Le rôle des limites projectives dans le développement des mémoires procédurale et sémantique

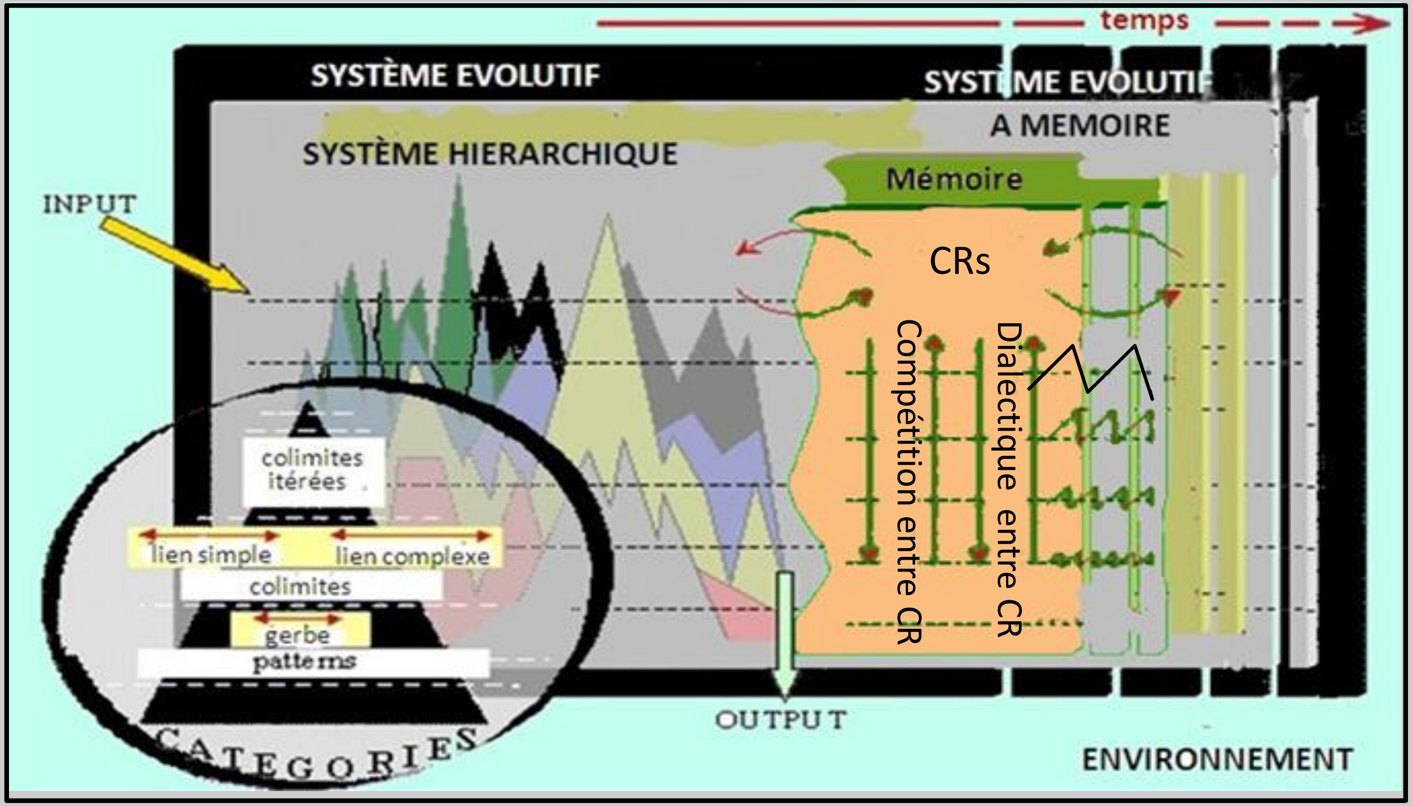
par

Andrée C. Ehresmann

Université de Picardie Jules Verne
ehres@u-picardie.fr
http://ehres.pagesperso-orange.fr
http://vbm-ehr.pagesperso-orange.fr

Séminaire Mamuphi ENS Paris, Janvier 2013

SYSTÈME EVOLUTIF A MÉMOIRE



Modèle 'catégorique' pour des systèmes complexes :

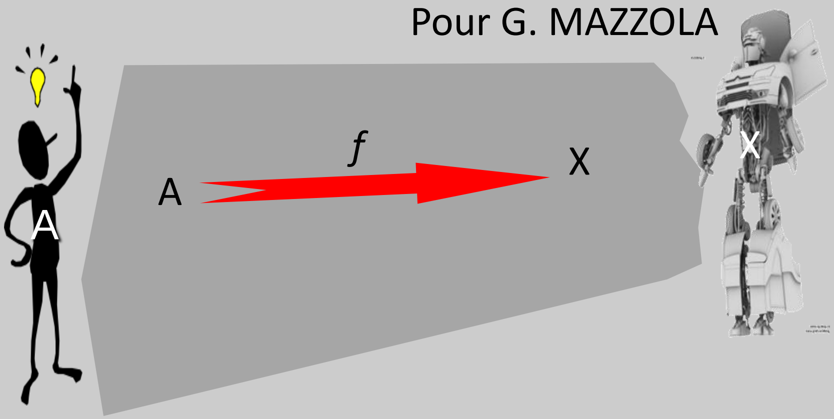
évolutifs,

avec une *hiérarchie* enchevêtrée de composants qui varient,

développant une *Mémoire* robuste mais flexible,

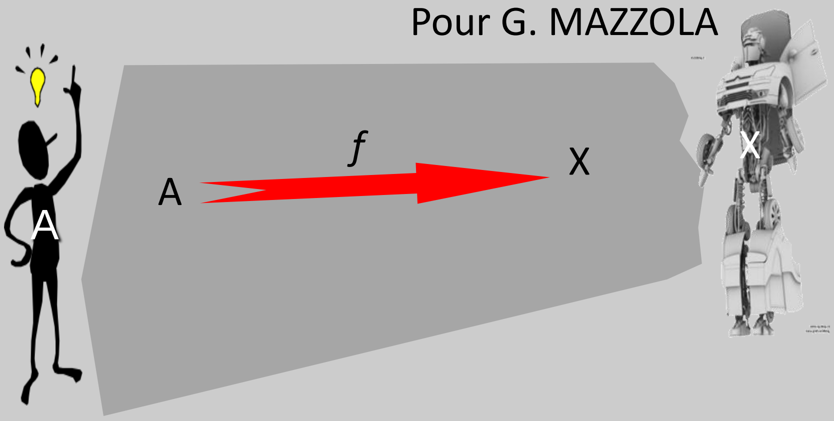
auto-organisés par un réseau de sous-systèmes fonctionnels, les *Co-Régulateurs,* avec des temporalités différentes.

SIGNIFICATION DES FLECHES



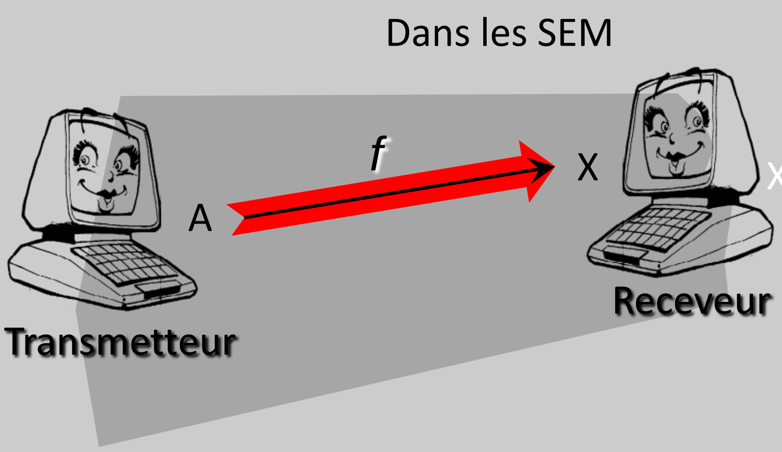
Mazzola interprète f: A → X comme une *perspective sur* X *d'adresse* A :
"Intuitivement, on se place sur A et on regarde X ". Cp. avec Aristote où la vision correspond à une force qui sort des yeux et vient sur l'objet.

SIGNIFICATION DES FLECHES

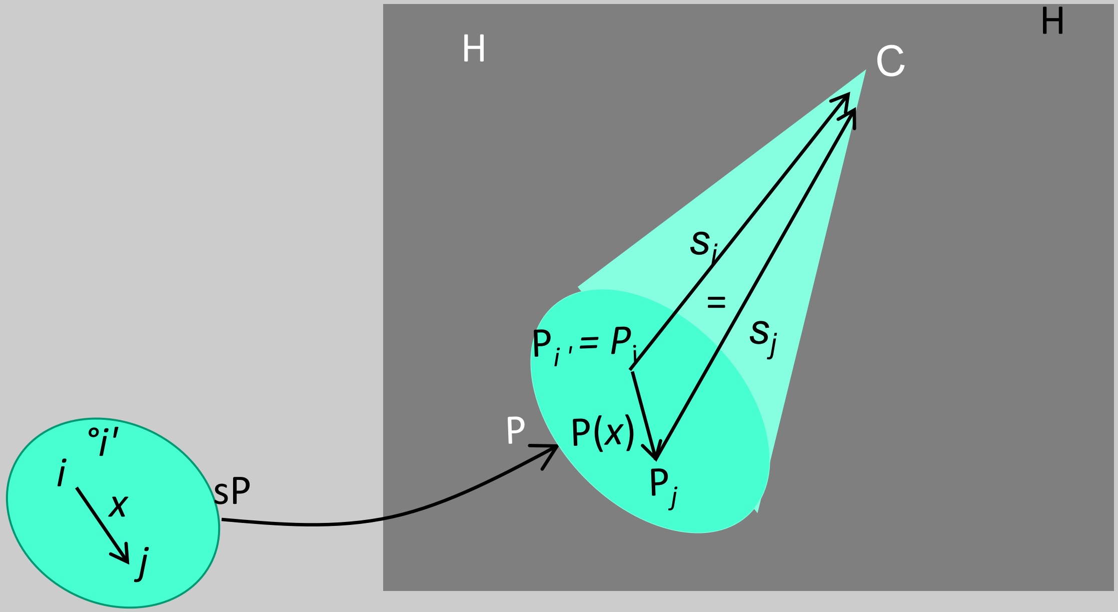


Mazzola interprète f: A → X comme une *perspective sur* X *d'adresse* A : "Intuitivement, on se place sur A et on regarde X ". Cp. avec Aristote où la vision correspond à une force qui sort des yeux et vient sur l'objet.

Dans les SEM, f est interprété comme un canal par où une information ou une commande peut être transmise de A vers X : A est le transmetteur, X le receveur, A agit, X reçoit.
La flèche f est souvent étiquetée par son *délai de propagation, sa force,* et un *index d'activité* signalant si elle est active ou non en *t.*



PATTERNS ET CONES DANS UNE CATEGORIE H



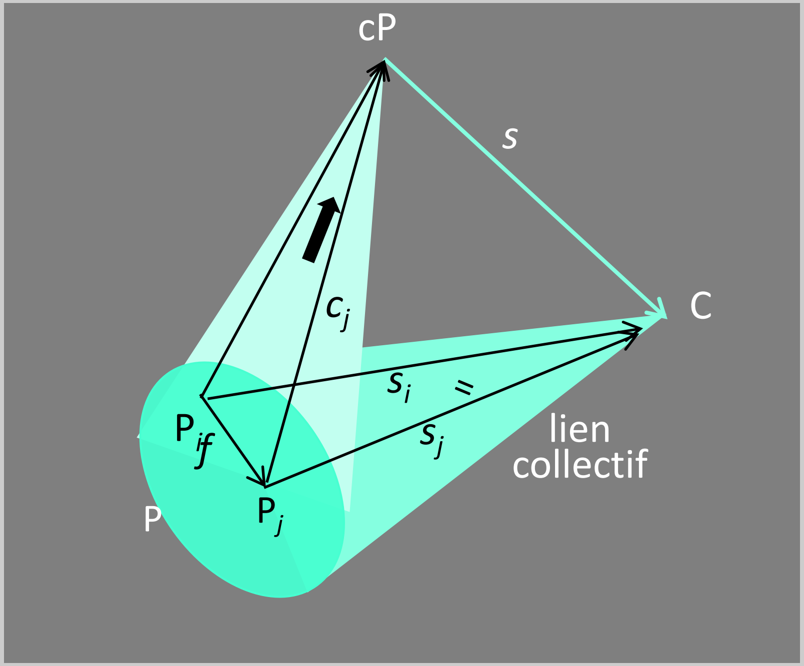
Pattern P = homomorphisme d'un graphe sP vers H, aussi appelé *diagramme*
= famille d'objets P_i avec des liens distingués entre eux.

Cône (inductif) de base P et de sommet C = famille de flèches s_i: $P_i \to C$ corrélées par les liens distingués de P :

pour tout $P(x)$: $P_i \to P_j$ on a $s_i = P(x)s_j$ (triangle commutatif).

On dit aussi *lien collectif de* P vers C, interprété comme représentant une action collective des P_i sur C, en accord avec leurs liens distingués.

COLIMITES ET SOMMES

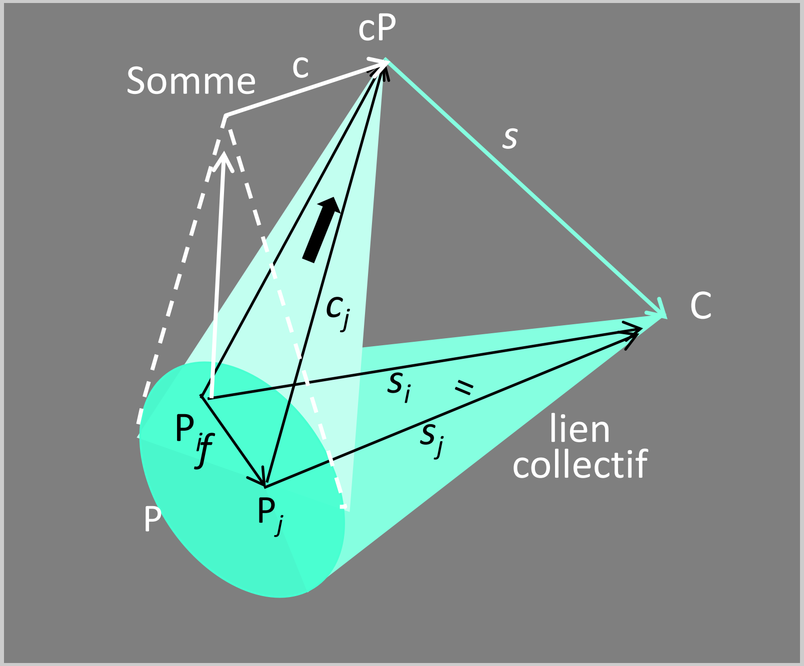


P admet une *colimite* cP s'il existe un lien collectif (c_i) de P vers cP par lequel tout lien collectif (s_i) de P vers un C se factorise via un *u*nique

$$s : cP \rightarrow C \text{ tel que } s_i = c_i s \text{ pour tout } i.$$

cP a le même rôle fonctionnel que P agissant collectivement ; il correspond à un 'recollement' du pattern P, et P à une 'décomposition' de cP. Avec des délais de propagation, cP est activé *après* tous les P_i .

COLIMITES ET SOMMES



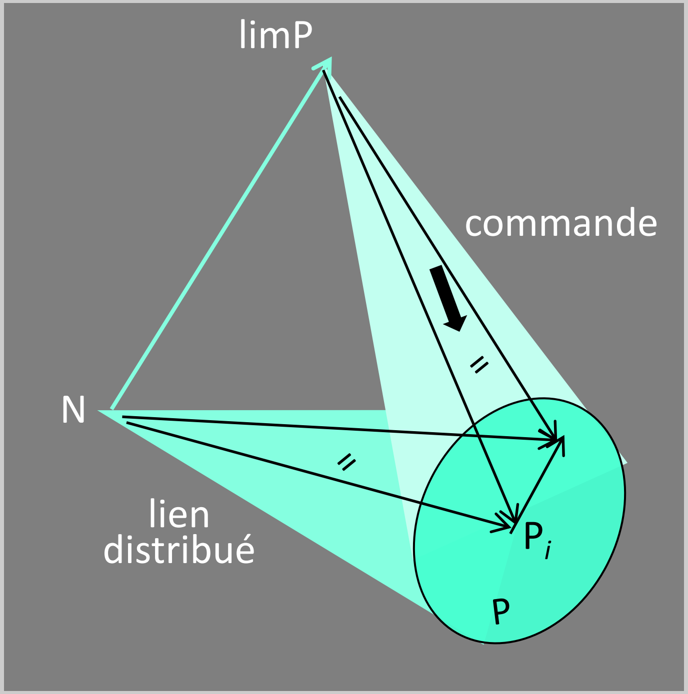
P admet une *colimite* cP s'il existe un lien collectif (c_i) de P vers cP par lequel tout lien collectif (s_i) de P vers un C se factorise via un *u*nique

s : cP →C tel que $s_i = c_i s$ pour tout i.

cP a le même rôle fonctionnel que P agissant collectivement ; il correspond à un 'recollement' du pattern P, et P à une 'décomposition' de cP. Avec des délais de propagation, cP est activé *après* tous les P_i .

Si la famille (P_i) (sans liens distingués) a une colimite S, on l'appelle *coproduit* (ou *somme*) des P_i, et il existe c: S → cP recollant les (c_i).

LIMITES PROJECTIVES



Un *cône* projectif ou *lien distribué* de N vers un pattern P est le 'dual' d'un lien collectif. Il est interprété comme une information envoyée par N et 'distribuée' entre les P_i (ou une famille de commandes de N sur P) que P ne peut décoder que collectivement.

La *limite projective* limP de P a un rôle de *commande de* P. Elle actualise la capacité des P_i à décoder ensemble un message dont chacun ne reçoit qu'une partie. Les N → limP correspondent aux messages décodés par P.

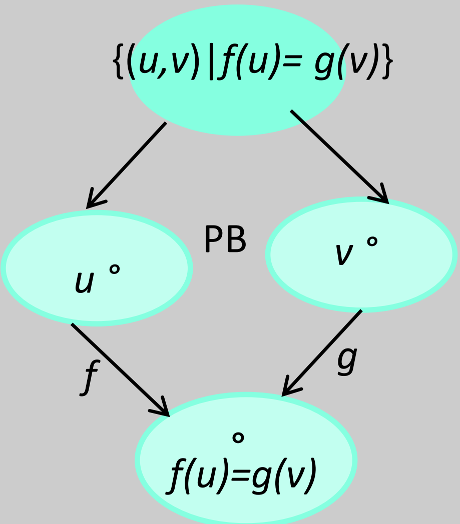
EXEMPLES DE (CO)LIMITES

1. Dans un ordre :

$$\Pi = \lim = \inf, \quad \Sigma = \mathrm{colim} = \sup$$

2. Dans Set:

$\Pi(P_i)_{i \in I} = \{(u_i) \mid u_i \in P_i \text{ pour tout } i \in I\}$.

