Ω-termes

Soit $\Omega = (S, F)$ une signature et X une famille S-indexée d'ensembles de variables, disjointe de F.
Pour $s \in S$, l'ensemble $T_\Omega(X)_s$ des termes de type s sur Ω avec variables dans X est défini inductivement par :

- pour x dans X_s,
$$x \in T_\Omega(X)_s$$
- pour $f\colon s_1 \times \ldots \times s_m \to s$ dans F, pour tout $t_1 \in T_\Omega(X)_{s_1}$, …, $t_m \in T_\Omega(X)_{s_m}$,
$$f(t_1, \ldots, t_m) \in T_\Omega(X)_s$$

L'ensemble $T_\Omega(X) = \bigsqcup_{s \in S} T_\Omega(X)_s$ est l'ensemble des Ω-termes avec variables dans X.

Algèbre des termes

Les termes définissent l'algèbre des termes notée $\mathcal{T}_\Omega(X)$ où :

- l'ensemble de valeurs $\mathcal{T}_\Omega(X)_s$ de type s est $T_\Omega(X)_s$
- pour $f\colon s_1 \times \ldots \times s_m \to s$ dans F, pour tous $t_1 \in T_\Omega(X)_{s_1}$, …, $t_m \in T_\Omega(X)_{s_m}$,
  $f^{\mathcal{T}_\Omega(X)}(t_1, \ldots, t_m)$ est le terme $f(t_1, \ldots, t_m)$ de $T_\Omega(X)_s$.

Pour une Ω-algèbre $\mathcal{A}$, toute fonction $f\colon X \to \mathcal{A}$ définit un morphisme d'algèbres unique

$$f^*\colon \mathcal{T}_\Omega(X) \to \mathcal{A}$$

exprimant l'évaluation des termes dans $\mathcal{A}$.

$\mathcal{T}_\Omega(X)$ est initiale dans $\mathbf{Alg}(\Omega)$.

Algèbre terminale

$1_{\mathbf{Alg}(\Omega)}$ est l'algèbre définie par

- un singleton comme ensemble de valeurs pour tout type s
- pour $f\colon s_1 \times \ldots \times s_m \to s$, l'unique fonction d'interprétation définie sur ces valeurs.

$1_{\mathbf{Alg}(\Omega)}$ est terminale dans $\mathbf{Alg}(\Omega)$.

Algèbres à observations près

Considérons l'algèbre $\mathcal{A}_{true \neq false}$ qui interprète les constantes $true$ et $false$ avec deux valeurs distinctes et avec une seule valeur pour tous les autres types que $bool$.

$$\begin{tikzcd}
\mathcal{A} \arrow[rr, "g"] \arrow[dr, "f"'] & & \mathcal{B} \arrow[dl, "f'"] \\
& \mathcal{A}_{true \neq false} &
\end{tikzcd}$$

La catégorie $\mathbf{Alg}(\Omega)/\mathcal{A}_{true \neq false}$ caractérise les algèbres qui distinguent les 2 valeurs de vérité.

Foncteur d'oubli

Soit $\sigma : \Omega_1 = (S_1, F_1) \to \omega_2 = (S_2, F_2)$ un morphisme de signature.

$U_\sigma : \mathbf{Alg}(\Omega_2) \to \mathbf{Alg}(\Omega_1)$ défini par

- pour $\mathcal{B}$ Ω_2-algèbre, $U_\sigma(\mathcal{B})$ est la Ω_1-algèbre définie par :
  - $A_s = B_{\sigma_S(s)}$ pour $s \in S_1$
  - $f^{\mathcal{A}} = \sigma_F(f)^{\mathcal{B}}$ pour $f : s_1 \times s_m \to s$ dans F_1
- pour $g : \mathcal{B} \to \mathcal{B}'$ un morphisme dans $\mathbf{Alg}(\Omega_2)$, $U_\sigma(g) : U_\sigma(\mathcal{B}) \to U_\sigma(\mathcal{B}')$ est défini par :

$$U_\sigma(g)_s(a) = g_s(a) \text{ pour } a \in B_{\sigma_S(s)}$$

est le foncteur d'oubli selon le morphisme de signature σ

Dans le cadre de la modélisation géométrique, ...

Nous considérerons :

- la signature $\Omega = (\{bool, int\}, \{true, false :\rightarrow bool, 0, 1, \ldots :\rightarrow int, + : int \times int \rightarrow int\})$ contenant les types de données des booléens et des entiers, munis des constantes usuelles et de l'addition de deux entiers
- l'algèbre $\mathcal{A}$ définie par $\mathcal{A}_{bool} = \{true, false\}$ et $\mathcal{A}_{int} = \mathbb{N}$ et l'interprétation de l'opération $+$ par l'addition dans $\mathbb{N}$.

Quelques mots de conclusion sur les catégories

Contraste entre :

- un excès de formalité pour des notions simples (identité/réflexivité, associativité/transitivité)
- des notations et des concepts clés pour caractériser et étudier différentes classes d'objets et raisonner à leur propos

Quelques mots de conclusion sur les catégories

Contraste entre :

- un excès de formalité pour des notions simples (identité/réflexivité, associativité/transitivité)
- des notations et des concepts clés pour caractériser et étudier différentes classes d'objets et raisonner à leur propos

Intérêt pour

- Abstraire et analyser
- Mettre en évidence de nouvelles connexions

Graphes comme une catégorie ?

Bien sûr,

Graphes comme une catégorie ?

