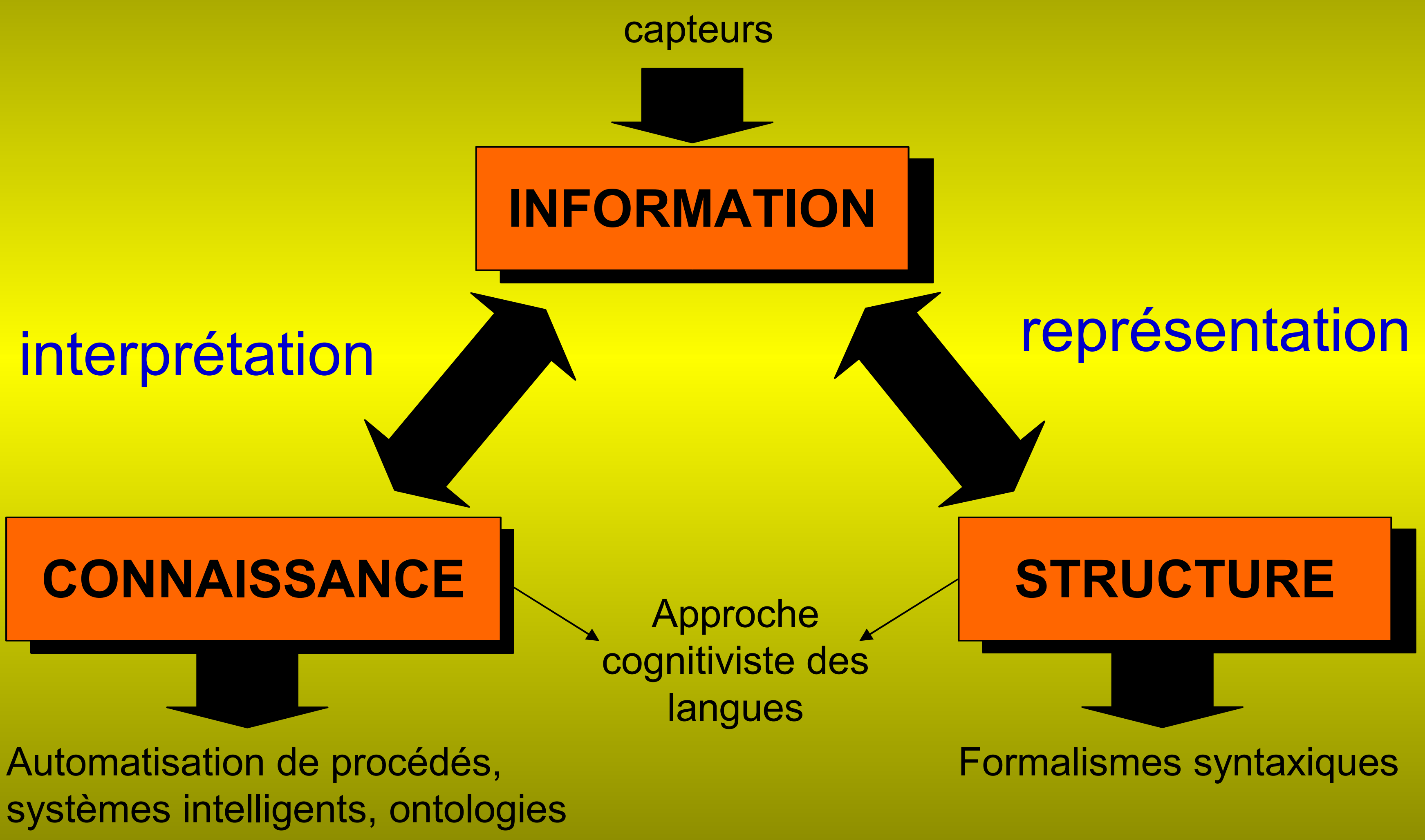
L'Intelligence Artificielle distribué appliquée aux processus physiques

LISTIC – Université de Savoie

LABORATOIRE D'INFORMATIQUE, SYSTÈMES, TRAITEMENT DE L'INFORMATION ET DE LA CONNAISSANCE

Le système d’informations



Problématique

- Proposer une représentation des connais-sances centrée sur la notion de service et permettant :
  - Le contrôle des processus physiques (syst. industriels)
  - L'aide à la conception de systèmes de mesure ou de contrôle
  - L'automatisation du processus de composition de services en milieu distribué.