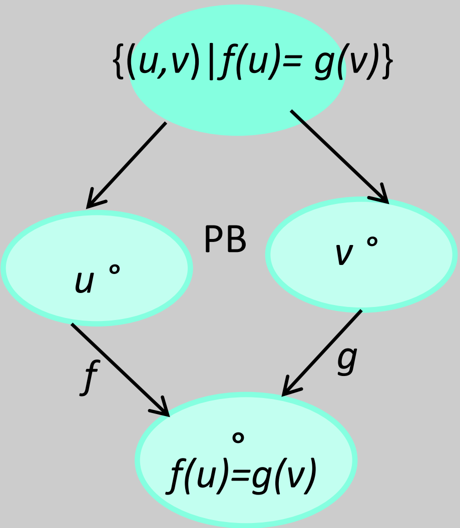


$\lim P = \{(u_i) \mid u_i \in P_i,\ P(x)(u_i) = u_j \text{ pour tout } x: i \to j \text{ dans } sP\}$

EXEMPLES DE (CO)LIMITES

1. Dans un ordre :

$$\Pi = \lim = \inf, \quad \Sigma = \mathrm{colim} = \sup$$



2. Dans Set:

$\Pi(P_i)_{i\in I} = \{(u_i) \mid u_i \in P_i \text{ pour tout } i \in I\}$.

$\lim P = \{(u_i) \mid u_i \in P_i\,,\ P(x)(u_i) = u_j \text{ pour tout } x: i \to j \text{ dans sP}\}$

$\Sigma(P_i)_{i\in I} = \{(i, u_i) \mid u_i \in P_I \text{ pour tout } i \in I\}$.

$$\mathrm{colim}\, P = \Sigma(P_i)_{i\in I}/R,$$

où R est l'équivalence engendrée par

$(i, u_i)\ R\ (j, u_j)$ s'il existe $x: i \to j$ tel que $P(x)(u_i) = u_j$

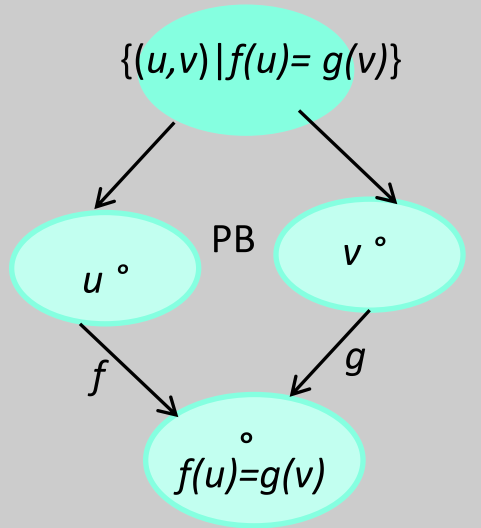
EXEMPLES DE (CO)LIMITES

1. Dans un ordre :

$$\Pi = \lim = \inf, \quad \Sigma = \mathrm{colim} = \sup$$

2. Dans Set:

$$\Pi(P_i)_{i \in I} = \{(u_i) \mid u_i \in P_i \text{ pour tout } i \in I\}.$$



$$\lim P = \{(u_i) \mid u_i \in P_i \,, P(x)(u_i) = u_j \text{ pour tout } x\colon i \to j \text{ dans sP}\}$$

$$\Sigma(P_i)_{i \in I} = \{(i, u_i) \mid u_i \in P_I \text{ pour tout } i \in I\}.$$

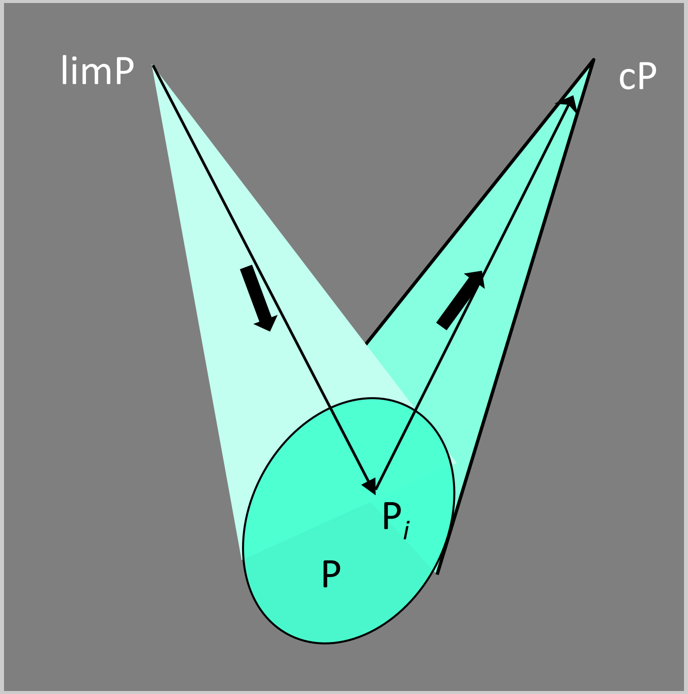
$$\mathrm{colim}\ P = \Sigma(P_i)_{i \in I}/R,$$

où R est l'équivalence engendrée par

$$(i, u_i)\ R\ (j, u_j) \text{ s'il existe } x\colon i \to j \text{ tel que } P(x)(u_i) = u_j$$

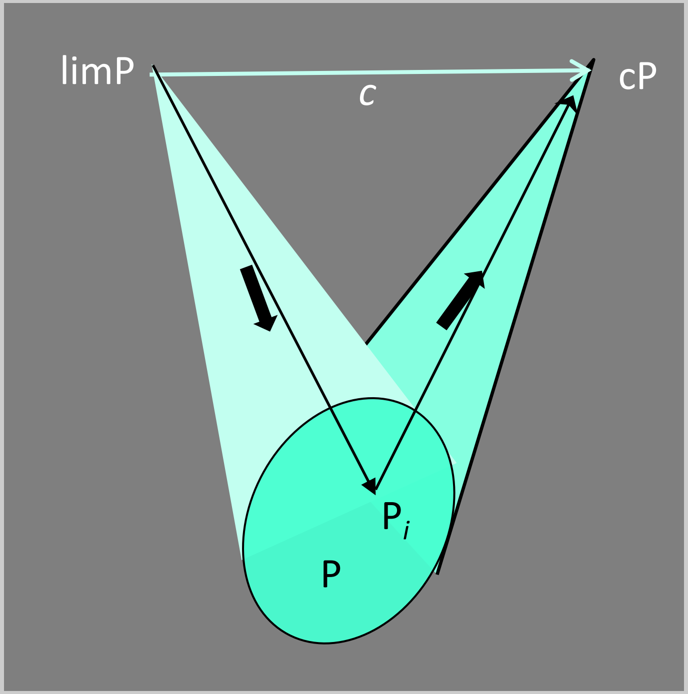
3. Dans une catégorie libre (exemple : $NEUR_t$) ou un groupoïde, un pattern connexe fini ne peut avoir qu'une (co)limite triviale. Dans Diff, il n'existe pas toujours un pullback.

LIMITE / COLIMITE



P peut avoir à la fois une limite limP et une colimite cP. Les composés de fléches canoniques de limP à P_i et de P_i à cP sont en général différents.

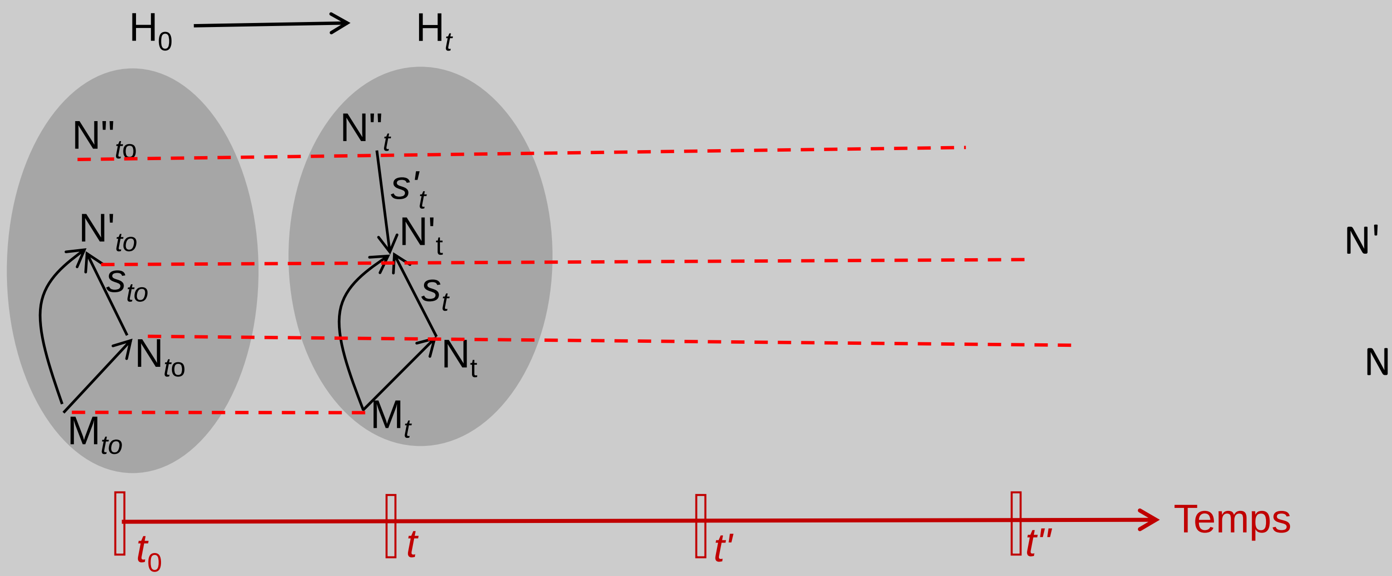
LIMITE / COLIMITE



P peut avoir à la fois une limite limP et une colimite cP. Les composés de fléches canoniques de limP à P_i et de P_i à cP sont en général différents.

Mais si de plus P est connexe, alors ils sont tous égaux à un unique lien 'canonique’ *c* de limP vers cP.

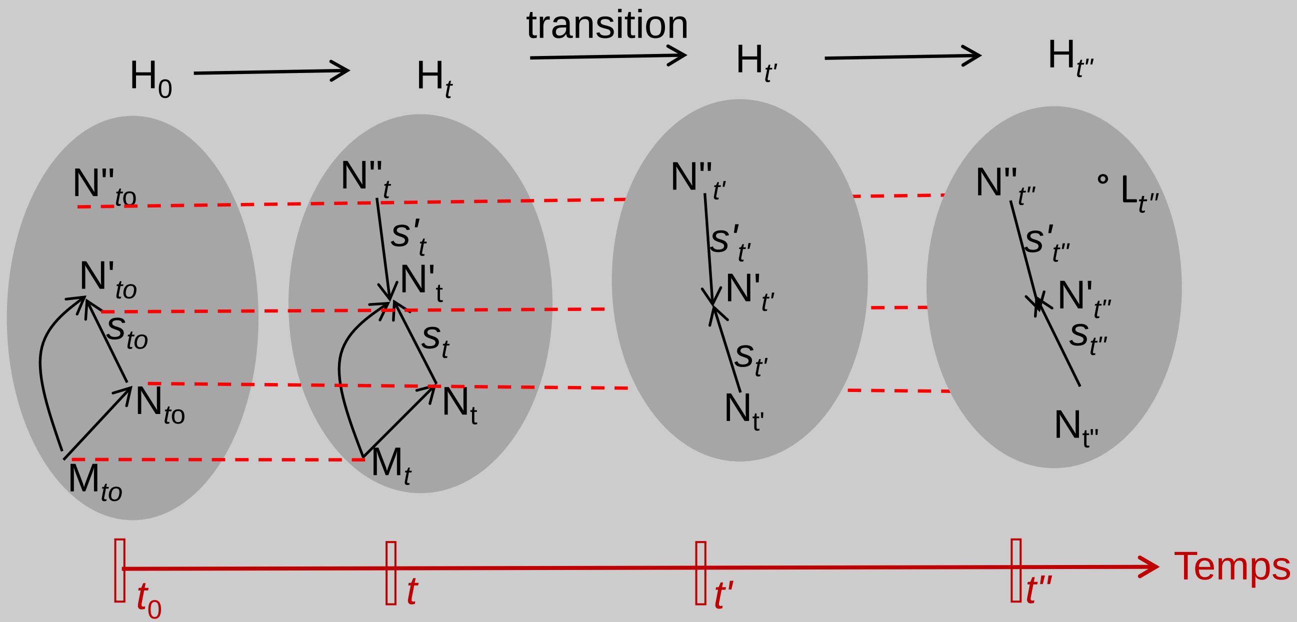
SYSTÈME ÉVOLUTIF



Un *système évolutif* **H** est formé de : 1. E*chelle de temps* T contenue dans $\mathbf{R}_+$;

2. pour chaque *t* de T une catégorie H_t *configuration* en *t.*

SYSTÈME ÉVOLUTIF

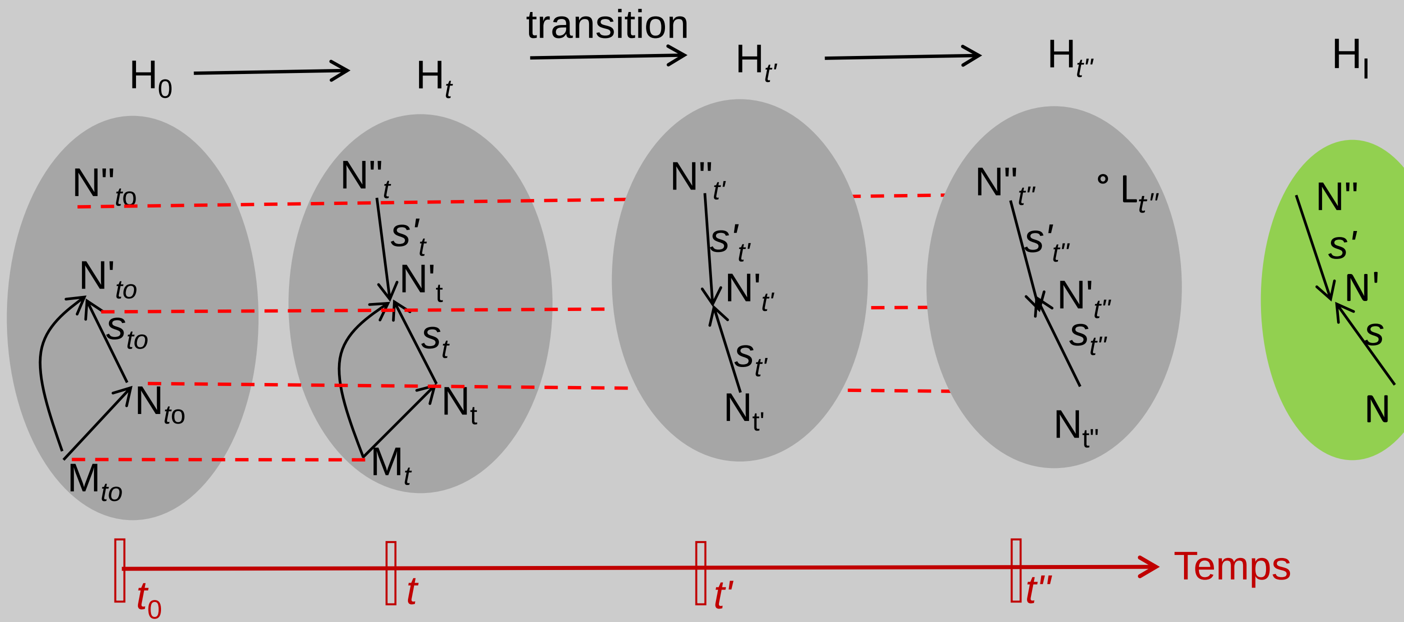


Un *système évolutif* **H** est formé de : 1. E*chelle de temps* T contenue dans $\mathbf{R}_+$;

2. pour chaque t de T une catégorie H_t *configuration* en t.

3. pour $t < t'$, un foncteur *transition* $k_{t,t'}$ d'une sous-catégorie de H_t vers $H_{t'}$ vérifiant : T*ransitivité* : Si N_t a un état $N_{t'} = k_{t,t'}(N_t)$ en t', alors il a aussi un état en t'' : $N_{t''} = k_{tt''}(N_t)$ ssi $N_{t'}$ a $N_{t''}$ pour état en t''.

SYSTÈME ÉVOLUTIF



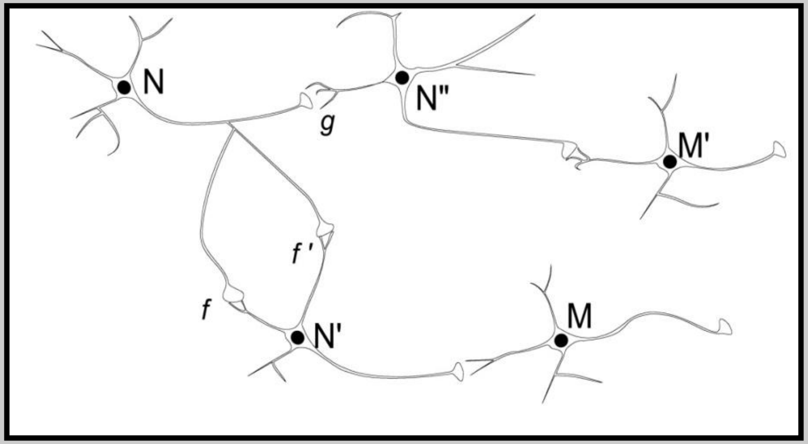
Un *système évolutif* **H** est formé de : 1. E*chelle de temps* T contenue dans $\mathbf{R}_+$;

2. pour chaque t de T une catégorie H_t *configuration* en t.

3. pour $t < t'$, un foncteur *transition* $k_{t,t'}$ d'une sous-catégorie de H_t vers $H_{t'}$ vérifiant : T*ransitivité* : Si N_t a un état $N_{t'} = k_{t,t'}(N_t)$ en t', alors il a aussi un état en t'' : $N_{t''} = k_{tt''}(N_t)$ ssi $N_{t'}$ a $N_{t''}$ pour état en t''.

C*omposant* N de **H** = Ensemble maximal de N_t liés par transitions. *Lien* défini de même. A tout I = $]t, t''[$ on associe la catégorie H_I des composants et liens dont le domaine d'existence contient I.

SYSTÈME EVOLUTIF DES NEURONES Neur



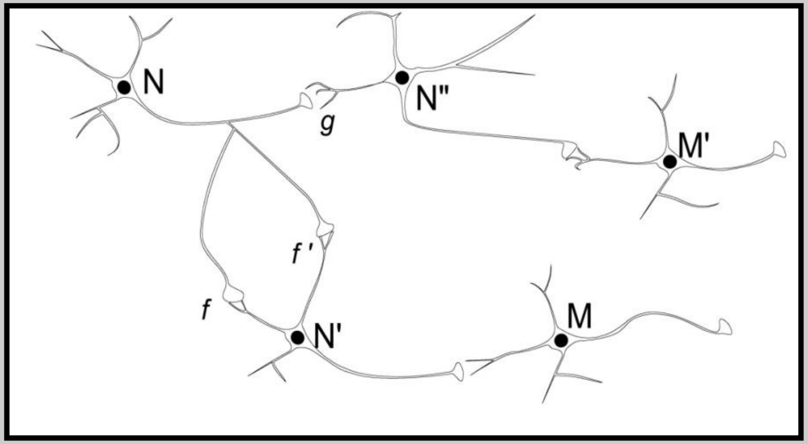
Graphe des neurones en t :

Objet N_t = état en t d'un neurone N existant en t, avec son activité en t.

Flèche $f_i : N_t \to N'_t$ = synapse de N vers N', pondérée par son *délai de propagation,* sa *force* autour de t *et un index d'activité* indiquant si la synapse est active ou non en t.

La force d'une synapse varie selon la *règle de Hebb* : elle augmente si les activités de N et N' sont corrélées autour de t.