Bien sûr,

mais autant de catégories de graphes qu'il y a de classes de graphes :

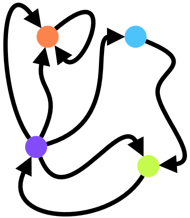
- nœuds et arcs étiquetés ou non,
- contraintes sur les informations attachées aux nœuds ou arcs.
- calculs sur les informations ou pas

Multi-graphes

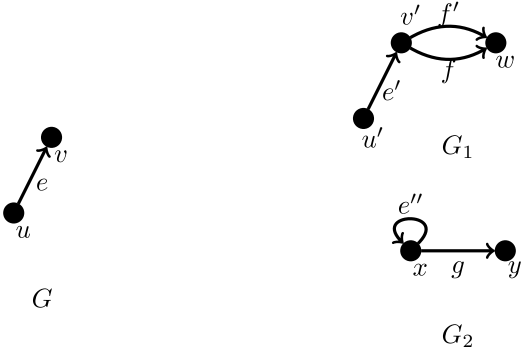
Graphe (Multigraphe orienté avec boucles)

Un graphe $G = (V, E, s, t)$ est défini par :

- ensemble de nœuds V,
- ensemble d'arcs E,
- fonction source $s : E \to V$,
- fonction cible $t : E \to V$,



Exemples de graphes



Catégorie $\mathbf{Graph}$

Soit $G = (V_G, E_G, s_G, t_G)$ et $H = (V_H, E_H, s_H, t_H)$ 2 graphes.
Un morphisme de graphes $m = (m_V, m_E) \colon G \to H$ est défini par deux fonctions

- $m_V \colon V_G \to V_H$
- et $m_E \colon E_G \to E_H$

préservant les sources et cibles des arcs, i.e., assurant les diagrammes commutatifs suivants :

$$\begin{tikzcd} E_G \arrow[r, "s_G"] \arrow[d, "m_E"'] & V_G \arrow[d, "m_V"] \\ E_H \arrow[r, "s_H"'] & V_H \end{tikzcd} \qquad \begin{tikzcd} E_G \arrow[r, "t_G"] \arrow[d, "m_E"'] & V_G \arrow[d, "m_V"] \\ E_H \arrow[r, "t_H"'] & V_H \end{tikzcd} \tag{2}$$

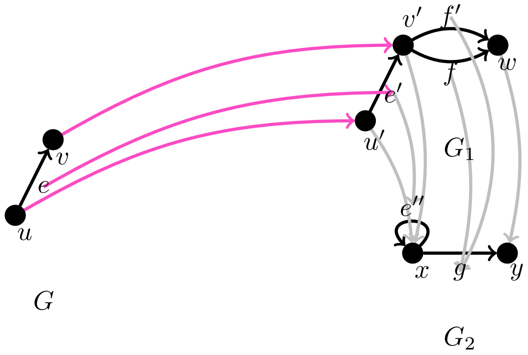
Graphes particuliers de $\mathbf{Graph}$

Dans $\mathbf{Graph}$:

- le graphe vide $G_\emptyset = (\emptyset, \emptyset, 1_\emptyset, 1_\emptyset)$ est initial
- le graphe réduit à un seul nœud avec une boucle $(\{v\}, \{e\}, s\colon e \mapsto v, t\colon e \mapsto v)$ est terminal.

Un morphisme $m = (m_V, m_E)\colon G \to H$ dans $\mathbf{Graph}$ est un mono (resp. un épi) si et seulement si m_V and m_E sont des fonctions injectives (resp. surjectives).

Exemples de morphismes de graphes



$G \longrightarrow G_1$

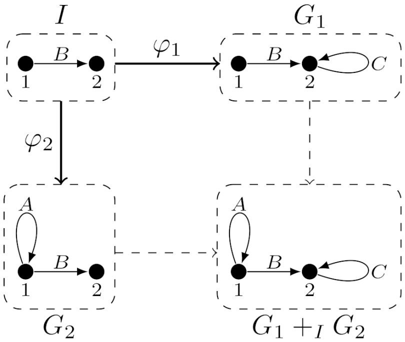
$G_1 \longrightarrow G_2$

Héritage des constructions catégoriques des ensembles

Les multi-graphes sont par essence un couple d'ensembles (nœuds et arcs) reliés par les fonctions source et cible.

Les constructions catégoriques des ensembles (produit, produit fibré/pullback, somme amalgamée/pushout) se transposent pour les multi-graphes.

Somme amalgamée (pushout) pour les graphes

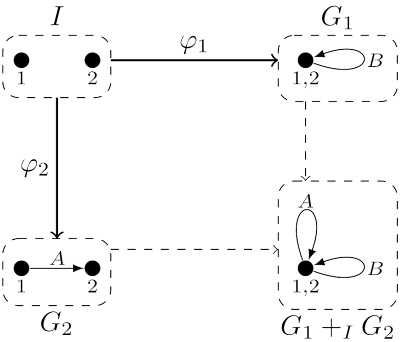


$$G_1 \xleftarrow{\varphi_1} I \xrightarrow{\varphi_2} G_2\text{, i.e. } G_1 +_I G_2$$

$$\approx$$

Union disjointe de G_1 et G_2 quotientée par I

Somme amalgamée (pushout) pour les graphes



$$G_1 \xleftarrow{\varphi_1} I \xrightarrow{\varphi_2} G_2, \text{ i.e. } G_1 +_I G_2$$

$$\approx$$

Union disjointe de G_1 et G_2 quotientée par I

Graphes étiquetés

Graphe étiqueté ou coloré

- décorations = ajout d'information sur les éléments (nœuds ou arcs) des graphes
- information = élément d'un ensemble fini (ou alphabet)

Ajout de fonctions d'étiquetage :