Notion de service

- Notion de service : centrale dans la conception de systèmes distribues (Web Services, Grid Services).
- modèle classique client/serveur limité à exécuter une tâche simple pour un client donné $\Rightarrow$ insuffisant face à la multiplicité et diversité des ressources (Web Services).
- services dynamiques: capables de s'adapter et de se composer pour en fournir d'autres.
- offre classique de services (client/serveur) $\Rightarrow$ rem-placée par la génération dynamique de service.

Notion de service

- La connaissance locale est constituée d’une connaissance initiale à laquelle vient s’ajouter la connaissance distante au fur et à mesure des échanges.
- Construction et génération dynamiques des services.
- Basé sur la construction d’un canal d’information (IF channel).

La représentation des connaissances

- Approche téléologique : modélisation par une hiérarchie (causale) de buts.
- Environnement : systèmes de mesure ou de contrôle distribués.
- Nécessité de définir les buts et leurs attributs reliés au contexte physique local.

Intérêt de l’approche

- Problème principal des approches orientées but : la faible expressivité de la représentation des buts.
- **Une solution** : proposer une structure formelle qui permet des manipulations algébriques et qui s’intègre dans un modèle ontologique.
- Avantages :
  - repose sur une base mathématique valide
  - utilise des approches formelles éprouvées (FCA, IF)
  - définit un cadre ontologique qui repose sur des capacités de partage et de hiérarchisation.

Structure des buts

3 concepts: $\left\{ \begin{array}{l} \textbf{Contexte physique}, \\ \textbf{Type de buts} \text{ (universal)} \\ \textbf{Instance de buts} \text{ (particular)} \end{array} \right.$

- **Contexte physique : un tuple c**

R : ensemble fini de rôles physiques

Φ : ensemble fini d'entités physiques (instances)

$r \in R$,

$\mu(r) : R \rightarrow Nat$, arité

$$\boxed{\mathbf{c = (r, \mu(r), \varphi_1, \varphi_2, \ldots \varphi_{\mu(r)})}}$$

$\{\varphi_1, \varphi_2, \ldots \varphi_{\mu(r)}\} \subseteq \Phi$: ensemble d'entités physiques reliées à chaque rôle

- **type de but γ, la paire (a, χ)**

$$\boxed{\gamma = (\, a, \chi \,)}$$

a : un symbole d'action (singleton)

Ψ : ensemble fini de types d'entités physiques

$a \in A$,

χ : ensemble non vide de tuples $(r, \mu(r), \psi_1, \psi_2, \ldots \psi_{\mu(r)})$

$\{\psi_1, \psi_2, \ldots \psi_{\mu(r)}\} \subseteq \Psi$: ensemble de types d'entités physiques reliées à chaque rôle

• **instance de but g, la paire (a, C)**

$$\boxed{g = (a, C)}$$

$a \in A,$

A : un symbole d'action (singleton)

Φ : ensemble fini d'instances d'entités physiques

C : ensemble non vide de tuples $(r, \mu(r), \varphi_1, \varphi_2, \ldots \varphi_{\mu(r)})$

$\{\varphi_1, \varphi_2, \ldots \varphi_{\mu(r)}\} \subseteq \Phi$: ensemble des instances d'entités physiques

Exemples

Rôle physique (quantité physique)

Verbe d'action

arité=1, nécessite un seul type d'entité

γ_1 = (to_acquire, {(pressure, 1, liquid_volume)})

g_1 = (to_acquire, {(pressure, 1, SFArea1)})

Type d'entité physique

instance d'entité physique

Information Flow (IF)

- théorie formalisant les échanges d'information entre systèmes complexes et basé sur deux entités fondamentales : les classifications et les info-morphismes.
- Les *classifications* sont la donnée de deux ensem-bles, les *objets* (*tokens*) qui seront les éléments étudiés et les *types* qui seront utilisés pour représenter l'information disponible sur les objets.
- notion d'information formalisée par une relation binaire entre les 2 ensembles ("est de type").
- Les autres entités de base de la théories sont les infomorphismes, paire contravariante de fonctions entre classifications.

Information Flow (IF)

- les classifications et les infomorphismes forment une catégorie.
- A partir de ce formalisme, on peut caractériser des logiques à partir du comportement des objets et des types correspondants.
- par le biais des infomorphismes, l'étude est étendue à des systèmes complexes composés de la mise en relation de plusieurs sous-parties (composition de classifications).

La théorie des flots d'information (IF)

- Objectif : définir et situer l'information dans les systèmes de toute nature.
- Observation : les flots d'information ne sont possibles que dans un système distribué et interconnecté.
- Moyens : des bases mathématiques rigoureu-ses et une approche philosophiquement con-sistante.