SYSTÈME EVOLUTIF DES NEURONES Neur



Graphe des neurones en t :

Objet N_t = état en t d'un neurone N existant en t, avec son activité en t.

Flèche $f_i : N_t \to N'_t$ = synapse de N vers N', pondérée par son *délai de propagation,* sa *force* autour de t *et un index d'activité* indiquant si la synapse est active ou non en t.

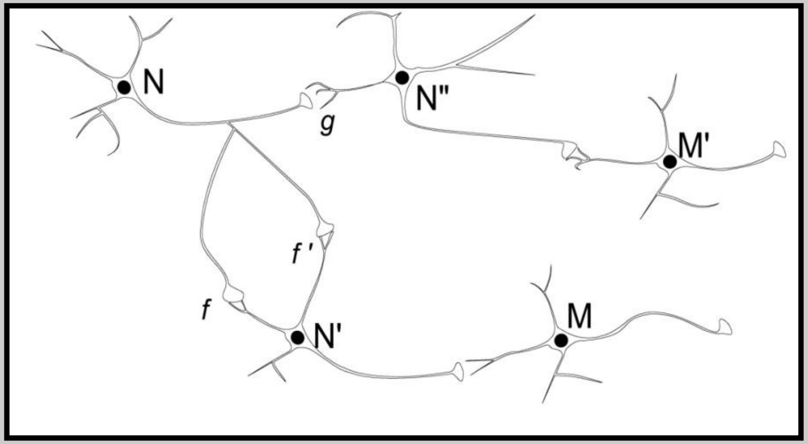
La force d'une synapse varie selon la *règle de Hebb* : elle augmente si les activités de N et N' sont corrélées autour de t.

Dans **Neur** :

= la catégorie Neur_t est la catégorie des chemins de ce graphe,

= la transition de t vers t' *r*elie l'état N_t en t d'un neurone N à son nouvel état $N_{t'}$ en t' à condition que N existe encore en t', et de même pour les chemins synaptiques.

SYSTÈME EVOLUTIF DES NEURONES Neur



Graphe des neurones en *t* :

Objet N_t = état en *t* d'un neurone N existant en *t*, avec son activité en *t.*

Flèche $f_i : N_t \rightarrow N'_t$ = synapse de N vers N', pondérée par son *délai de propagation,* sa *force* autour de *t et un index d'activité* indiquant si la synapse est active ou non en *t.*

La force d'une synapse varie selon la *règle de Hebb* : elle augmente si les activités de N et N' sont corrélées autour de *t.*

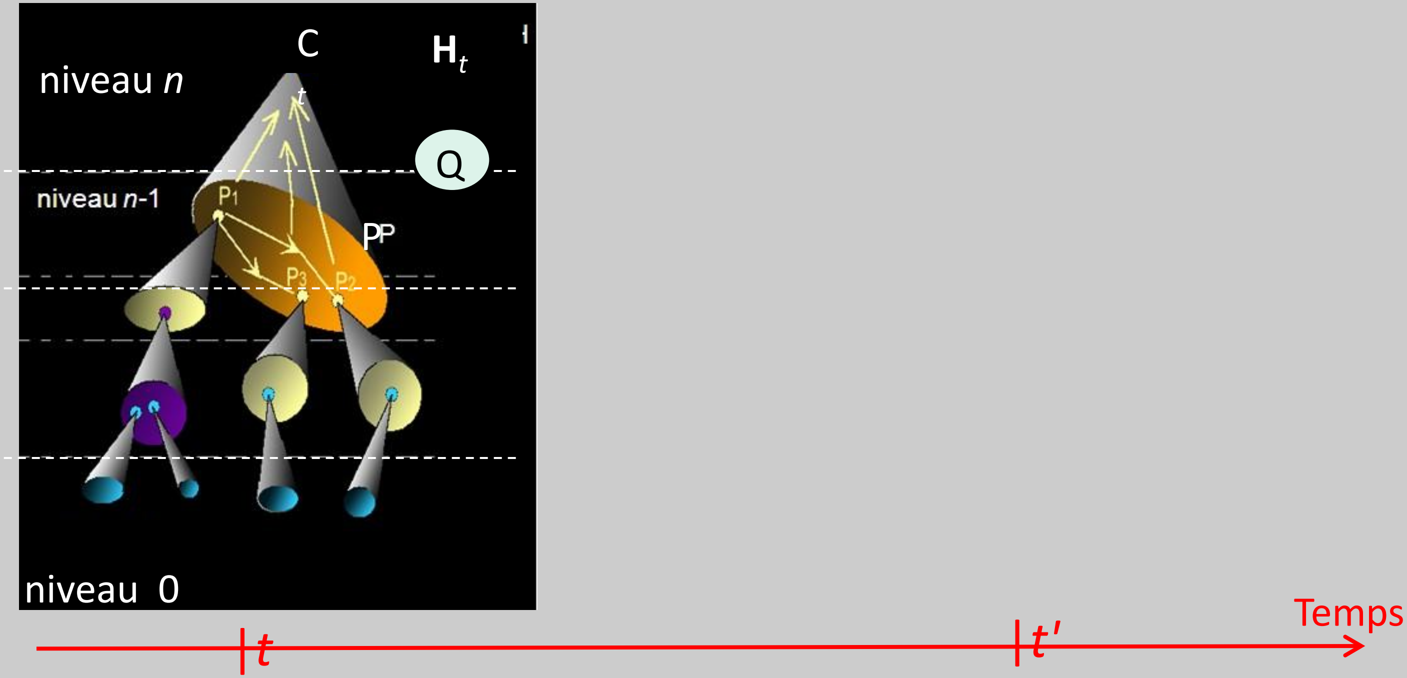
Dans **Neur** :

= **l**a catégorie $Neur_t$ est la catégorie des chemins de ce graphe,

= la transition de *t* vers *t' r*elie l'état N_t en *t* d'un neurone N à son nouvel état $N_{t'}$ en *t'* à condition que N existe encore en *t'*, et de même pour les chemins synaptiques.

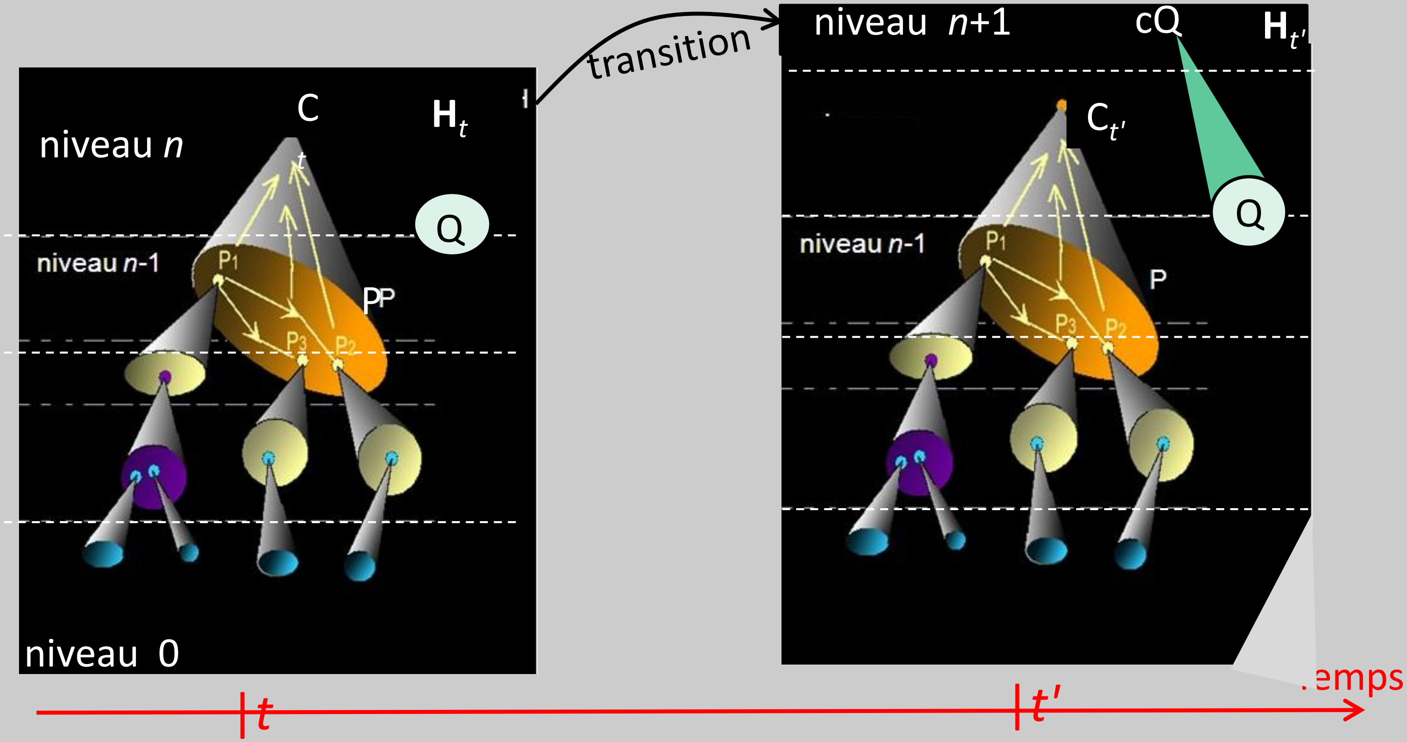
Dégénérescence du code neuronal : "More than one combination of neuronal groups can yield a particular output, and a given single group can participate in more than one kind of signaling function. …" (Edelman, 1989).

SYSTÈME HIERARCHIQUE EVOLUTF (SHE)



Catégorie hiérarchique: ses objets sont répartis en 'niveaux de complexité' de sorte que C de niveau $n > 0$ soit la (co)limite d'au moins un pattern P à valeurs dans les niveaux $< n$. Alors C admet des *ramifications* jusqu'au niveau 0 ; la longueur de la plus courte est *l'ordre de complexité* de C

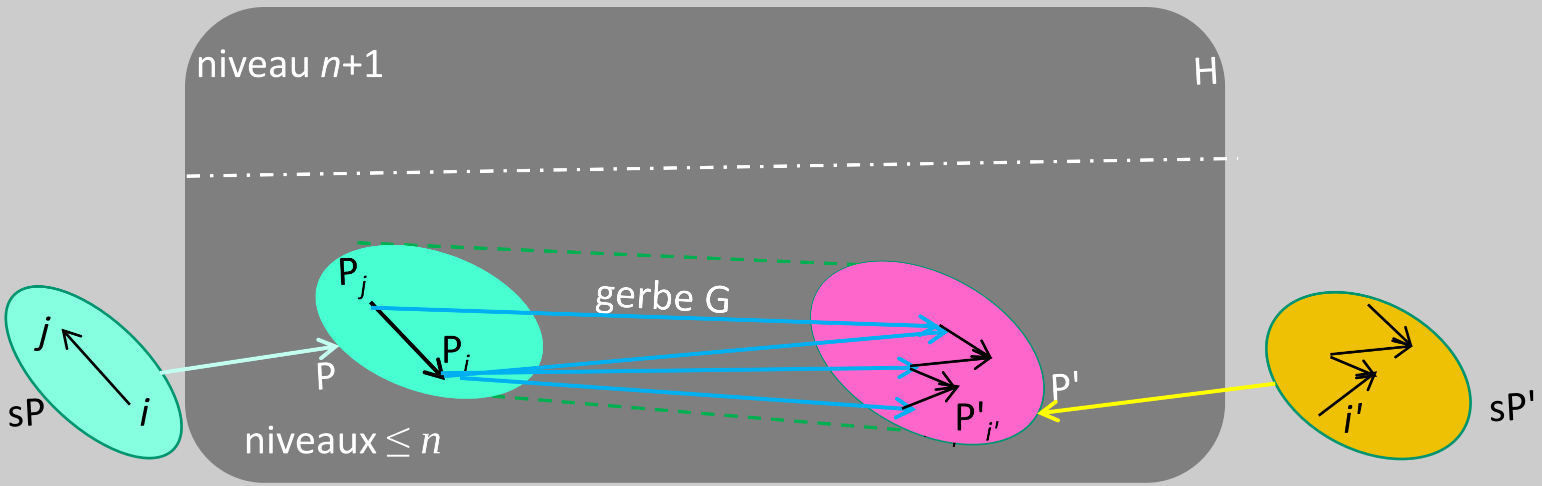
SYSTÈME HIERARCHIQUE EVOLUTF (SHE)



Catégorie hiérarchique: ses objets sont répartis en 'niveaux de complexité' de sorte que C de niveau $n > 0$ soit la (co)limite d'au moins un pattern P à valeurs dans les niveaux $< n$. Alors C admet des *ramifications* jusqu'au niveau 0 ; la longueur de la plus courte est *l'ordre de complexité* de C

SHE = système évolutif **H** tel que les H_t soient des catégories hiérarchiques et les transitions préservent le niveau. Un lien est pondéré par un *délai de propagation*, une *force* et un *index d'activité* en chaque *t.*

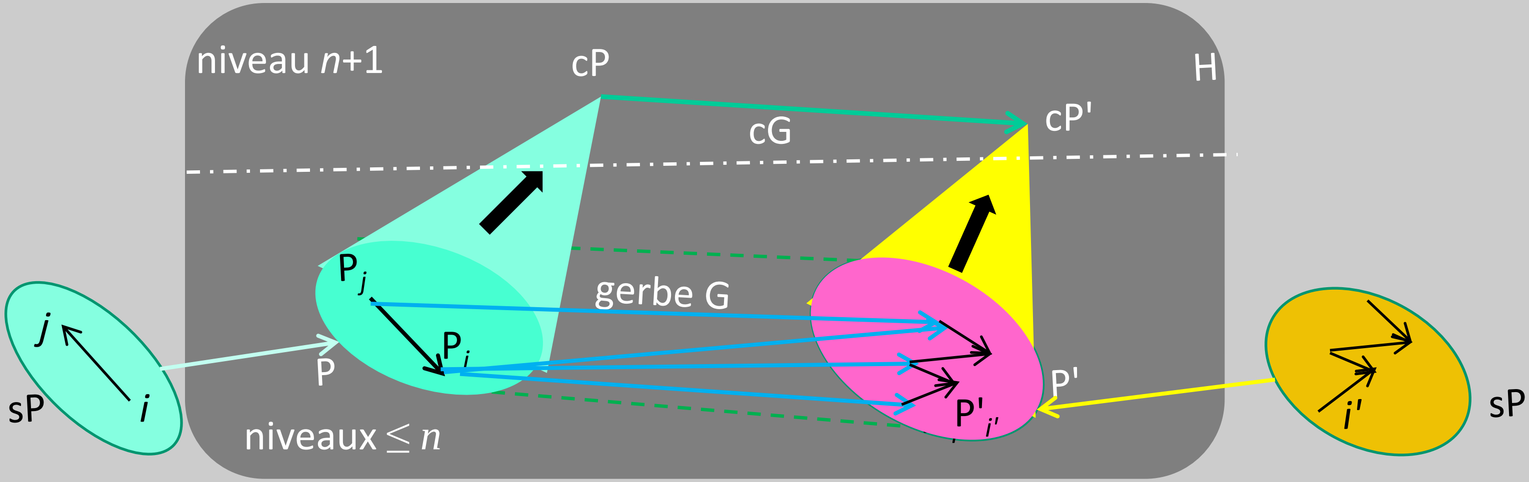
GERBES. LIENS SIMPLES



Gerbe G de P vers P' = ensemble maximal de morphismes entre P et P' tel que:

1. Tout P_i est lié par G à au moins un $P'_{i'}$ et s'il l'est à plusieurs, ils sont liés par un zigzag de liens distingués de P'.
2. Le composé d'un g ϵ G avec un lien distingué de P ou de P' est dans G.

GERBES. LIENS SIMPLES

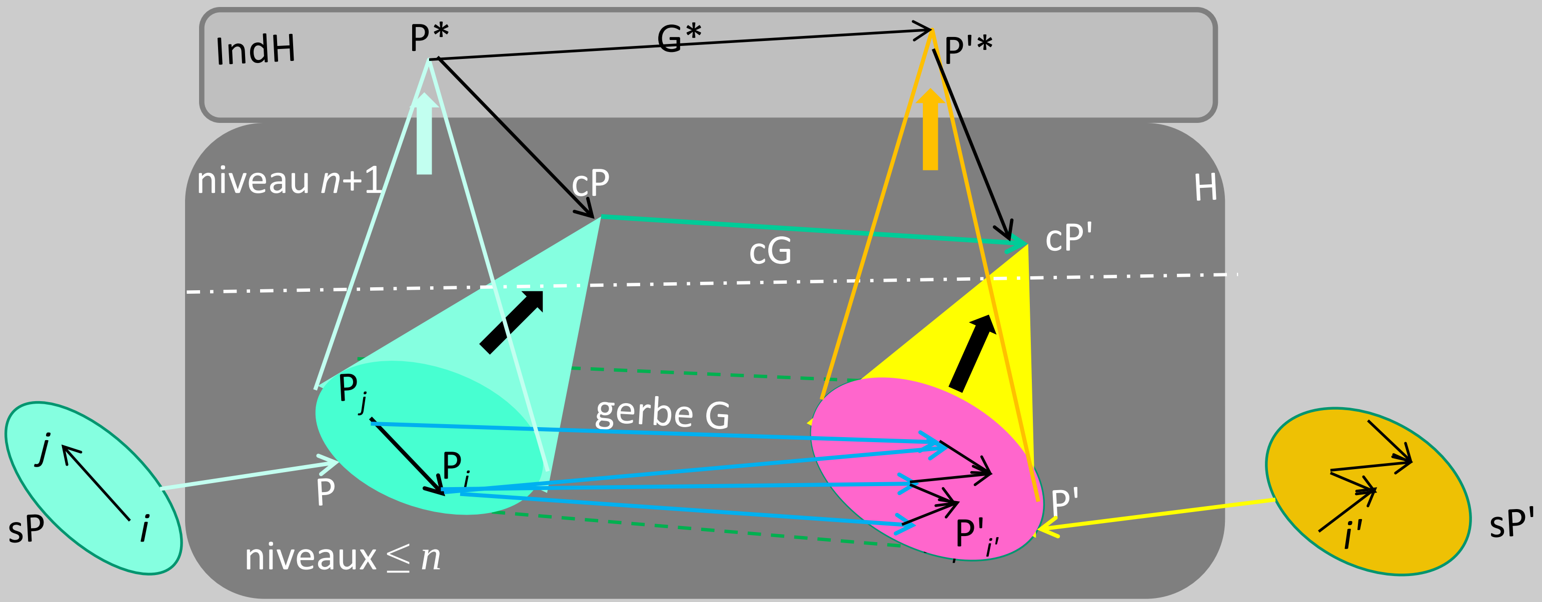


Gerbe G de P vers P' = ensemble maximal de morphismes entre P et P' tel que:

1. Tout P_i est lié par G à au moins un $P'_{i'}$ et s'il l'est à plusieurs, ils sont liés par un zigzag de liens distingués de P'.
2. Le composé d'un g ϵ G avec un lien distingué de P ou de P' est dans G.

Si P et P' ont des colimites cP et cP' dans H, G se recolle en un morphisme cG : cP → cP', appelé *lien* (P, P')-*simple,* ou *n-simple* si P et Q sont de niveaux < n+1.