- $l_E : E \to \Sigma_E$ pour les arcs
- $l_V : V \to \Sigma_V$ pour les nœuds

$$\Sigma_E \xleftarrow{le} E \overset{s}{\underset{t}{\rightrightarrows}} V \xrightarrow{lv} \Sigma_V$$

Catégorie des graphes étiquetés Σ-$\mathbf{Graph}$

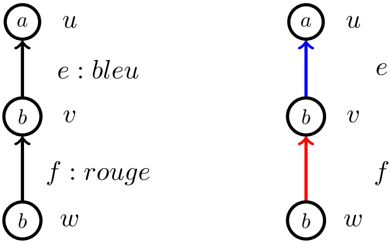
Soit $\Sigma = (\Sigma_V, \Sigma_E)$ un couple d'alphabets.
Un graphe étiqueté $G = (V_G, E_G, s_G, t_G, lv_G, le_G)$ sur Σ est défini par :

- un graphe (V_G, E_G, s_G, t_G)
- une fonction d'étiquetage des nœuds $lv_G \colon V_G \to \Sigma_V$
- une fonction d'étiquetage des arcs $le_G \colon E_G \to \Sigma_E$.

Les morphismes des graphes étiquetés $m = (m_V, m_E)$ assurent la commutativité des diagrammes dans $\mathbf{Set}$:

$$\begin{tikzcd} V_G \arrow[rr, "m_V"] \arrow[dr, "lv_G"'] & & V_H \arrow[dl, "lv_H"] \\ & \Sigma_V & \end{tikzcd} \qquad \begin{tikzcd} E_G \arrow[rr, "m_E"] \arrow[dr, "le_G"'] & & E_H \arrow[dl, "le_H"] \\ & \Sigma_E & \end{tikzcd}$$

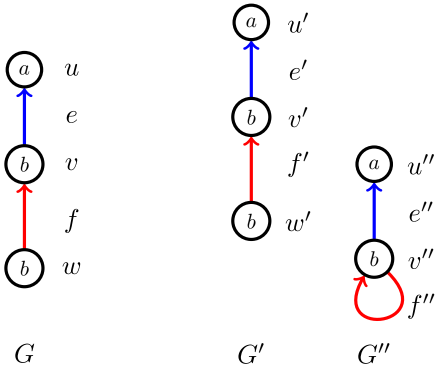
Exemple d'un graphe étiqueté



Deux visualisations d'un même graphe étiqueté :

- les nœuds sont étiquetés par une lettre de l'alphabet $\mathbb{A} = \{a, b, \ldots, z\}$
- et les arcs sont étiquetés par une couleur de l'ensemble $\mathbb{C} = \{bleu, rouge, noir, vert, \ldots\}$.

Exemple de morphismes de graphes étiquetés



- Morphisme de G vers G' défini en associant à chaque élément de G, nœud ou arc, sa version primée dans G'.
- Morphisme de G vers G'' défini par
  - $u \mapsto u'', v \mapsto v'', w \mapsto v''$ pour les nœuds
  - $e \mapsto e'', f \mapsto f''$ pour les arcs.

Structuration des informations

- Jusqu'à présent, la décoration des graphes est laissée complètement libre.
- *Comment exprimer des agencement entre les décorations ?*

  Par exemple, comment imposer qu'un arc rouge relie nécessairement des nœuds étiquetés par b ?

Un graphe type TG spécifie les relations entre les décorations.
Typer un graphe G selon TG :

- les nœuds de TG représenteront les types de nœuds de G
- et les arcs de TG représenteront les types des arcs de G.

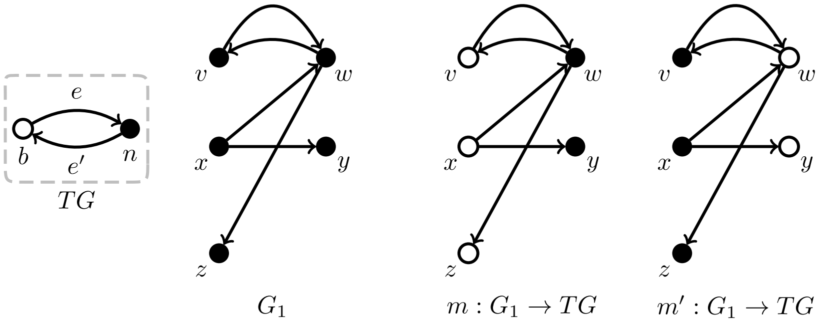
Catégorie des graphes typés $\mathbf{Graph}/\,\mathrm{TG}$

La catégorie des graphes typés sur un graphe type TG $\mathbf{Graph}_{\mathrm{TG}}$ est la catégorie slice $\mathbf{Graph}/\mathrm{TG}$.

Attention :

- Rigoureusement, un graphe typé n'est plus un graphe, mais un morphisme de graphes $G \to \mathrm{TG}$
- Il peut exister plusieurs morphismes $G \to \mathrm{TG}$, chacun de ces morphismes décrivant un graphe typé distinct.

Graphes bipartis comme des graphes typés

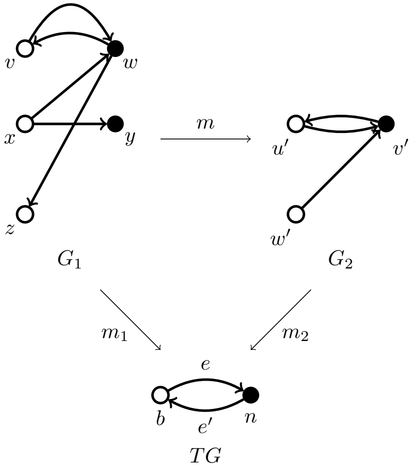


Convention graphique :

- les éléments (nœuds et arcs) du graphe cible G héritent pour décoration les éléments du graphe source TG
- les graphes typés sont représentés comme des graphes étiquetés, en inscrivant les types sur les éléments de G.

Morphismes de graphes typés

Les morphismes des graphes typés sont des triangles commutatifs avec le graphe type TG comme cible.



$m\colon G_1 \to G_2$

- pour les nœuds, $v \mapsto u'$, $w \mapsto v'$, $x \mapsto w'$, $y \mapsto v'$, et $z \mapsto u'$,
- chaque arc e de G_1 est envoyé sur l'arc de G_2 partageant les nœuds source et cible.

Propriétés de $\mathbf{Graph}_{\mathbf{TG}(\mathbf{\Sigma})}$

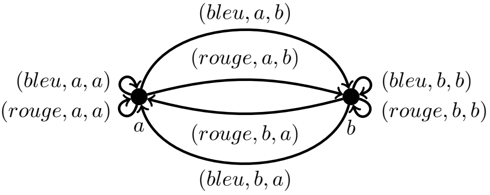
- $(\mathrm{TG}, \mathbf{1}_{\mathrm{TG}})$ est terminal dans $\mathbf{Graph}_{\mathrm{TG}}$,
- $(G_\emptyset, G_\emptyset \to \mathrm{TG})$ est initial dans $\mathbf{Graph}_{\mathrm{TG}}$,
- les monomorphismes (resp. épimorphismes) des graphes typés sont les fonctions injectives (resp. surjectives)
- et l'existence des produits, produits fibrés, et sommes amalgamées est assurée.

Graphes étiquetés comme des graphes typés

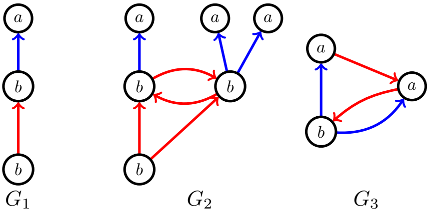
La catégorie des graphes étiquetés sur $\Sigma = (\Sigma_V, \Sigma_E)$ est équivalente à la catégorie $\mathbf{Graph}_{\mathbf{TG(\Sigma)}}$ des graphes typés sur le graphe $TG(\Sigma) = (V_\Sigma, E_\Sigma, s_\Sigma, t_\Sigma)$ défini par :

- $V_\Sigma = \Sigma_V$;
- $E_\Sigma = \Sigma_E \times \Sigma_V \times \Sigma_V$;
- $s_\Sigma \colon E_\Sigma \to V_\Sigma;\ (e, a, b) \mapsto a$;
- $t_\Sigma \colon E_\Sigma \to V_\Sigma;\ (e, a, b) \mapsto b$.

Graphes étiquetés comme des graphes typés



$TG(A, C)$



$A = \{a, b\}$
$C = \{bleu, rouge\}$

Informations pour calculer

Graphes étiquetés ou typés $\Rightarrow$ les décorations sont des éléments *immuables*.

Décorations comme des termes (exprimant des valeurs et des calculs) $\Rightarrow$ on parle alors d'attributs

Par simplicité, nous définissons les graphes attribués pour les seuls nœuds, sans attribut pour les arcs.

E-graphes : extension des graphes avec des données

$$E \underset{t}{\overset{s}{\rightrightarrows}} V \xleftarrow{sa} A \xrightarrow{ta} D \tag{3}$$

Un E-graphe $G = (V_G, D_G, E_G, A_G, s_G, t_G, sa_G, ta_G)$ est défini par

- un graphe (V_G, E_G, s_G, t_G),
- un ensemble de données D_G,
- un ensemble d'arcs d'attribution A_G
- et deux fonctions $sa_G \colon A_G \to V_G$ et $ta_G \colon A_G \to D_G$ pour les nœuds sources et cibles des arcs d'attribution.

Catégorie des E-graphes

Un morphisme d'E-graphes $m\colon G \to H$ est défini par 4 fonctions :

- $m_V\colon V_G \to V_H$,
- $m_D\colon D_G \to D_H$,
- $m_E\colon E_G \to E_H$,
- et $m_A\colon A_G \to A_H$

qui induisent des diagrammes commutatifs dans $\mathbf{Set}$ pour les fonctions sources et cibles des arcs et des arcs d'attribution.

Les E-graphes et leurs morphismes forment une catégorie notée $\mathbf{EGraph}$.

Graphes attribués comme des E-graphes définis sur une Ω-algèbre

Graphes attribués

Un graphe Ω-attribué $AG = (G, \mathcal{A})$ est défini par

- un E-graphe $G = (V_G, D_G, E_G, A_G, s_G, t_G, sa_G, ta_G)$
- et une Ω-algèbre $\mathcal{A}$ vérifiant $D_G = \bigsqcup_{s \in S} \mathcal{A}_s$.