GERBES. LIENS SIMPLES



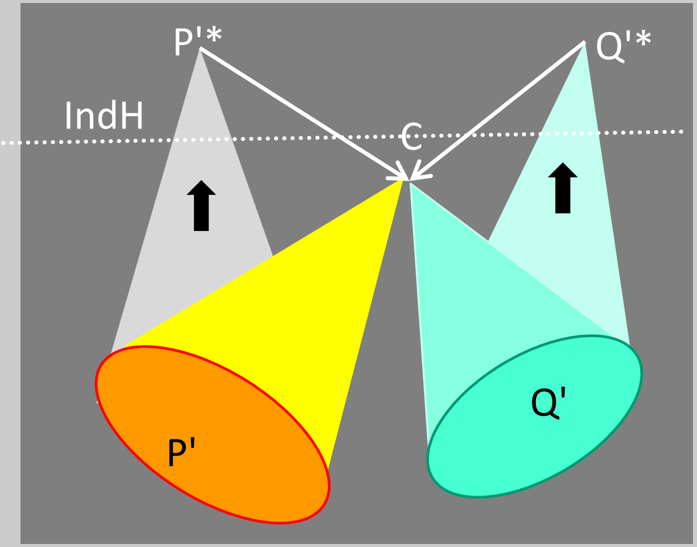
Gerbe G de P vers P' = ensemble maximal de morphismes entre P et P' tel que:

1. Tout P_i est lié par G à au moins un $P'_{i'}$ et s'il l'est à plusieurs, ils sont liés par un zigzag de liens distingués de P'.
2. Le composé d'un g ϵ G avec un lien distingué de P ou de P' est dans G.

Si P et P' ont des colimites cP et cP' dans H, G se recolle en un morphisme cG : cP → cP', appelé *lien* (P, P')-*simple,* ou *n-simple* si P et Q sont de niveaux < n+1.

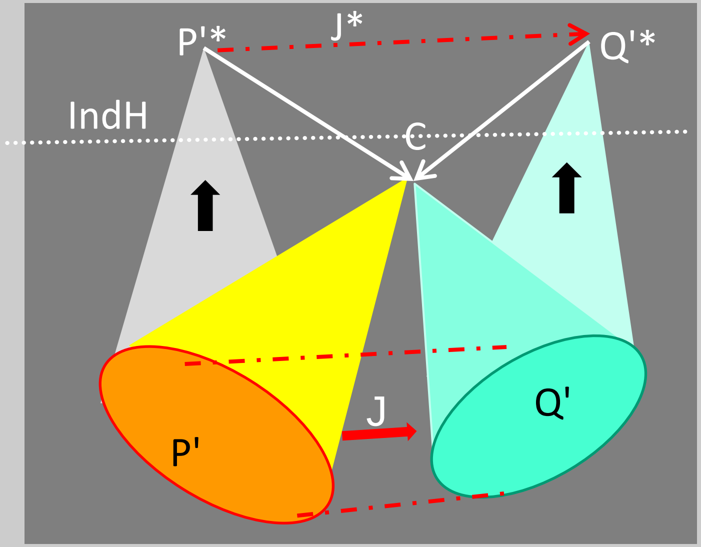
IndH = catégorie des gerbes entre patterns de H. C'est la complétion libre de H et H s'identifie à une sous-catégorie en identifiant C au pattern réduit à C.

PATTERNS (NON-)CONNECTES



Des patterns P' et Q' sont *homologues* si les catégories P'*↓H et Q'*↓H sont isomorphes (exemple : si P' et Q' ont la même colimite C).

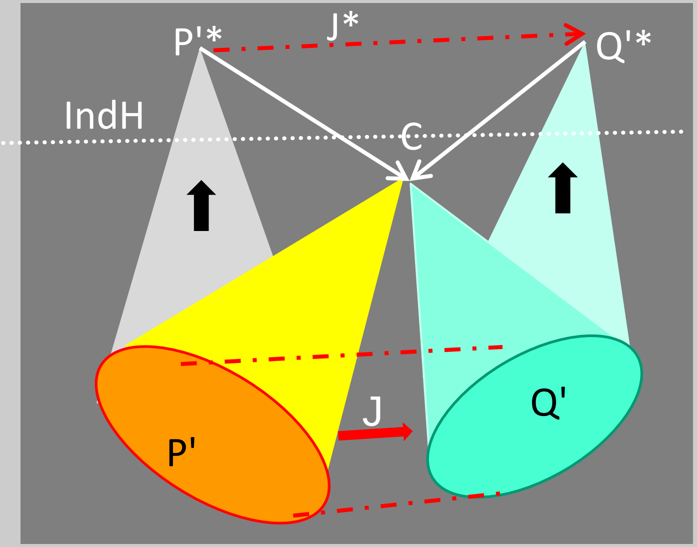
PATTERNS (NON-)CONNECTES



Des patterns P' et Q' sont *homologues* si les catégories P'*↓H et Q'*↓H sont isomorphes (exemple : si P' et Q' ont la même colimite C).

Des patterns homologues P' et Q' sont *connectés* s'il existe une gerbe J de P' vers Q' telle que cet isomorphisme soit défini par composition avec J* ; sinon ils sont *non-connectés*. On dit que C est *multiforme* s'il admet 2 décompositions non-connectées P' et Q'.

PATTERNS (NON-)CONNECTES

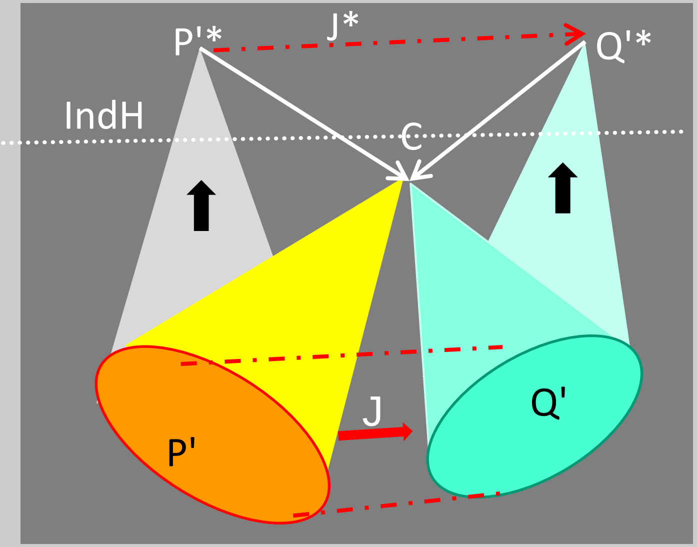


Des patterns P' et Q' sont *homologues* si les catégories P'*↓H et Q'*↓H sont isomorphes (exemple : si P' et Q' ont la même colimite C).

Des patterns homologues P' et Q' sont *connectés* s'il existe une gerbe J de P' vers Q' telle que cet isomorphisme soit défini par composition avec J* ; sinon ils sont *non-connectés*. On dit que C est *multiforme* s'il admet 2 décompositions non-connectées P' et Q'.

Par passage à H^{op}, on définit de même des pro-gerbes, des patterns *pro-homologues connectés* ou non, et des objets pro-multiformes.

PATTERNS (NON-)CONNECTES



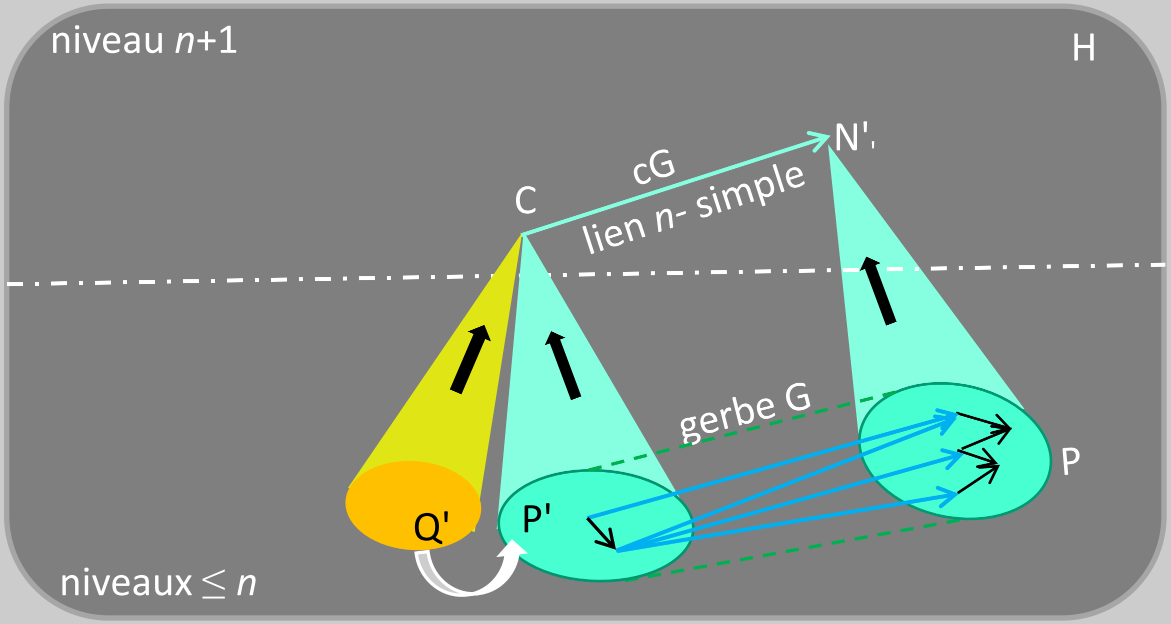
Des patterns P' et Q' sont *homologues* si les catégories P'*↓H et Q'*↓H sont isomorphes (exemple : si P' et Q' ont la même colimite C).

Des patterns homologues P' et Q' sont *connectés* s'il existe une gerbe J de P' vers Q' telle que cet isomorphisme soit défini par composition avec J* ; sinon ils sont *non-connectés*. On dit que C est *multiforme* s'il admet 2 décompositions non-connectées P' et Q'.

Par passage à H^{op}, on définit de même des pro-gerbes, des patterns *pro-homologues connectés* ou non, et des objets pro-multiformes.

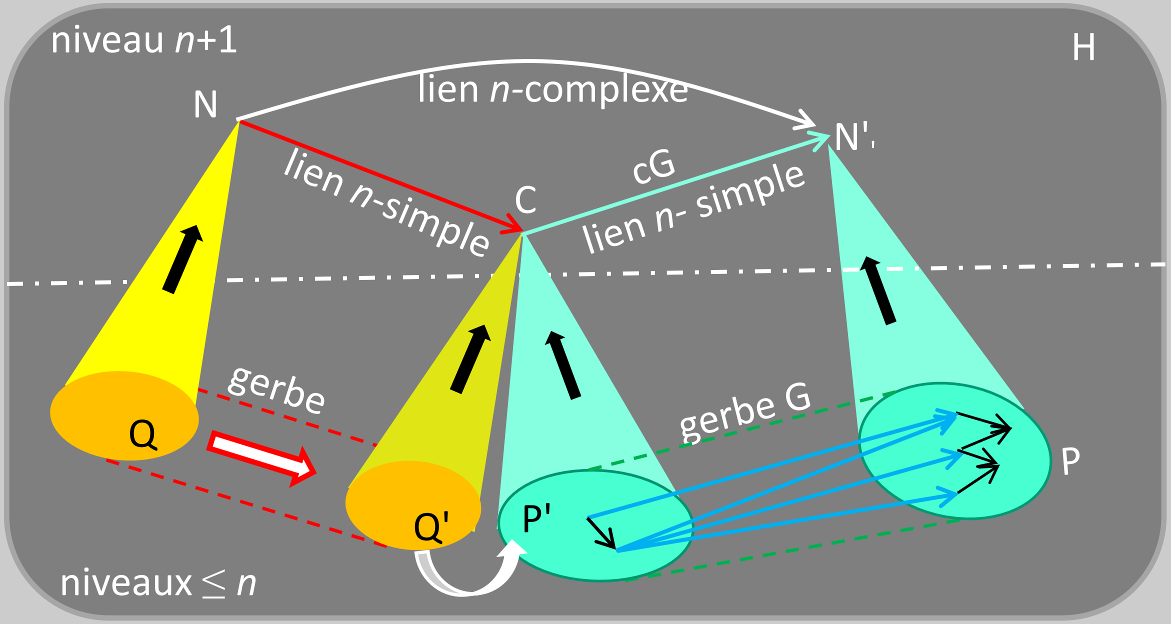
Dégénérescence neuronale = dans **Neur** il existe des patterns (= assemblées de neurones synchrones) (pro-)homologues non-connectés.

PRINCIPE DE MULTIPLICITE. LIENS COMPLEXES



H vérifie le *Principe de Multiplicité* (MP) s'il existe des objets (pro-)multiformes C La possibilité de *balancement* entre décompositions non-connectées P' et Q' de C donne de la flexibilité.

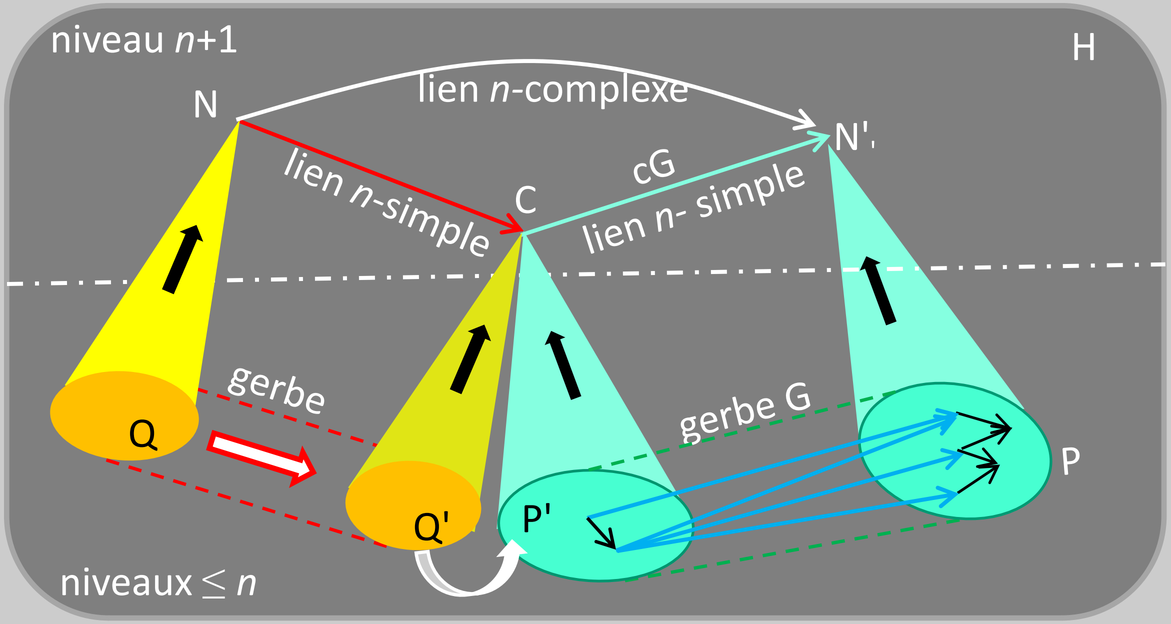
PRINCIPE DE MULTIPLICITE. LIENS COMPLEXES



H vérifie le *Principe de Multiplicité* (MP) s'il existe des objets (pro-)multiformes C La possibilité de *balancement* entre décompositions non-connectées P' et Q' de C donne de la flexibilité.

MP **==>** Emergence de *liens n-complexes* **c**omposant des liens *n*-simples recollant des gerbes non adjacentes.

PRINCIPE DE MULTIPLICITE. LIENS COMPLEXES

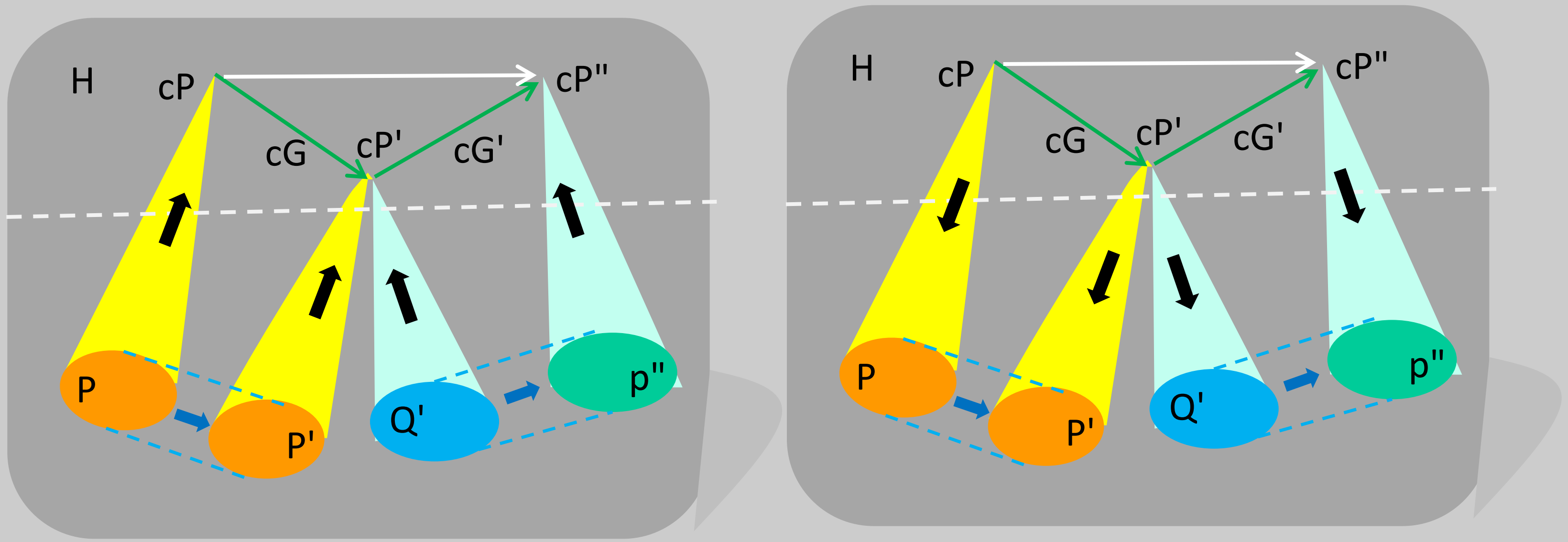


H vérifie le *Principe de Multiplicité* (MP) s'il existe des objets (pro-)multiformes C La possibilité de *balancement* entre décompositions non-connectées P' et Q' de C donne de la flexibilité.

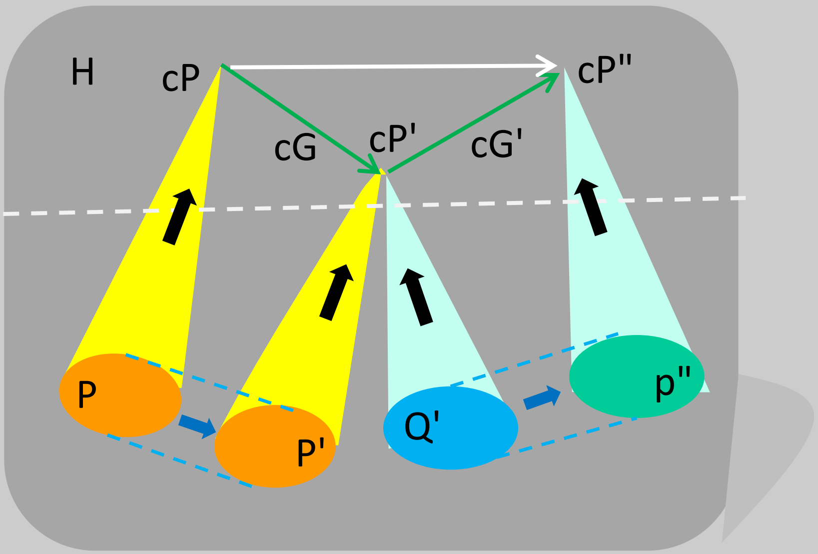
MP **==>** Emergence de *liens n-complexes* **c**omposant des liens *n*-simples recollant des gerbes non adjacentes.