Catégorie des graphes attribués

Un morphisme de graphes Ω-attribués $m\colon (G, \mathcal{A}) \to (H, \mathcal{B})$ est un couple $(m_G, m_\mathcal{A})$ avec

- $m_G\colon G \to H$ un morphisme de E-graphes
- et $m_\mathcal{A}\colon \mathcal{A} \to \mathcal{B}$ un morphisme d'algèbres

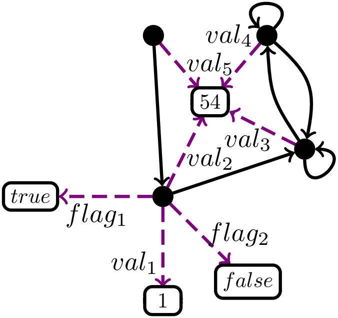
assurant que le diagramme ci-dessous commute pour tous les types s de S,

$$\begin{tikzcd}
\mathcal{A}_s \arrow[r, "(m_\mathcal{A})_s"] \arrow[d, hook] & \mathcal{B}_s \arrow[d, hook] \\
D_G \arrow[r, "(m_G)_D"'] & D_H
\end{tikzcd}$$

où les flèches verticales représentent des inclusions dans $\mathbf{Set}$.

Les graphes Ω-attribués et leurs morphismes forment la catégorie Ω-$\mathbf{AGraph}$.

Un graphe attribué



Arcs d'attribution :

$$\{val_i \mid i \in 1..5\} \cup \{flag_j \mid j \in 1..2\}$$

Graphes typés pour spécifier les arcs d'attribution

Graphes attribués : n'importe quelle donnée peut être associée à n'importe quel nœud.

Graphes attribués typés :

- catégorisation des attributs selon les types de données
- utilisation de l'algèbre terminale $1_{\mathbf{Alg}(\Omega)}$ pour typer les attributs

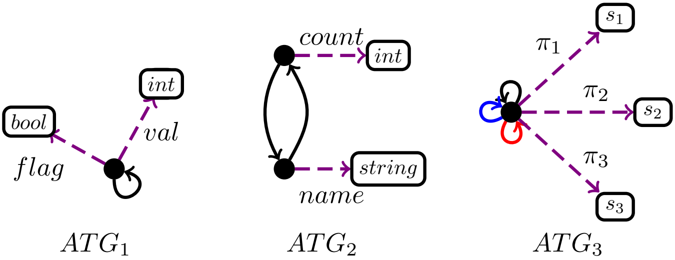
Graphes attribués typés

Un graphe type Ω-attribué est un graphe Ω-attribué $\mathrm{ATG} = (\mathrm{TG}, \mathbf{1}_{\mathbf{Alg}(\Omega)})$.

La catégorie Ω-$\mathbf{AGraph}_{\mathbf{ATG}}$ des graphes attribués typés sur ATG est la catégorie slice Ω-$\mathbf{AGraph}/\mathrm{ATG}$.
Chaque arc d'un graphe type Ω-attribué $\mathrm{ATG} = (\mathrm{TG}, \mathbf{1}_{\mathbf{Alg}(\Omega)})$ décrit les attributs associés aux nœuds

- par un identificateur, l'arc lui-même,
- et par un type de donnée, la valeur dans $\mathbf{1}_{\mathbf{Alg}(\Omega)}$ cible de l'arc.

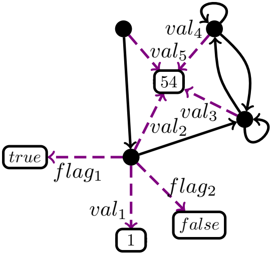
Exemples de graphes types



Quelques commentaires

- Un arc d'attribution d'un graphe type décrit un attribut par
  - un nom, le nom de l'arc lui-même,
  - un ensemble de valeurs, toutes les valeurs de type la cible de l'arc d'attribution.

- Pas d'exigence en terme d'unicité ou de multiciplicité des arcs d'attribution

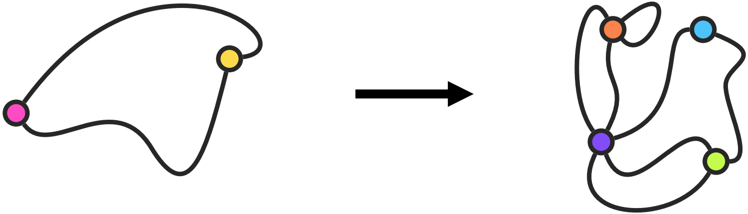


Catégories des graphes : que retenir ?

- $\mathbf{Graph}$ catégorie la plus basique, définie par un ensemble de nœuds et un ensemble d'arcs
- Décorations des graphes comme des étiquettes, des graphes types, des attributs ou leur combinaison
- Morphismes de graphes : *notion clé pour effectuer les transformations de graphes*

Comment réécrire les graphes ?

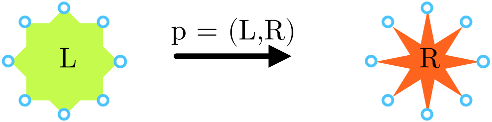
$$\bullet \overset{*}{—} \bullet \overset{o}{—} \bullet \overset{r}{—} \bullet \overset{a}{—} \bullet \overset{n}{—} \bullet \overset{t}{—} \bullet \overset{*}{—} \bullet \quad \rightarrow \quad \bullet \overset{*}{—} \bullet \overset{e}{—} \bullet \overset{u}{—} \bullet \overset{r}{—} \bullet \overset{*}{—} \bullet$$



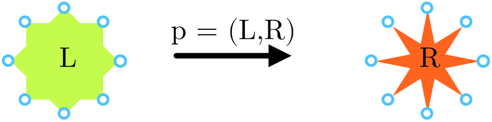
Pas de notion de début et de fin dans un graphe.

$\Rightarrow$ Identification d'éléments de jonction à l'aide de morphismes de graphes.