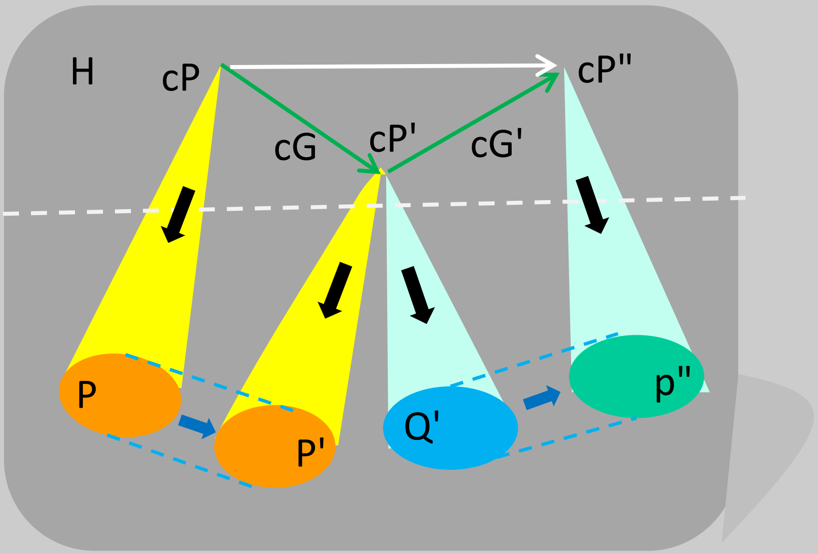
THEOREME D'EMERGENCE. *MP est nécessaire pour l'existence d'objets d'ordre de complexité > 1. Il est préservé par complexification et permet l'émergence, via complexifications successives, de composants d'ordre de complexité croissant.*

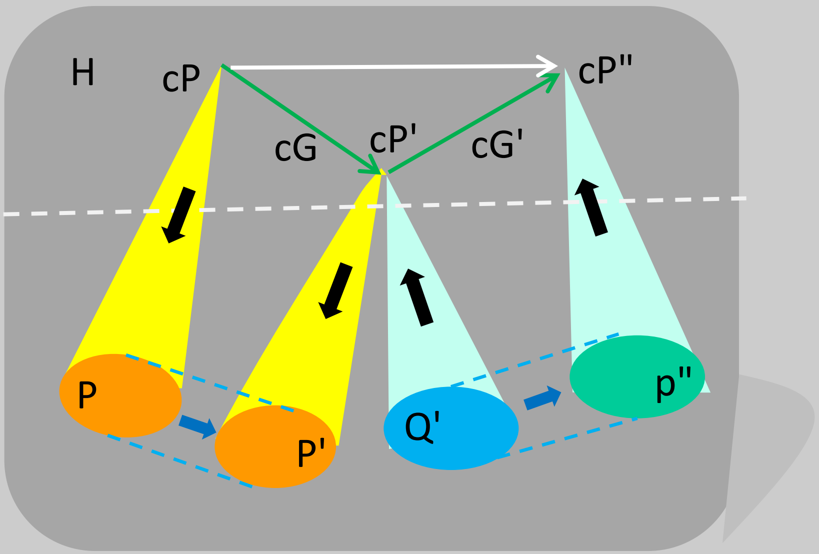
DIFFERENTES SORTES DE LIENS COMPLEXES



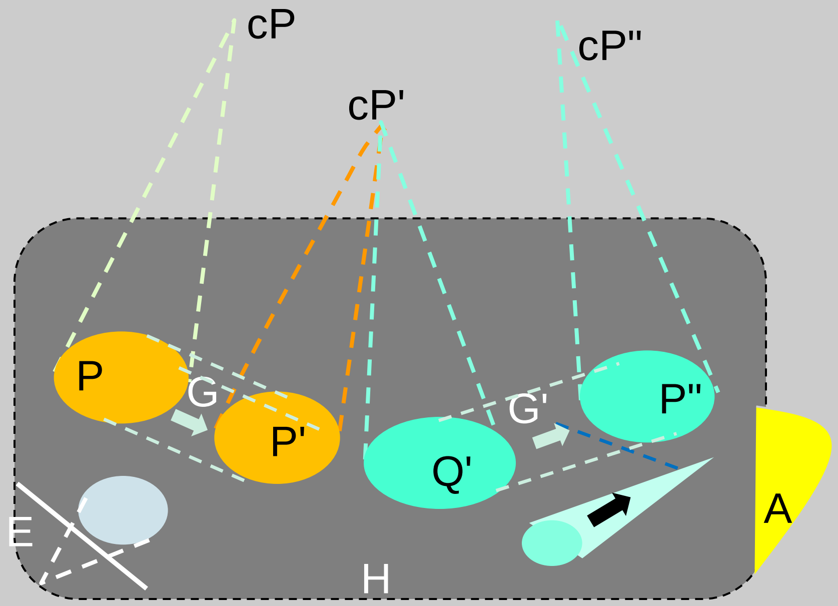
DIFFERENTES SORTES DE LIENS COMPLEXES





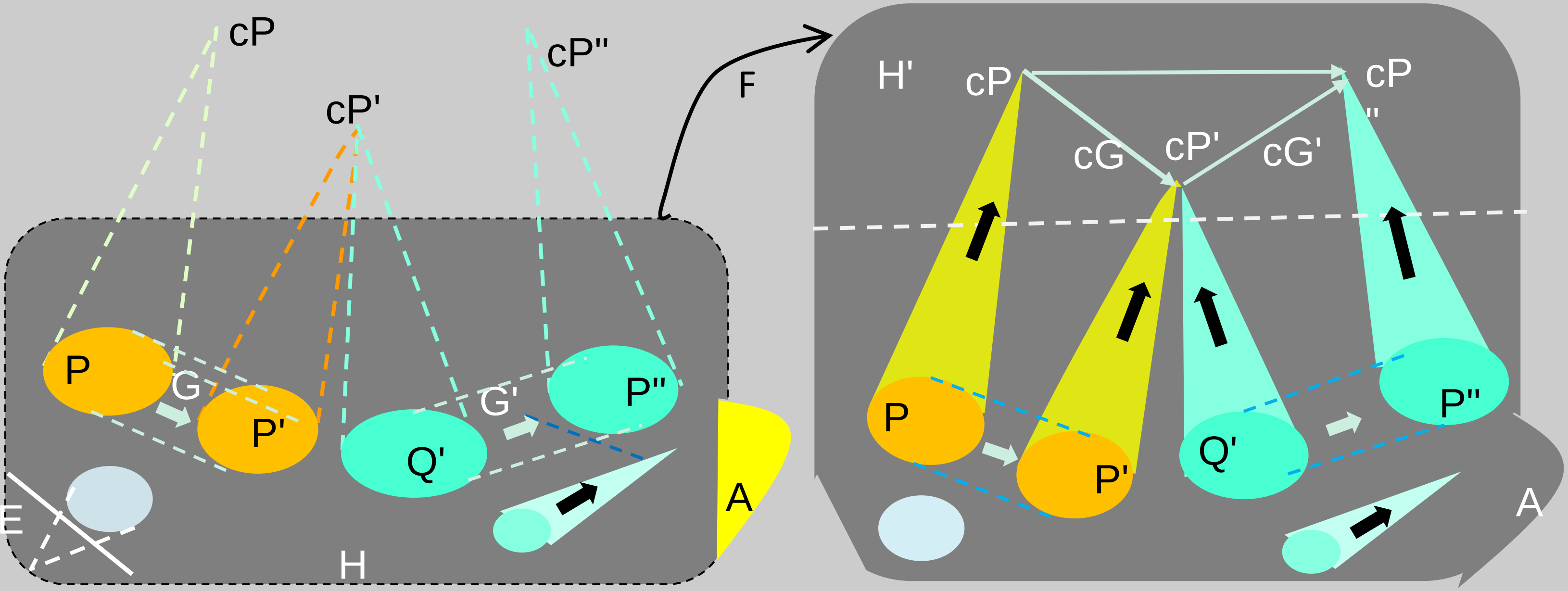


COMPLEXIFICATION POUR UNE PROCEDURE



Procédure Pr = (E, A, U, V), où : E = sous-graphe E de H, A = graphe, U = ensemble de cônes-colimite dans H, et V = ensemble de patterns P sans colimite dans H. Objectifs : 'éliminer' E, 'absorber' A, préserver les cônes-colimite de U et 'ajouter' une colimite cP' à P' ϵ V telle que cP' = cQ' si P' et Q' sont homologues. La procédure est *mixte* si on a aussi un ensemble W de patterns Q auxquels ajouter une limite.

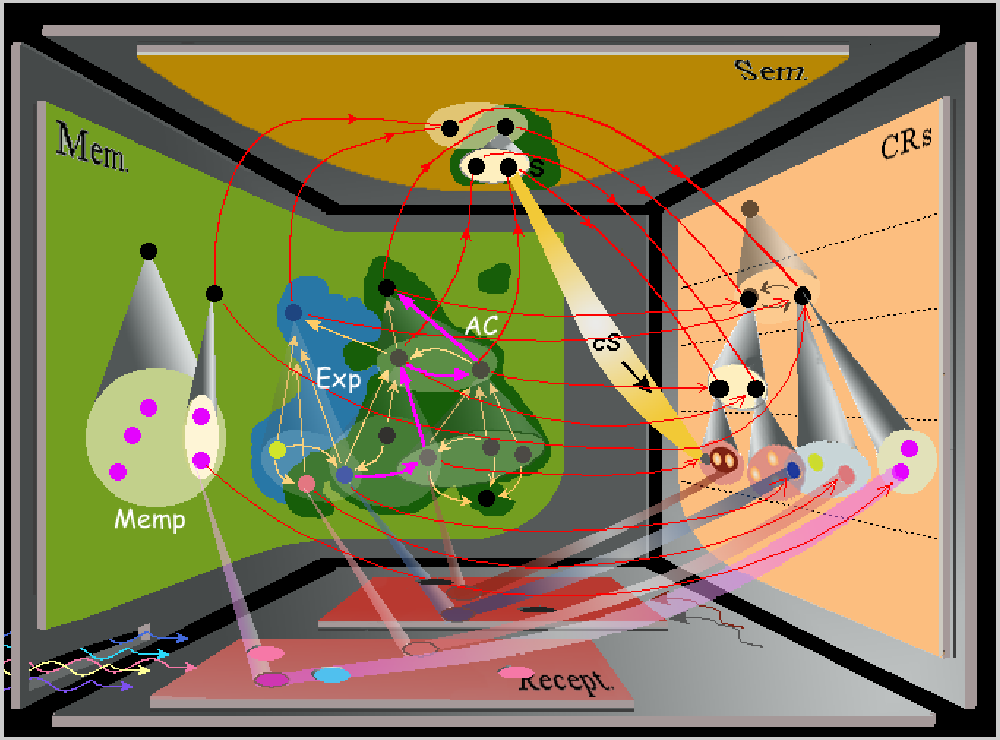
COMPLEXIFICATION POUR UNE PROCEDURE



Procédure Pr = (E, A, U, V), où : E = sous-graphe E de H, A = graphe, U = ensemble de cônes-colimite dans H, et V = ensemble de patterns P sans colimite dans H. Objectifs : 'éliminer' E, 'absorber' A, préserver les cônes-colimite de U et 'ajouter' une colimite cP' à P' є V telle que cP' = cQ' si P' et Q' sont homologues. La procédure est *mixte* si on a aussi un ensemble W de patterns Q auxquels ajouter une limite.

La *complexification* de H pour Pr est solution du problème universel : 'plonger' H\E dans une catégorie H' réalisant ces objectifs. Elle est construite comme prototype d'une certaine esquisse (A. & C. Ehresmann, 1972) .

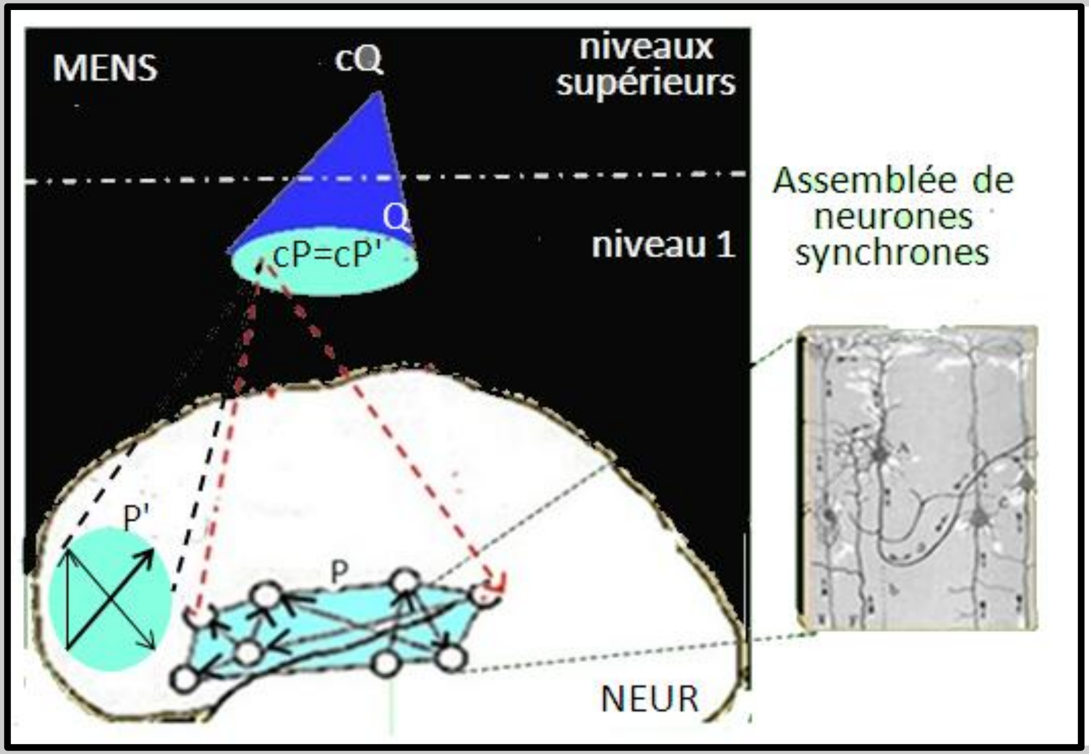
SYSTEME EVOLUTIF A MEMOIRE (SEM)



Un *SEM* est un SHE muni de :

- un sous-SHE appelé *Mémoire* **(Mem),** modélisant une mémoire flexible se développant au cours du temps ; **Mem** contient des sous-SHE, **Proc** (mémoire *procédurale*) et **Sem** (mémoire *sémantique*).
- un ensemble de sous-SE 'fonctionnels', appelés *Co-Régulateurs*, qui modulent la dynamique via leurs interactions compétitives. Chaque CR a un accès différentiel à **Mem** et à ***Proc*,**; il a sa propre échelle de temps discrète, selon laquelle il opère par étapes.

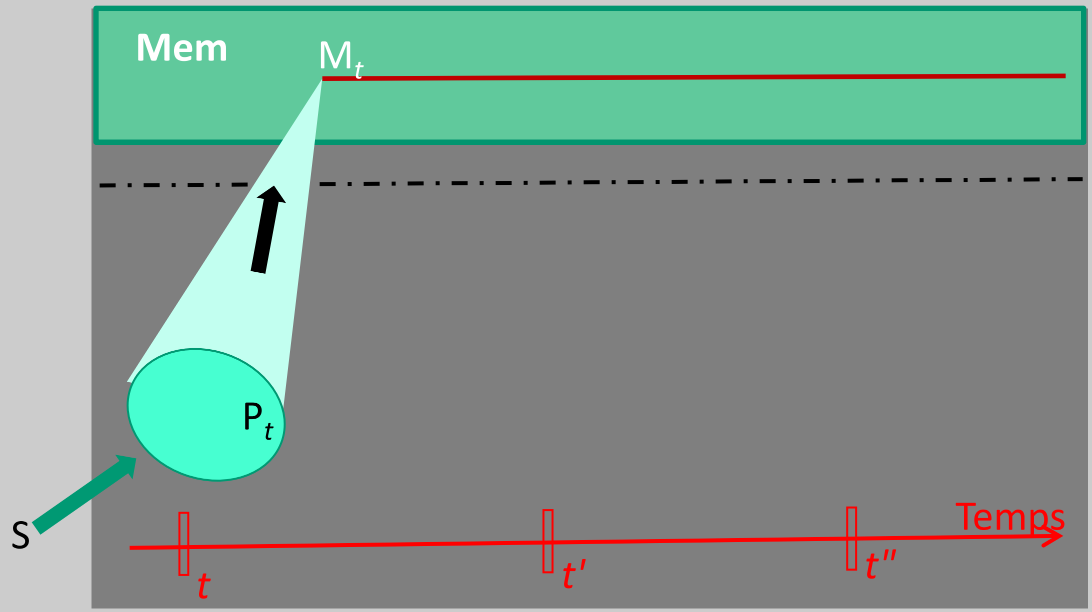
STRUCTURE GENERALE DE MENS



MENS est un SHE dont le niveau 0 est le système évolutif **NEUR** représentant le système neuronal 'physique'.

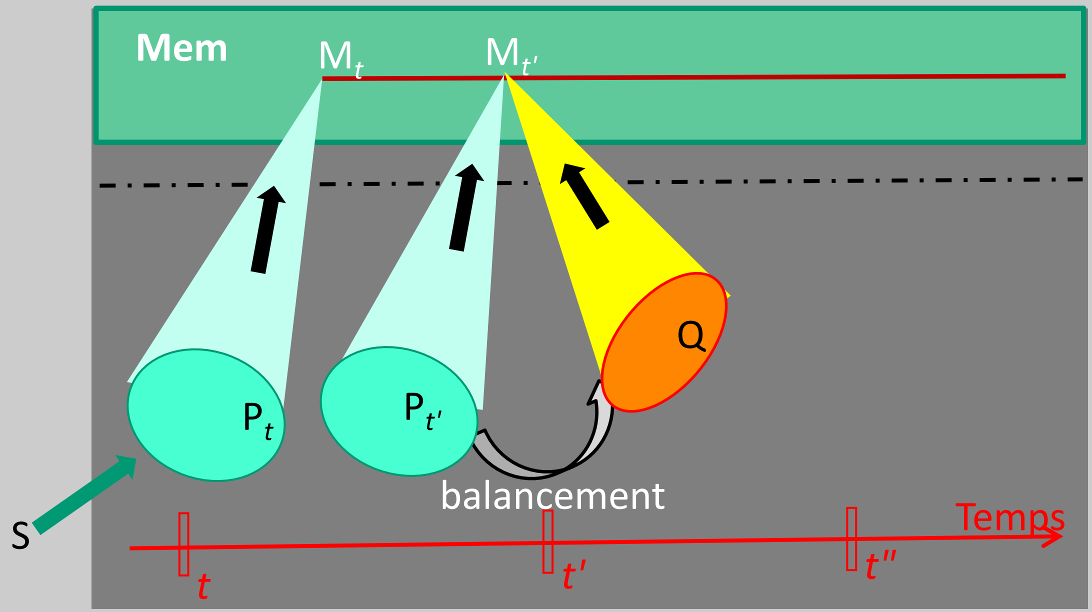
Les niveaux supérieurs sont obtenus par une suite de complexifications mixtes de **Neur**. Les composants, appelés *cat(égorie)-neurones,* représentent des objets mentaux cP sous forme de (co)limite des différentes super-assemblées neuronales P, P' qu'ils activent synchroniquement (Hebb et Edelman)

FORMATION DE LA MÉMOIRE PERCEPTUELLE



Un stimulus simple S active un pattern P. Si S se répète ou persiste, les liens distingués de P se renforcent (Hebb), et il se forme un cat-neurone M qui devient la colimite M de P dans **MENS** (via une complexification).