Règles de transformations de graphes

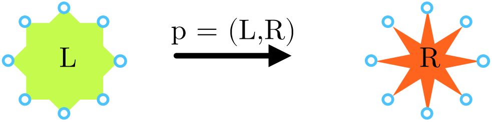


Règles de transformations de graphes

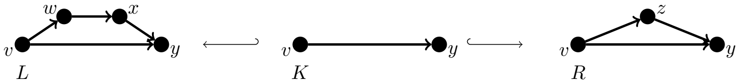


Nœuds communs aux membres gauches et droits sont évidés

Règles de transformations de graphes



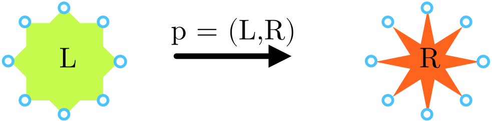
Nœuds communs aux membres gauches et droits sont évidés



Nœuds (et arcs) communs sont placés dans l'interface

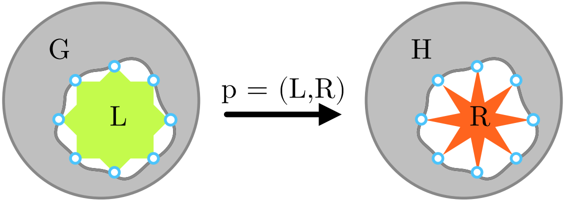
Règles de transformations de graphes

Comment généraliser la réécriture des chaînes aux graphes



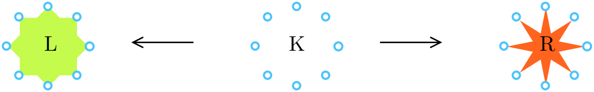
Règles de transformations de graphes

Comment généraliser la réécriture des chaînes aux graphes

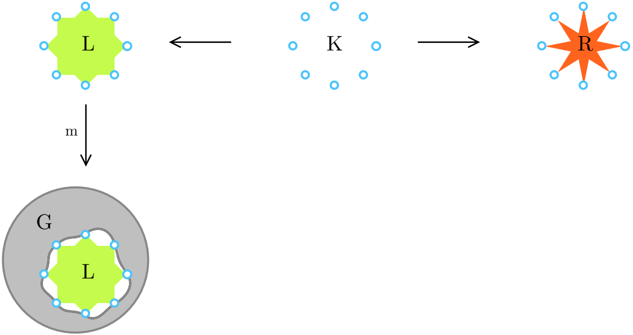


- Trouver dans G un graphe similaire à L,
- L'enlever de G,
- Reconnecter R avec les éléments restants de G en préservant les éléments de l'interface.

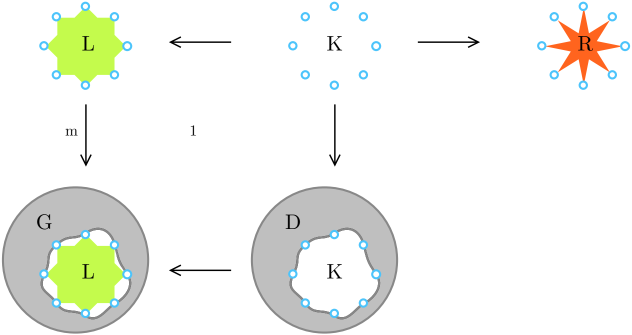
Application des règles de transformations de graphes



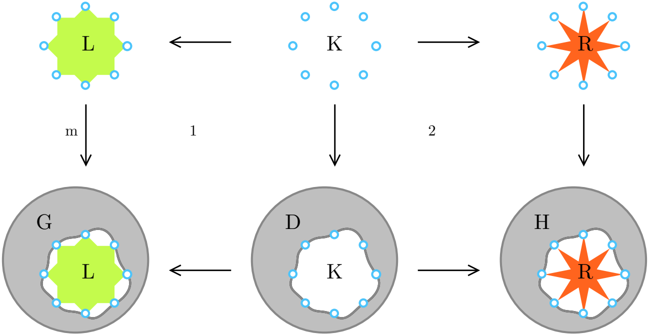
Application des règles de transformations de graphes



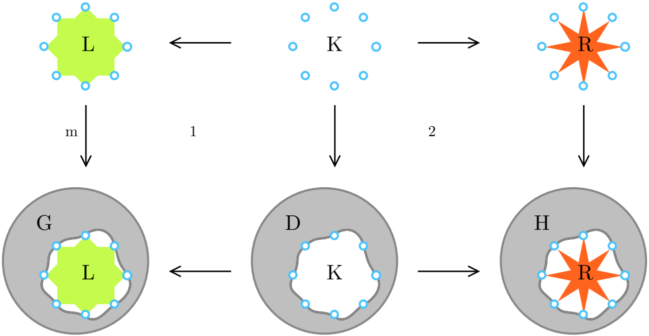
Application des règles de transformations de graphes



Application des règles de transformations de graphes



En termes catégoriques



1 et 2 sont deux carrés commutatifs, définis comme des pushout.

Y-a-t-il un souci ?

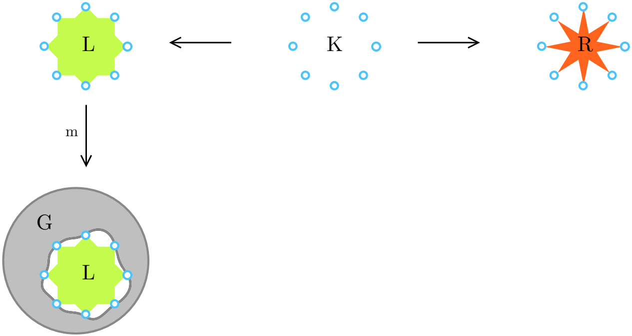
Y-a-t-il un souci ?

Le span $L \leftarrow K \rightarrow D$ n'est pas donné,

Y-a-t-il un souci ?

Le span $L \leftarrow K \rightarrow D$ n'est pas donné,

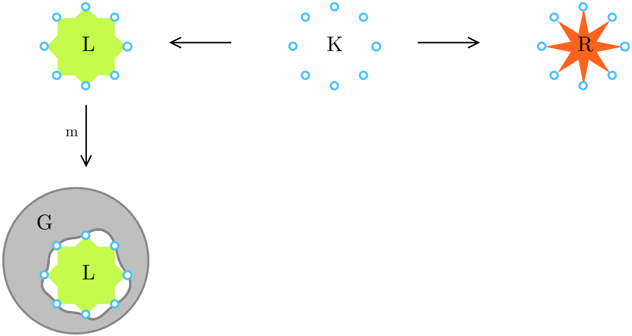
seulement le diagramme $K \rightarrow L \rightarrow G$.



Y-a-t-il un souci ?

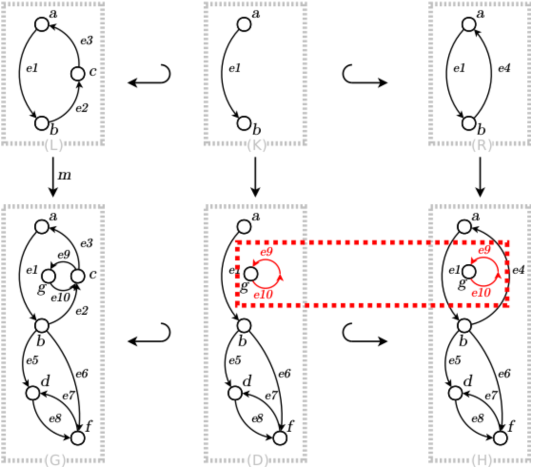
Le span $L \leftarrow K \rightarrow D$ n'est pas donné,

seulement le diagramme $K \rightarrow L \rightarrow G$.



D existe-t-il nécessairement ?

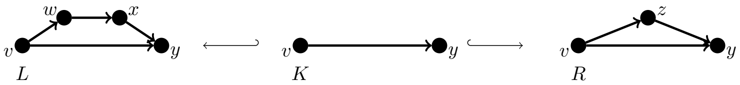
Contre-exemple



Réécriture dans $\mathbf{Graph}$

Une règle, ou production, p est un span $L \overset{l}{\hookleftarrow} K \overset{r}{\hookrightarrow} R$ de monomorphismes.

Les graphes L, K et R sont respectivement appelés *membre gauche*, *interface* et *membre droit* de la règle,



Match pour une règle

Un match pour une règle $p = L \overset{l}{\hookleftarrow} K \overset{r}{\hookrightarrow} R$ est un monomorphisme $m\colon L \hookrightarrow G$.

Un match satisfait la condition de recollement si $K \overset{l}{\hookrightarrow} L \overset{m}{\hookrightarrow} G$ admet un complément de somme amalgamée, i.e., s'il existe un objet D et une paire de morphismes $K \overset{k}{\to} D \overset{g}{\to} G$ tel que :

$$
\begin{tikzcd}
L \arrow[d, hook, "m"'] \arrow[dr, phantom, "PO"] & K \arrow[l, hook', "l"'] \arrow[d, "k"] \\
G & D \arrow[l, "g"]
\end{tikzcd}
$$

Condition d'arcs pendants

En se restreignant aux monomorphismes, la condition de recollement se ramème à la condition d'arcs pendants suivante :

Aucun nœud de $m(V_L) \setminus m(l(V_K))$ est
la source ou la cible d'un arc de $E_G \setminus m(E_L)$.

Dérivation directe

Une règle $p = L \overset{l}{\hookleftarrow} K \overset{r}{\hookrightarrow} R$ transforme G en H via un match $m \colon L \hookrightarrow G$ s'il existe un diagramme

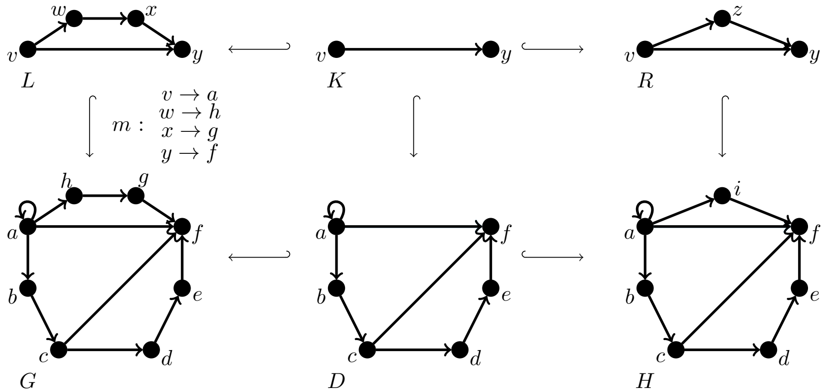
$$\begin{tikzcd}
L \arrow[d, hook, "m"'] \arrow[dr, phantom, "PO"] & K \arrow[l, hook', "l"'] \arrow[r, hook, "r"] \arrow[d] \arrow[dr, phantom, "PO"] & R \arrow[d, "m'"] \\
G & D \arrow[l] \arrow[r] & H
\end{tikzcd}$$

où les deux carrés sont des sommes amalgamées.