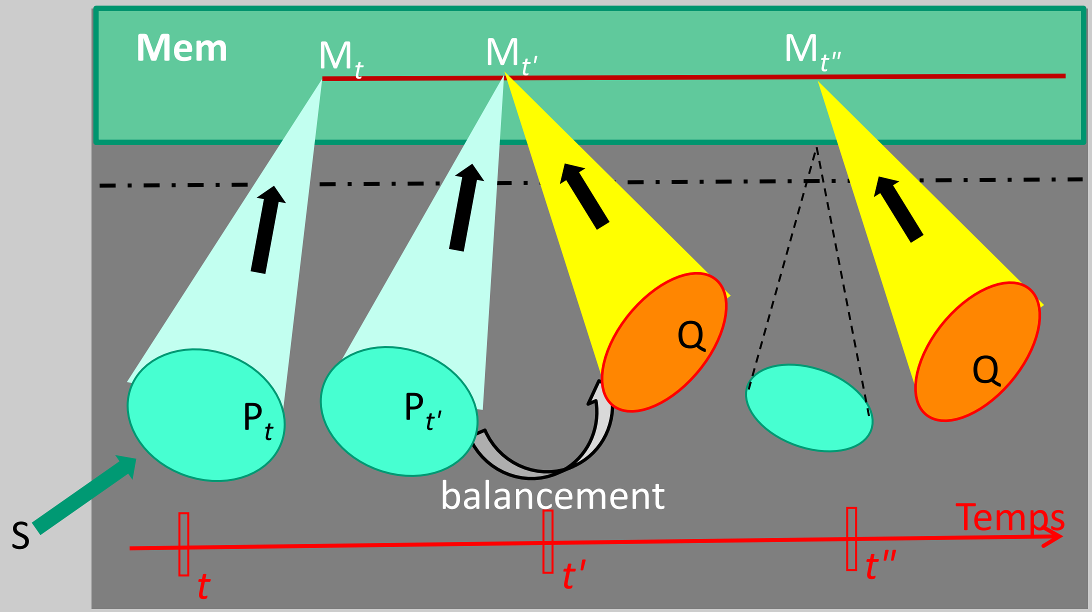
FORMATION DE LA MÉMOIRE PERCEPTUELLE



Un stimulus simple S active un pattern P. Si S se répète ou persiste, les liens distingués de P se renforcent (Hebb), et il se forme un cat-neurone M qui devient la colimite M de P dans **MENS** (via une complexification).

Dégénérescence neuronale => S peut aussi activer d'autres patterns Q. Le cat-neurone M mémoire de S est la colimite de chacun d'eux. Il prend donc sa propre identité en tant qu'*objet multiforme*.

FORMATION DE LA MÉMOIRE PERCEPTUELLE



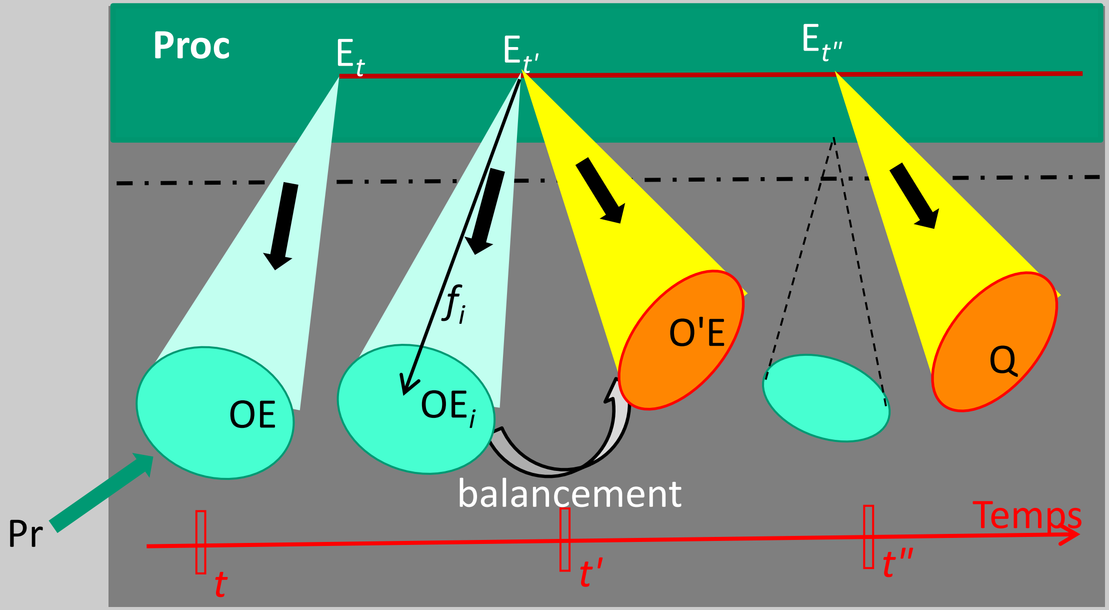
Un stimulus simple S active un pattern P. Si S se répète ou persiste, les liens distingués de P se renforcent (Hebb), et il se forme un cat-neurone M qui devient la colimite M de P dans **MENS** (via une complexification).

Dégénérescence neuronale => S peut aussi activer d'autres patterns Q. Le cat-neurone M mémoire de S est la colimite de chacun d'eux. Il prend donc sa propre identité en tant qu'*objet multiforme*.

M peut se dissocier de P au cours du temps pour s'adapter aux changements

===> *Mémoire robuste, flexible et plastique.*

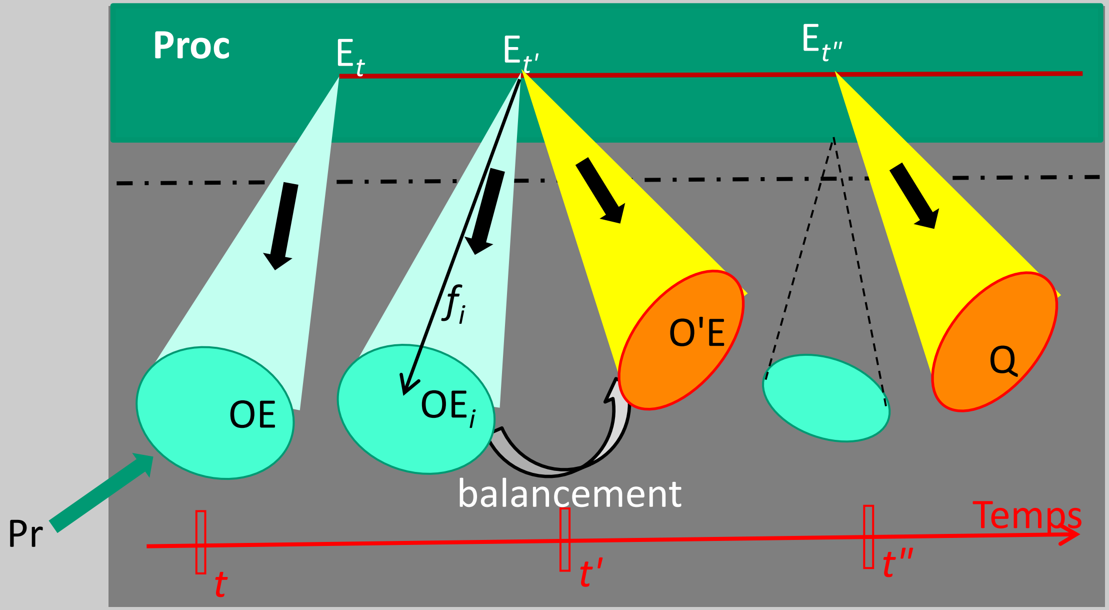
MEMOIRE PROCEDURALE Proc



Une procédure Pr active un pattern OE d'effecteurs. Par répétition, ses liens distingués se renforcent (Hebb) et elle sera mémorisée dans **Proc** par la limite E de OE. L'activation ultérieure de E activera OE via les 'commandes' f_i: E→ OE_i. Un composant de **Proc** sera appelé *procept* (cp. Gray & Tall)..

E est un objet pro-multiforme (dégénérescence neuronale) qui prend son identité et active d'autres patterns pro-homologues, puis se dissociera de OE.

MEMOIRE PROCEDURALE Proc

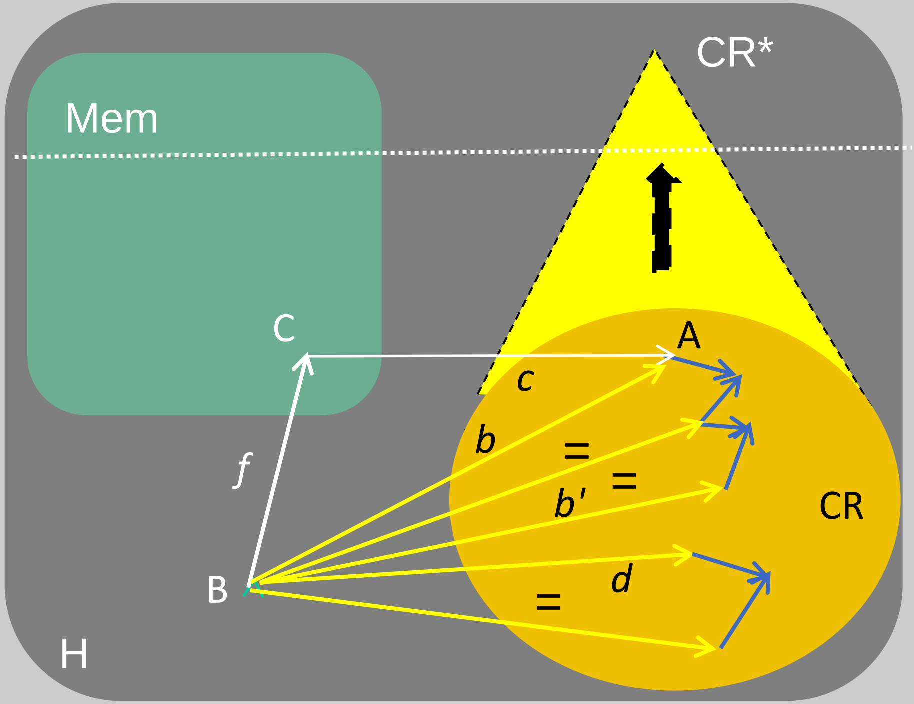


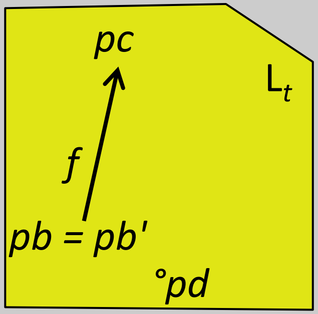
Une procédure Pr active un pattern OE d'effecteurs. Par répétition, ses liens distingués se renforcent (Hebb) et elle sera mémorisée dans **Proc** par la limite E de OE. L'activation ultérieure de E activera OE via les 'commandes' f_i: E→ OE_i. Un composant de **Proc** sera appelé *procept* (cp. Gray & Tall)..

E est un objet pro-multiforme (dégénérescence neuronale) qui prend son identité et active d'autres patterns pro-homologues, puis se dissociera de OE.

Proc se développe, à partir de procepts innés, par itération de complexi-fications mixtes dans lesquels seules des limites sont ajoutées.

PAYSAGE D'UN CO-REGULATEUR CR



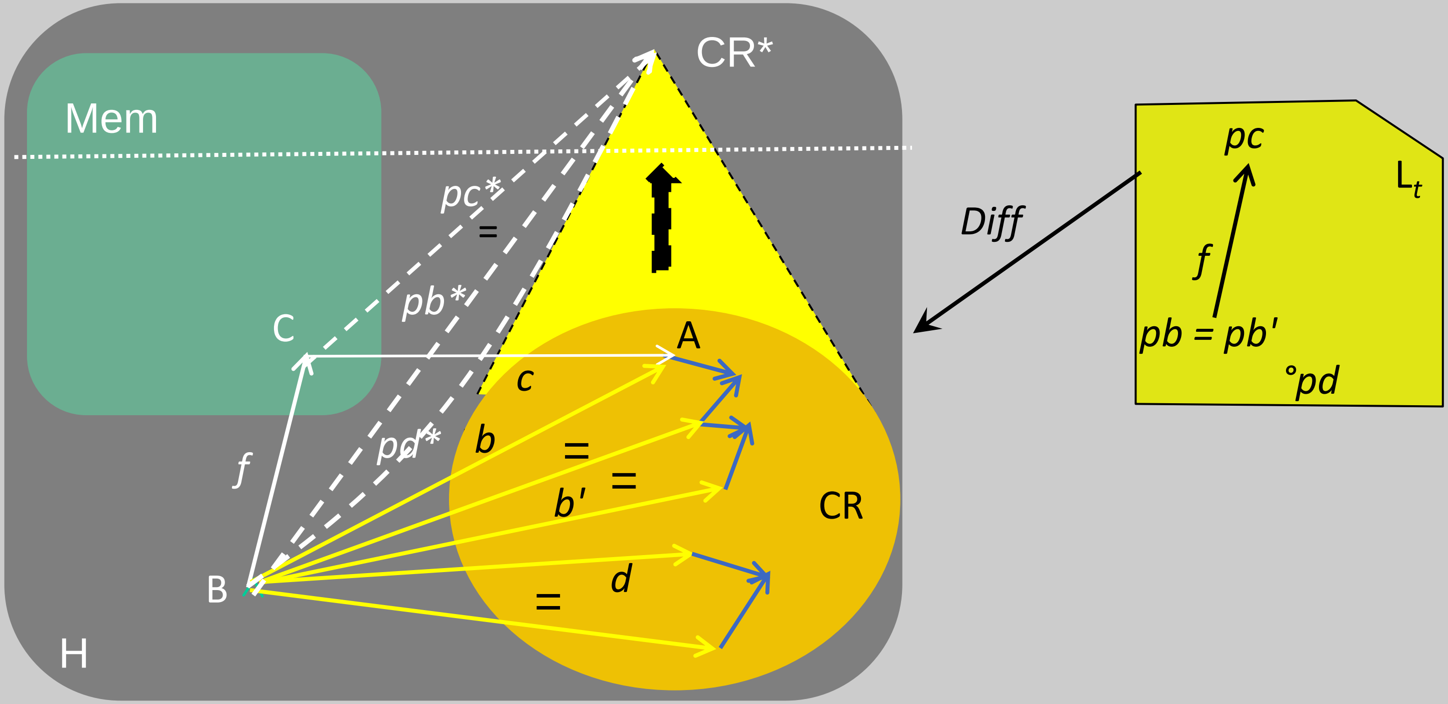


A chacune de ses étapes I = [*t,* t'[, CR le ***paysage*** de CR est une catégorie L_t :

- un objet (ou *perspective*) est une gerbe dans H_I d'un composant B vers CR I-*activée*, i.e. pour au moins un de ses liens *b* il existe un *s* dans I où *b* est actif et *s* + délai de propagation *d(b)(s)* en *s* est dans *I.*
- un lien *de pb* vers *pc* 'est' un lien *f* de B vers C corrélant les perspectives.

Le paysage est la vue partielle que CR peut avoir de **MENS** pendant l'étape.

PAYSAGE D'UN CO-REGULATEUR CR



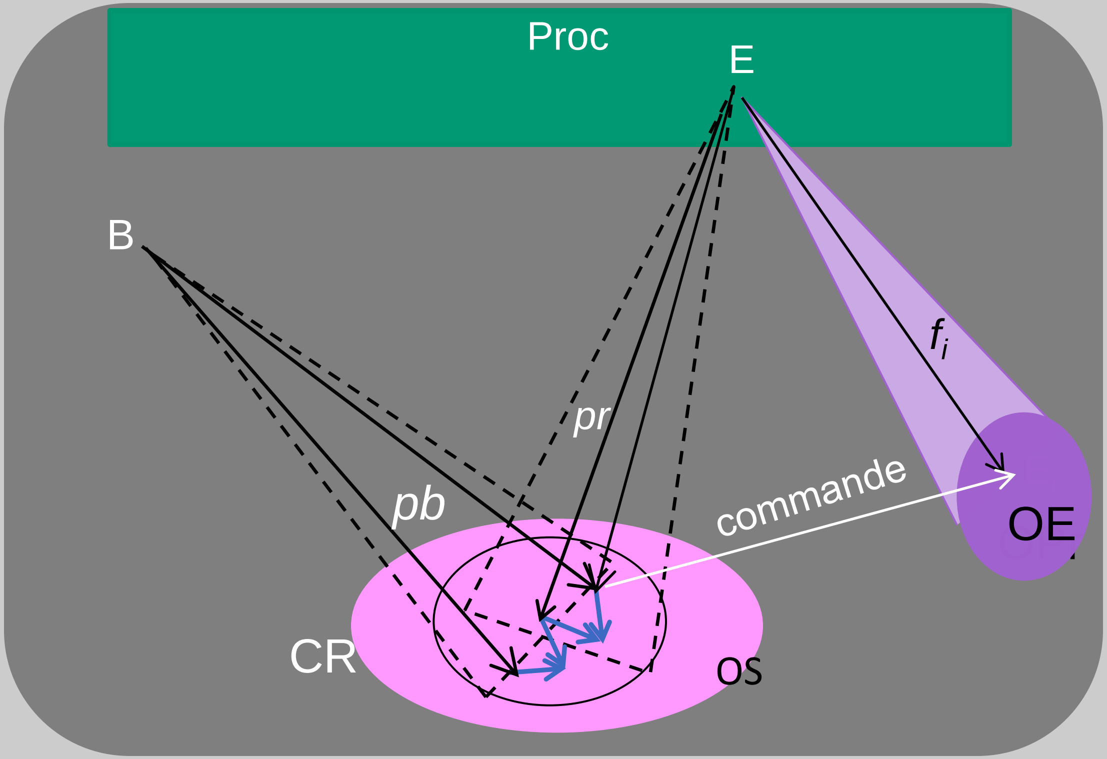
A chacune de ses étapes I = [*t,* t'[, CR le ***paysage*** de CR est une catégorie L_t :

- un objet (ou *perspective*) est une gerbe dans H_I d'un composant B vers CR I-*activée*, i.e. pour au moins un de ses liens *b* il existe un *s* dans I où *b* est actif et *s* + délai de propagation *d(b)(s)* en *s* est dans *I.*
- un lien *de pb* vers *pc* 'est' un lien *f* de B vers C corrélant les perspectives.

Le paysage est la vue partielle que CR peut avoir de **MENS** pendant l'étape.