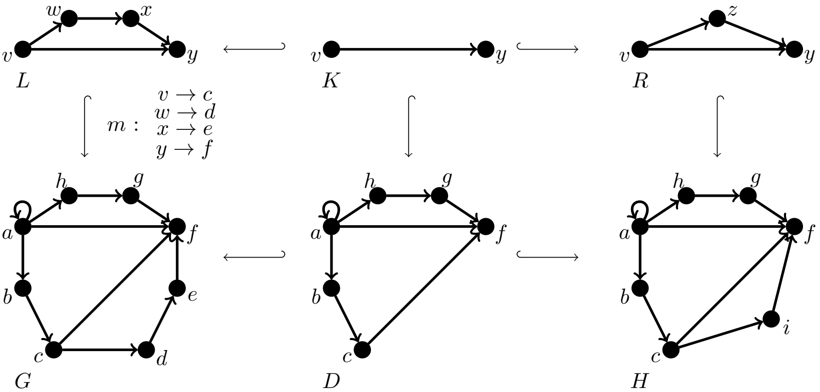
La dérivation directe $G \Rightarrow^{r,m} H$ existe et est unique (à isomorphisme près) si m satisfait la condition de recollement.

$m' \colon R \hookrightarrow H$ est appelé comatch de la dérivation.

Un premier exemple de dérivation directe



Un second exemple de dérivation directe



Commentaires

- Réécriture par double somme amalgamée bien fondée dans la catégorie $\mathbf{Graph}$
- *Bémol :* la construction du complément de la somme amalagamée est *ad hoc*.

La définition des catégories adhésives axiomatise l'existence des sommes amalgamées (et produits fibrés) à l'instar de $\mathbf{Set}$ afin d'assurer le bien-fondé de la réécriture.

Catégories adhésives

Définies à l'aide de la notion de *carré de Van Kampen*
Propriétés

- $\mathbf{Set}$ et $\mathbf{Graph}$ sont des catégories adhésives
- stabilité des monomorphismes par somme amalgamées :

$$\begin{tikzcd} C \arrow[r, hook] \arrow[d] & A \arrow[d] \\ B \arrow[r, hook] & D \end{tikzcd}$$

- unicité du complément de somme amalgamée (à isomorphisme près).
- préservation de l'adhésivité par slicing

Règles de graphes attribués

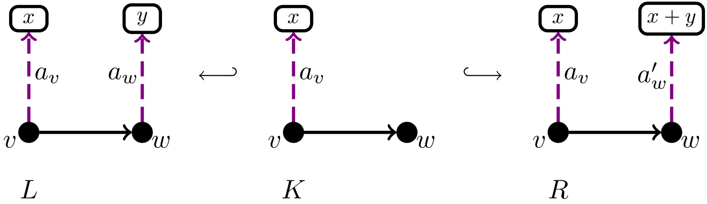
La catégorie des graphes attribués n'étant pas adhésive, elle est restreinte à la classe $\mathcal{M}$ de morphismes.

$\mathcal{M}$ est la classe des morphismes $m = (m_G, m_{\mathcal{A}})$ de Ω-$\mathbf{AGraph}$ où m_G est un monomorphisme et $m_{\mathcal{A}}$ est un isomorphisme constitue un système stable de monomorphismes.

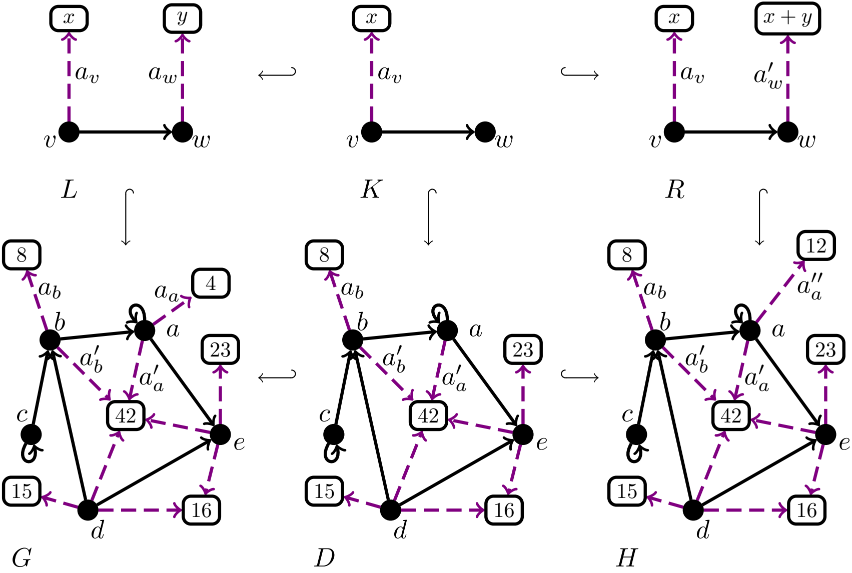
Une règle Ω-attribuée est un span $L \hookleftarrow K \hookrightarrow R$ de $\mathcal{M}$-morphismes dans Ω-$\mathbf{AGraph}$ où

- les graphes attribués partagent l'algèbre $\mathcal{T}_\Omega(X)$
- et les morphismes d'algèbres sont $1_{\mathcal{T}_\Omega(X)}$.

Une règle de graphes attribués



Une réécriture de graphes attribués



Match envoie l'arc $v \to w$ sur l'arc $b \to a$

En guise de conclusion partielle

Jusqu'à présent, panorama très partiel des transformations de graphes par l'approche catégorielle DPO

N'ont pas été abordé :

1. Approches alternatives à DPO
2. Des propriétés ou extensions telles que la confluence, l'application concurrente des règles, la préservation de propriétés, des conditions restrictives d'application
3. Les aspects outillages ou spécialisations en lien avec un domaine d'application

La session suivante s'intéresse au point 3 pour la modélisation géométrique.