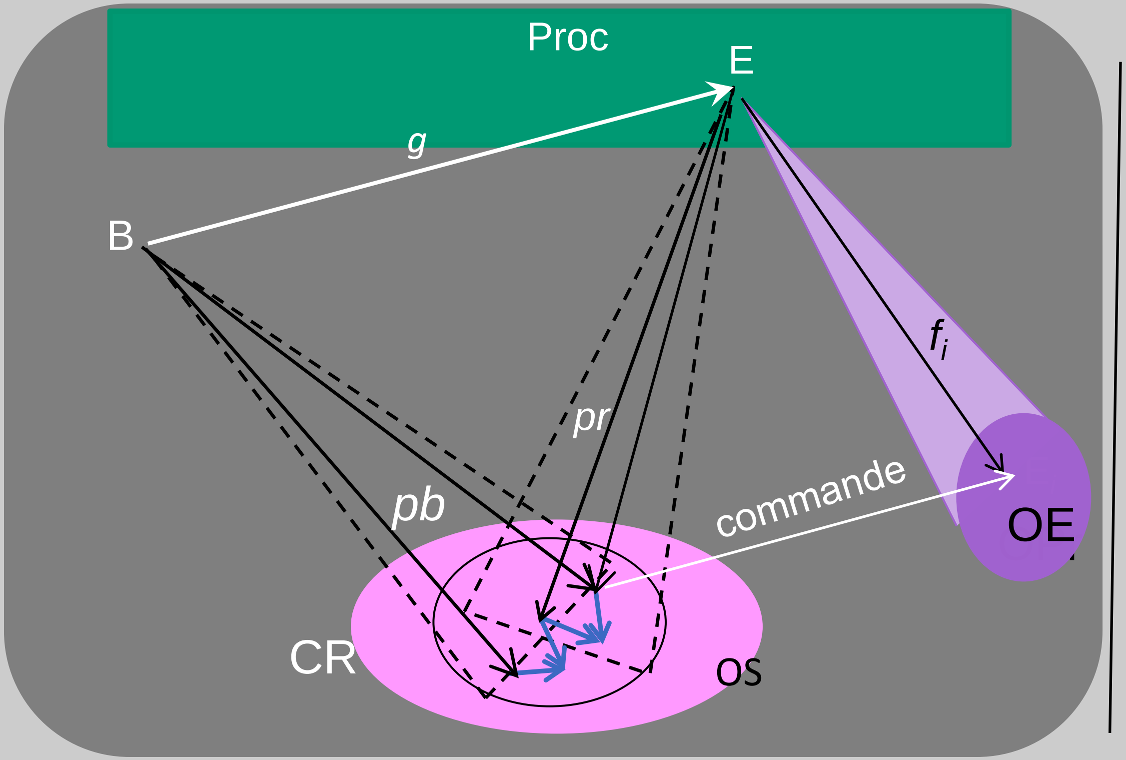
On a un foncteur *différence* *Diff* (non injectif) de L_t vers H_I. Si CR a une colimite CR*, une perspective 'est' un lien actif vers CR* factorisé par CR.

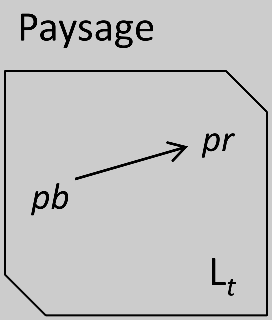
PROCEPTS ADMISSIBLE POUR UN CR



A CR est associé un ensemble de *procepts admissibles* E pour CR (relatifs à sa fonction) vérifiant : E admet une perspective *pr* pour CR telle que les commandes f_i de OS se factorisent via *pr* (=> CR peut 'commander' OS)..

PROCEPTS ADMISSIBLE POUR UN CR

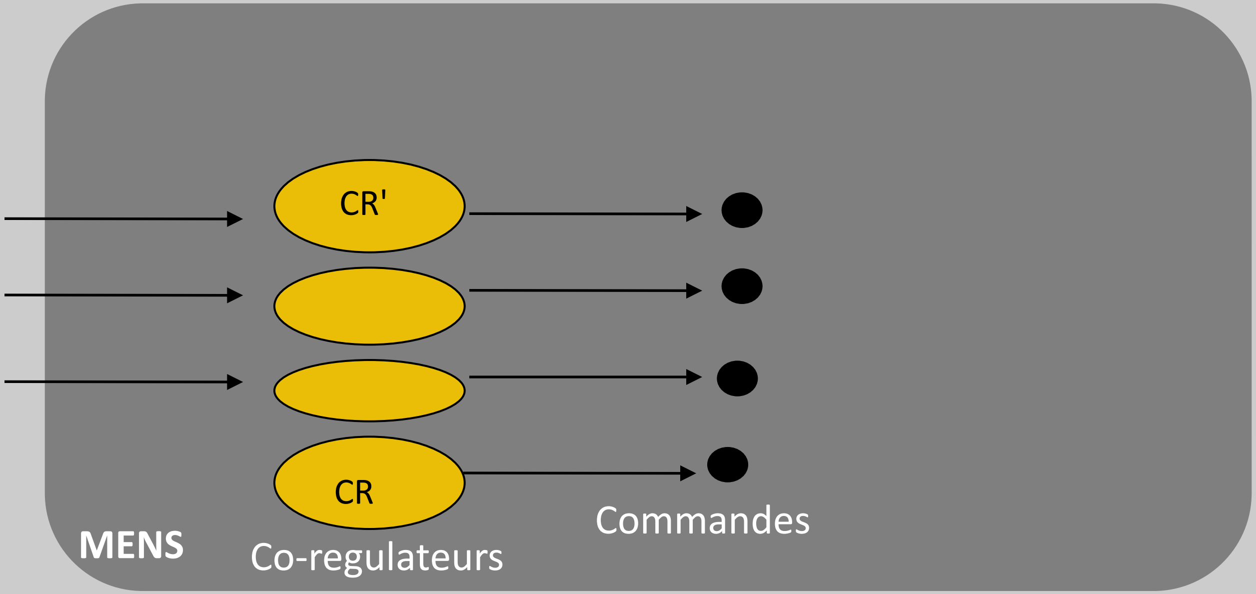




A CR est associé un ensemble de *procepts admissibles* E pour CR (relatifs à sa fonction) vérifiant : E admet une perspective *pr* pour CR telle que les commandes f_i de OS se factorisent via *pr* (=> CR peut 'commander' OS)..

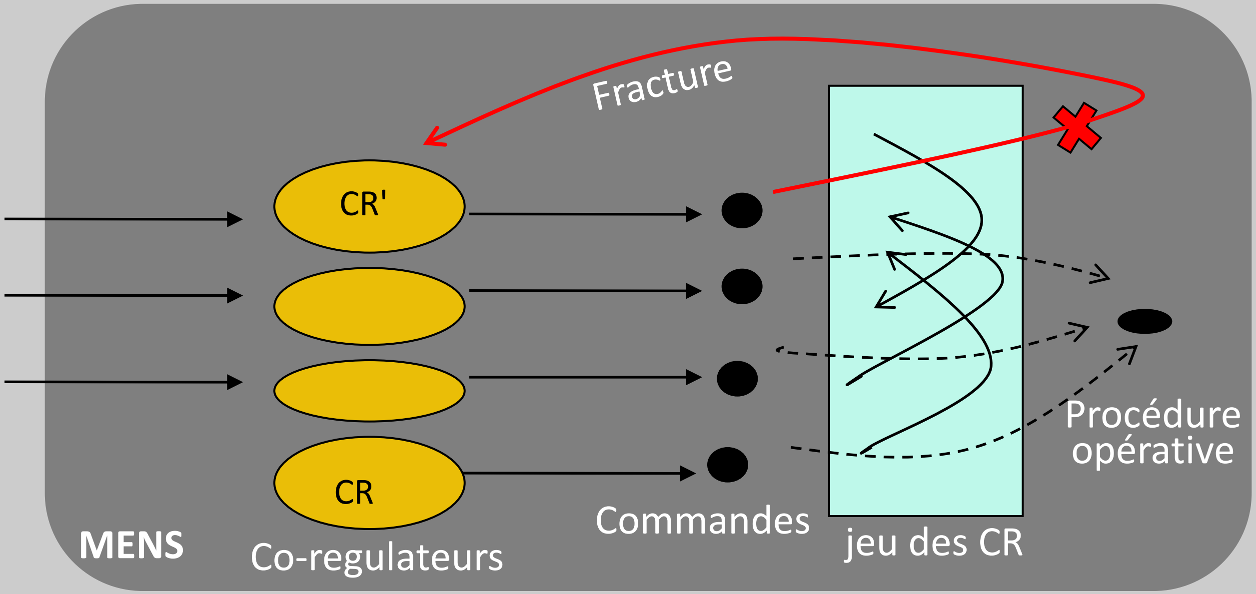
Un lien *g* d'un cat-neurone B vers E est appelé *lien activateur* de E si son composé avec *pr* est une perspective *pb* de B pour CR et si *pb* et *pr* sont simultanément actifs, de sorte que la force de *g* augmente (Règle de Hebb).

LA DYNAMIQUE MULTI-ECHELLES DE MENS



A chaque étape CR forme son paysage, choisit un procept admissible E, active des effecteurs OE => processus dynamique s'étendant jusqu'à la fin de l'étape. Le résultat est évalué et mémorisé à l'étape suivante.

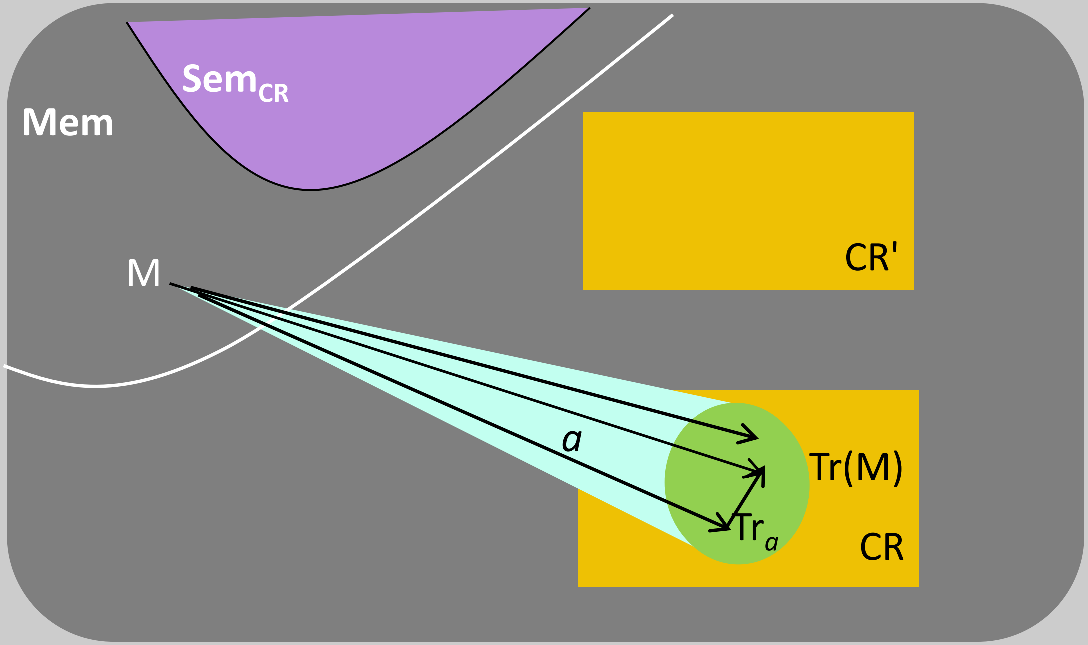
LA DYNAMIQUE MULTI-ECHELLES DE MENS



A chaque étape CR forme son paysage, choisit un procept admissible E, active des effecteurs OE => processus dynamique s'étendant jusqu'à la fin de l'étape. Le résultat est évalué et mémorisé à l'étape suivante.

Les commandes envoyées aux effecteurs par les différents co-régulateurs en *t* peuvent ne pas être cohérentes. D'où un *jeu entre co-regulateurs* rend flexible par la possibilité de balancements entre (ramifications des) effecteurs. La procédure opératoire globale en résulte ; elle peut causer des 'fractures' à certains co-régulateurs.

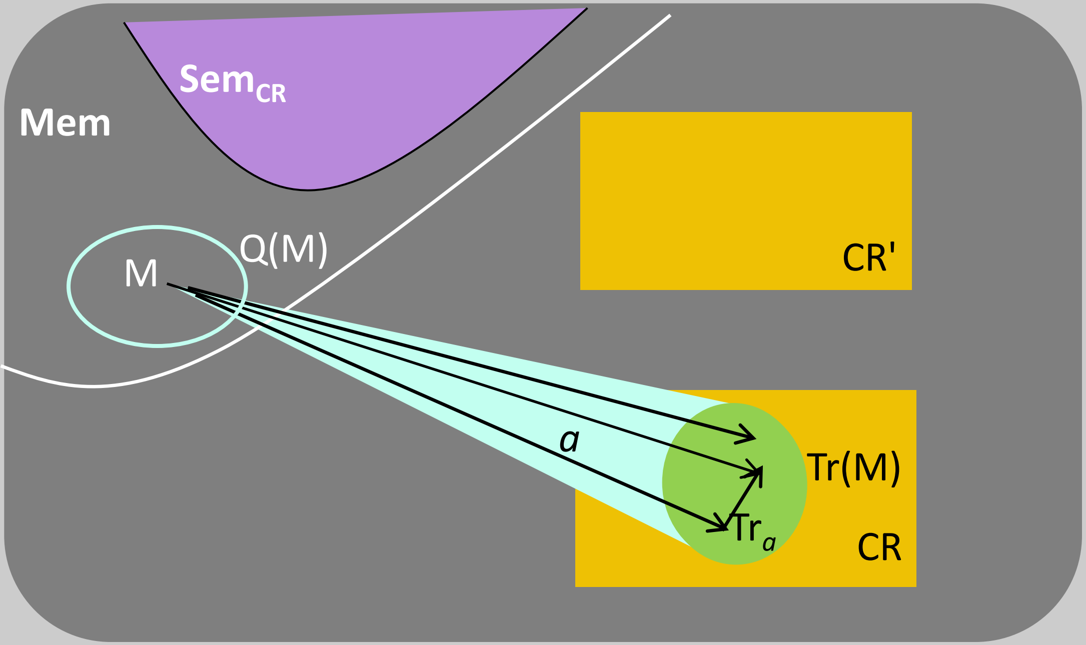
MEMOIRE SEMANTIQUE. I: CLASSES DE CR-SIMILARITE



Sem est un sous-SE de **Mem** développé en 3 étapes : CR-similarité (sous-conceptuel), CR-concept, concepts généraux (symbolique).

Pour un CR et un cat-neurone M ϵ **Mem**, la CR-*trace* de M est le pattern Tr(M) de CR activé par M en *t* (s'il existe): il est indexé par les aspects *a*: M → Tr$_a$ de B pour CR, et ses liens distingués les corrèlent.

MEMOIRE SEMANTIQUE. I: CLASSES DE CR-SIMILARITE

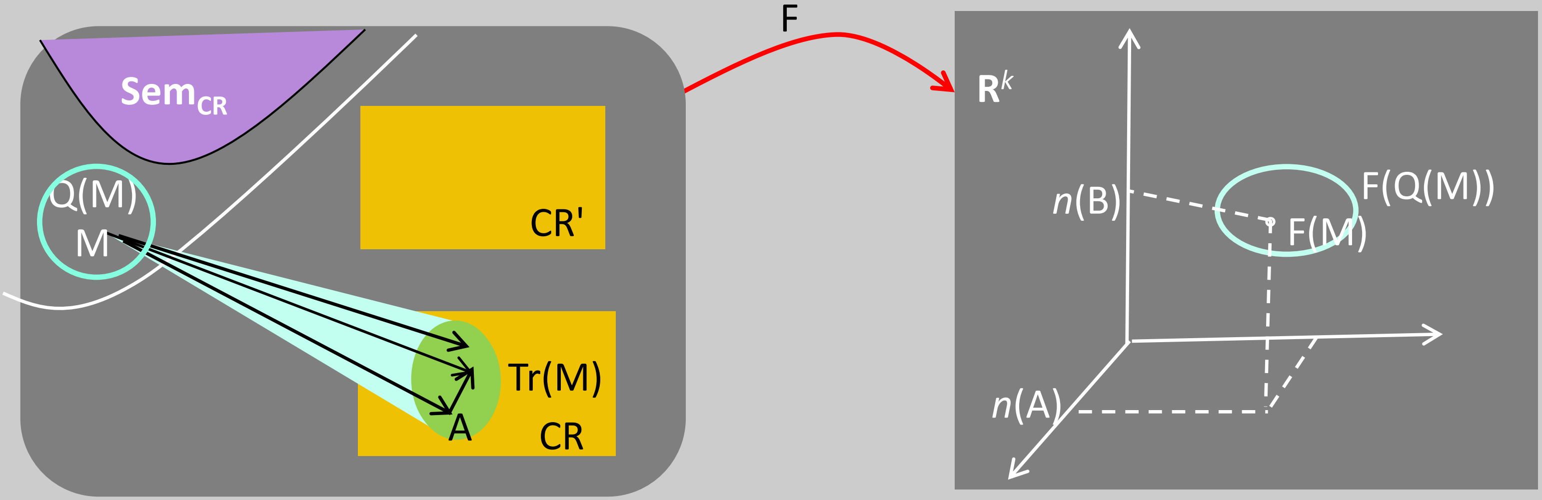


Sem est un sous-SE de **Mem** développé en 3 étapes : CR-similarité (sous-conceptuel), CR-concept, concepts généraux (symbolique).

Pour un CR et un cat-neurone M ϵ **Mem**, la CR-*trace* de M est le pattern Tr(M) de CR activé par M en *t* (s'il existe): il est indexé par les aspects *a*: M → Tr_a de B pour CR, et ses liens distingués les corrèlent.

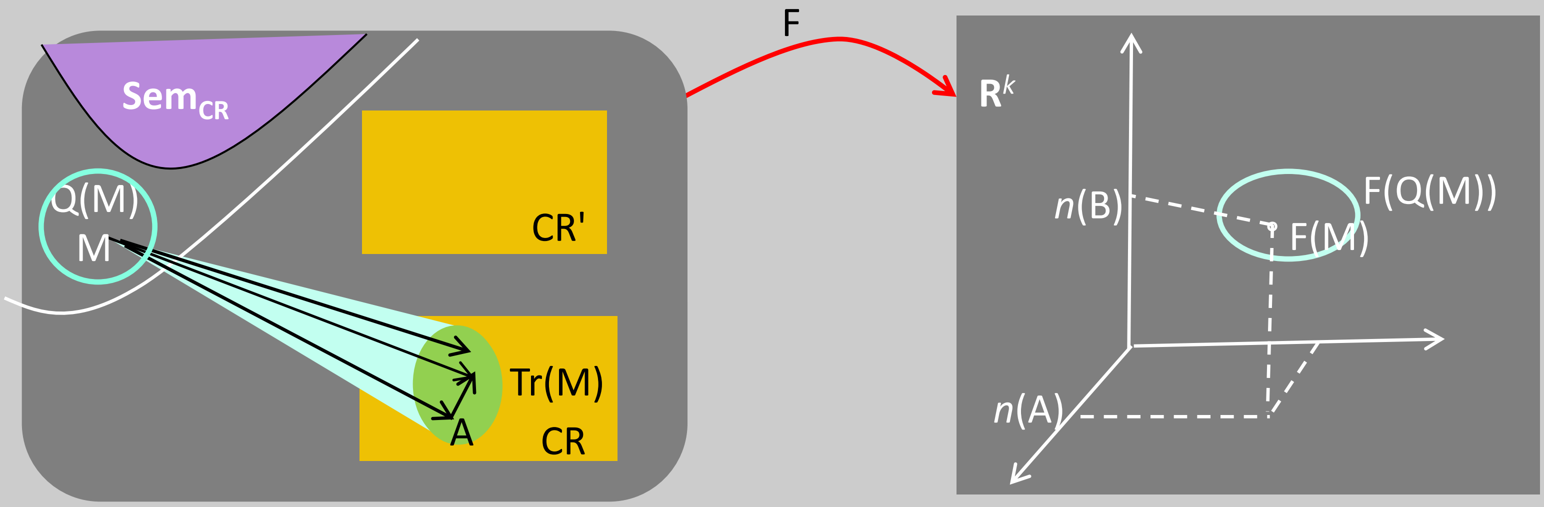
Des M ayant des CR-traces pro-homologues sont dits CR-*similaires.* Leur ensemble Q(M) est la *classe de* CR-*similarité* de M. La CR-similarité n'est reconnue qu'au niveau de co-régulateurs de niveau supérieur au CR.

COMPARAISON AVEC GÄRDENFORS



G. définit un 'espace conceptuel' géométrique D associé à des "quality dimensions" où comparer 2 objets. Un *concept naturel* est une région convexe de D.

COMPARAISON AVEC GÄRDENFORS

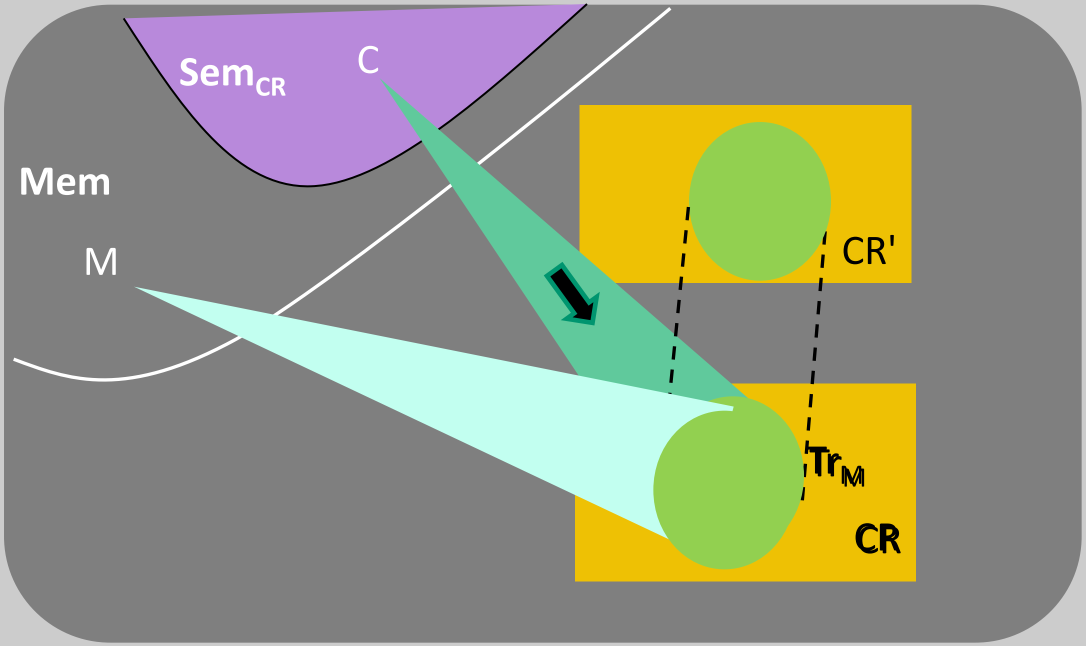


G. définit un 'espace conceptuel' géométrique D associé à des "quality dimensions" où comparer 2 objets. Un *concept naturel* est une région convexe de D.

A la 'qualité' relative à un CR, on associe un tel espace (indépendant du temps) : Si k est la nombre d'agents de CR, on définit une fonction partielle (non injective) F de **Mem** dans $\mathbf{R}^k$ associant à M le vecteur F(M) ayant pour coordonnées les activités n(A) des composants A de CR lorsque Tr(M) est activée. L'image par F de Q(M) est un"concept naturel".

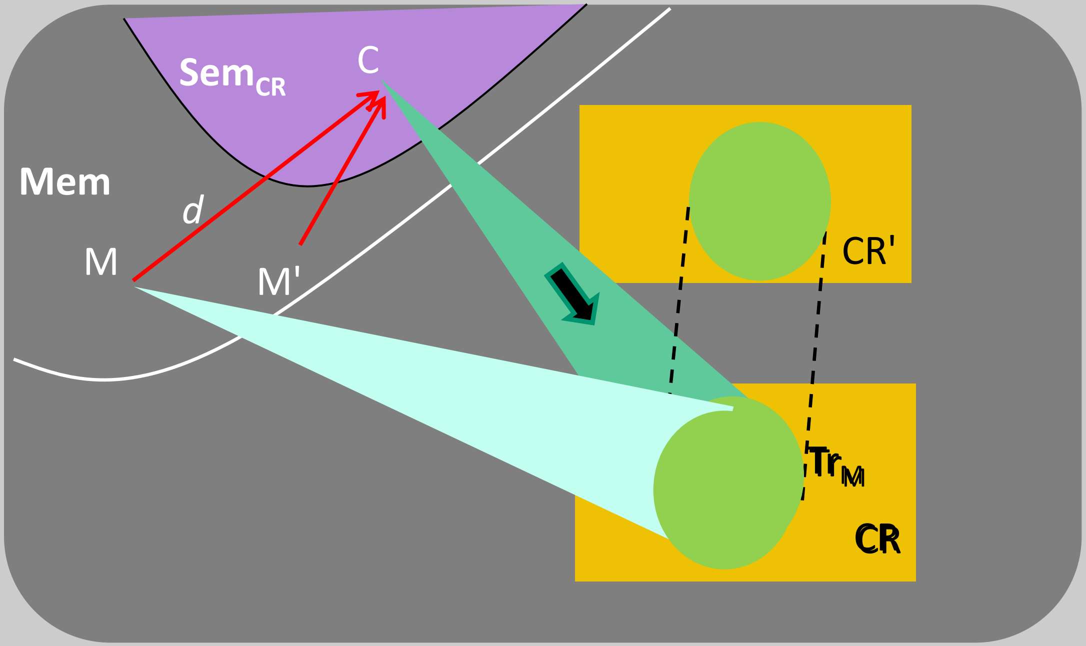
G. n'associe pas de 'symbole' (nos CR-concepts), mais sa notion de 'quality dimension' est plus générale que celle de CR.

(CR-)CONCEPTS BASIQUES



CR-***concept*** de M = limite projective C de sa trace Tr$_{CR}$(M). Les CR-concepts et leurs liens forment un sous-système évolutif **Sem**$_{CR}$ de **Mem** obtenu par complexifications projectives (via des co-régulateurs supérieurs CR').

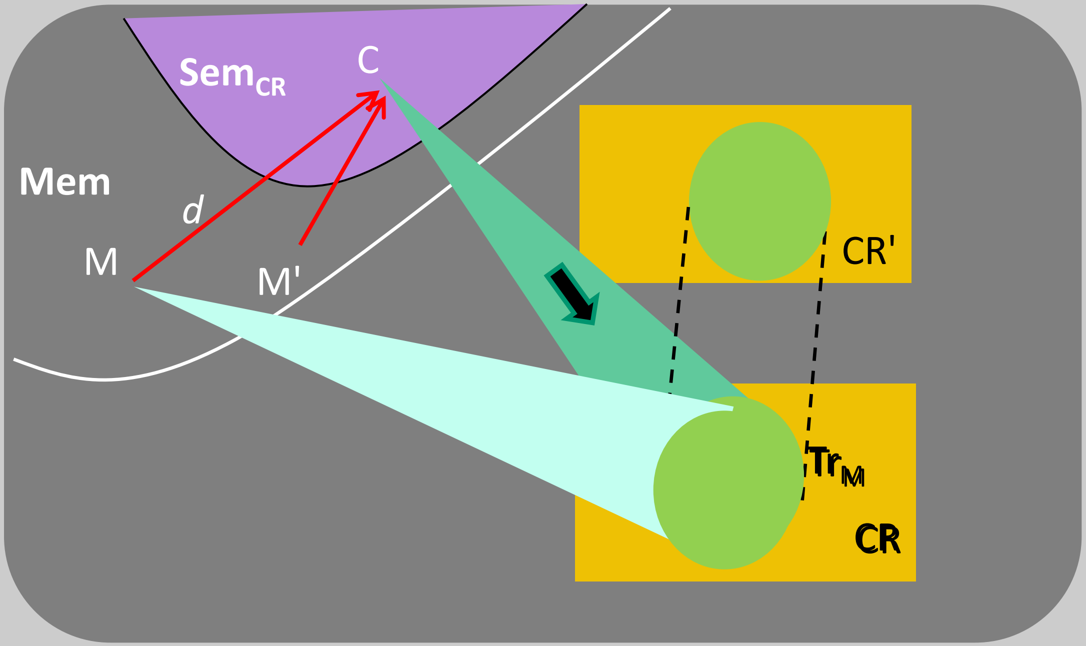
(CR-)CONCEPTS BASIQUES



CR-***concept*** de M = limite projective C de sa trace Tr_{CR}(M). Les CR-concepts et leurs liens forment un sous-système évolutif **Sem**$_{CR}$ de **Mem** obtenu par complexifications projectives (via des co-régulateurs supérieurs CR').

Le lien *d*: M →C définit M comme réflexion de C dans **Sem**$_{CR}$
Instance du CR-concept C = tout M' ayant C pour réflexion dans **Sem**$_{\mathbf{CR}}$**.** La force des liens entre concepts mesure leur distance. C joue le rôle d'un *prototype* (Rosch) ; la force de *d*: M → C mesure l'écart de similarité avec C.

(CR-)CONCEPTS BASIQUES



CR-***concept*** de M = limite projective C de sa trace Tr_{CR}(M). Les CR-concepts et leurs liens forment un sous-système évolutif $\mathbf{Sem}_{CR}$ de **Mem** obtenu par complexifications projectives (via des co-régulateurs supérieurs CR').

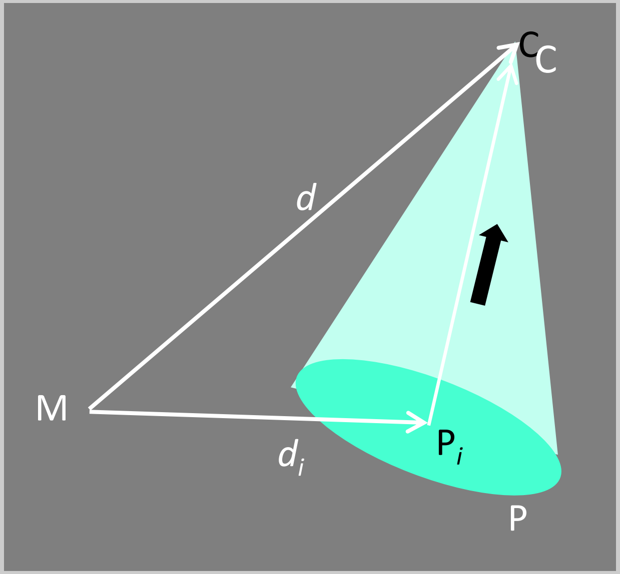
Le lien *d*: M →C définit M comme réflexion de C dans $\mathbf{Sem}_{CR}$

Instance du CR-concept C = tout M' ayant C pour réflexion dans $\mathbf{Sem}_{CR}$**.** La force des liens entre concepts mesure leur distance. C joue le rôle d'un *prototype* (Rosch) ; la force de *d*: M → C mesure l'écart de similarité avec C.

Au cours du temps C prend sa propre identité ; ses instances peuvent varier

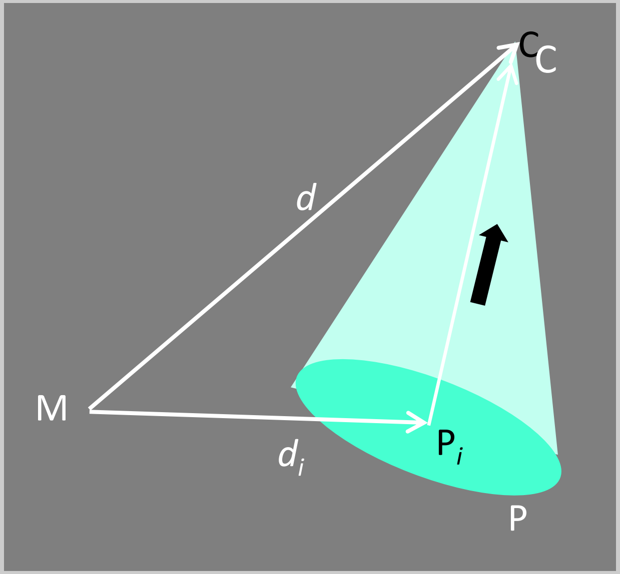
.

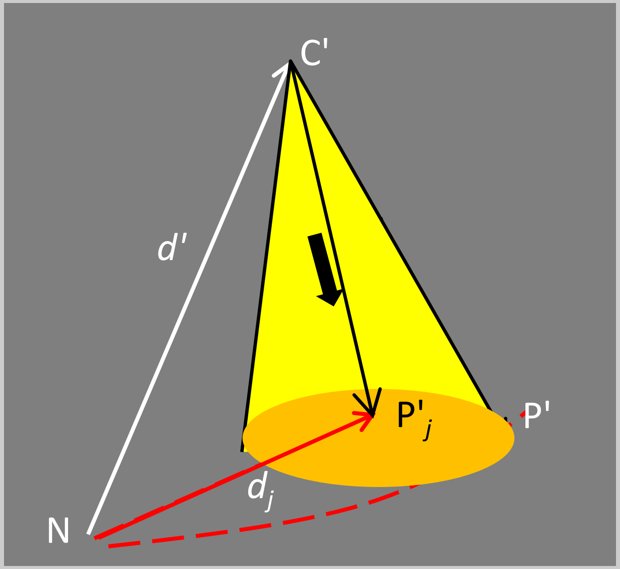
CONCEPTS GENERAUX



Sem est construite par complexifications mixtes de l'union des $\mathbf{Sem}_{CR}$.
Si C = colimP, où P est un pattern de concepts basiques, ses instances M sont toutes les instances des divers P_j.

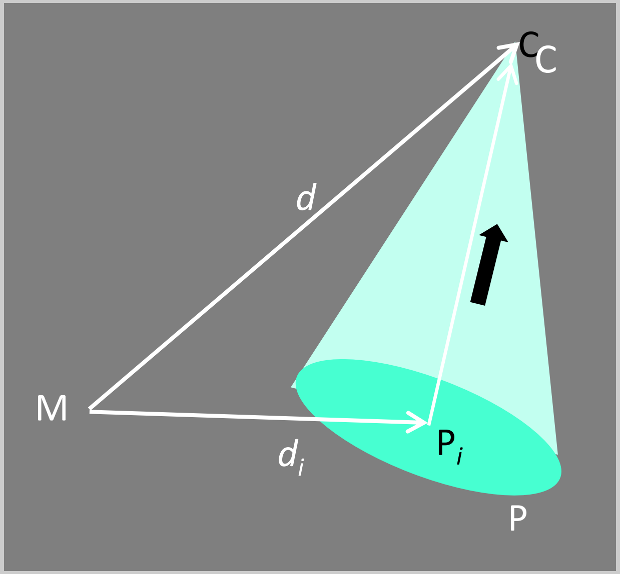
CONCEPTS GENERAUX

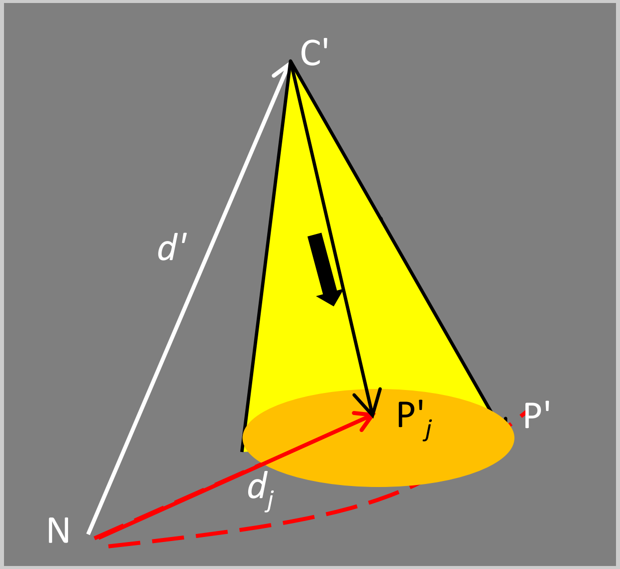




Sem est construite par complexifications mixtes de l'union des $\mathbf{Sem}_{CR}$.
Si C = colimP, où P est un pattern de concepts basiques, ses instances M sont toutes les instances des divers $P_{j.}$
Si C' = limP' une instance N de C' est une instance de chaque P'_j dont les liens canoniques d_j vers P'_j forment un lien distribué vers P', classifié par le lien canonique *d '* de N vers C'.

CONCEPTS GENERAUX





Sem est construite par complexifications mixtes de l'union des $\mathbf{Sem}_{CR}$.
Si C = colimP, où P est un pattern de concepts basiques, ses instances M sont toutes les instances des divers P_j.
Si C' = limP' une instance N de C' est une instance de chaque P'_j dont les liens canoniques d_j vers P'_j forment un lien distribué vers P', classifié par le lien canonique *d '* de N vers C'.

Un *attribut* d'un concept C est un lien dans **Sem** de C vers un CR-concept.
Les concepts prennent leur identité propre au cours du temps ; leur activation se fait via une de leurs instances, avec possibilité de va-et-vient entre instances.

COMPARAISON

Comparaison avec la « *Formal Concept Analysis* » (Wille & Ganter) :

C*ontexte formel* K := (G, M, I), où G est un ensemble d'objets, M un ensemble d'attributs, I une relation *g* I *m* si *m* est un attribut de *g*. Un *concept formel* est un couple C = (A, B) où A et B sont des parties de G et M resp. telles que :

A = B^I = {objets ayant tous les éléments de B pour attributs} et B = A^I.

Alors A = *extension* de C et B = *intension* de C.

COMPARAISON

Comparaison avec la « *Formal Concept Analysis* » (Wille & Ganter) :

C*ontexte formel* K := (G, M, I), où G est un ensemble d'objets, M un ensemble d'attributs, I une relation *g* I *m* si *m* est un attribut de *g*. Un *concept formel* est un couple C = (A, B) où A et B sont des parties de G et M resp. telles que :

A = B^I = {objets ayant tous les éléments de B pour attributs} et B = A^I.

Alors A = *extension* de C et B = *intension* de C.

Ici G = |**Sem|**, M = {CR-concepts], et I est défini par les flèches d'un concept C vers un CR-concept dans **Sem**. Nos concepts deviennent des concepts formels dans ce cadre, l'*intension* de C étant formée de ses attributs et son *extension* étant l'ensemble de ses instances. Mais on y perd la dimension temporelle de **Sem.**

Ceci précise l'idée que, dans une catégotie, on dit que l 'extension (resp. intension) de C est l 'ensemble des flèches de but (resp. source) C.

POUR PLUS D'INFORMATIONS

1. *Memory Evolutive Systems: Hierarchy, Emergence, Cogni-ion,* Elsevier, 2007.

2. MENS, a mathematical model for cognitive *systems,Journal of Mind Theory* 0-2, 2009

3. Les sites internet suivants contiennent nombre de nos articles :

http://ehres.pagesperso-orange.fr

http://vbm-ehr.pagesperso-orange.fr

MERCI