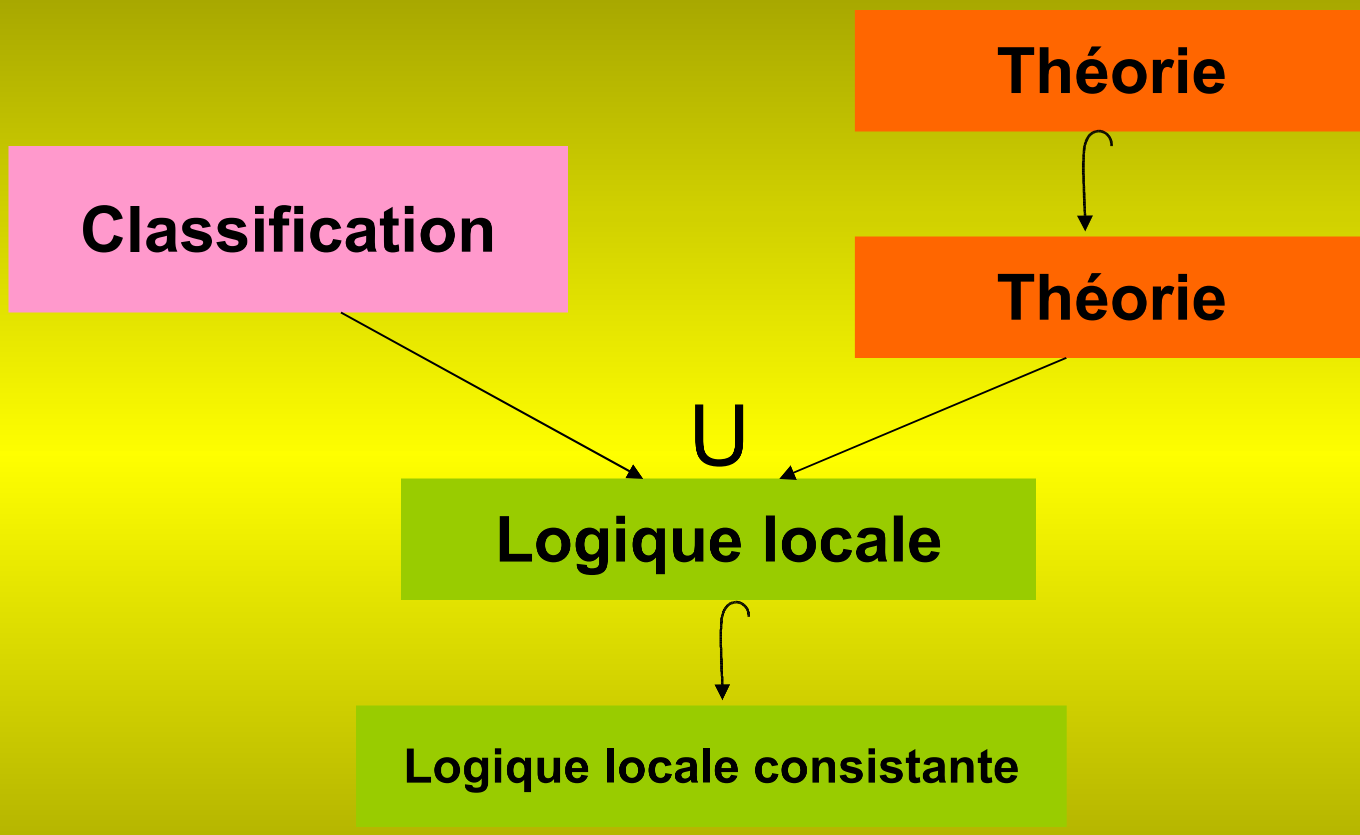
Les Classifications et les Infomorphismes

- Remarques préliminaires :

  La notion d'information n'est intéressante, que si on peut la **classifier.**
- Problème :

  Pour que la classification ait un sens dans de multiples contextes, sa définition doit être indépendante du type de classification

Formalisme IF



Idée de base (Kent 00, Schorl04):

les ontologies sont formalisables par des logiques locales.

Les Classifications

- Définition [Bar97] :

  Une classification $C = \langle tok(c), typ(c), \models_c \rangle$ consiste en :

  - Une classe d'objets à classifier appelés tokens de C (particulars ou instances)
  - Une classe d'objets permettant de classifier les tokens, appelée types de C
  - Une relation binaire $\models_c$ entre tok(c) et typ(c)

  Si $c \models_c \gamma$, alors c est de type γ dans C

La classification

- Représentation :

$$\begin{tikzcd} \mathtt{typ(c)} \arrow[d, dash, "\models_c"] \\ \mathtt{tok(c)} \end{tikzcd}$$

Remarque : les classifications ont reçu des appellations différentes en fonction du contexte d’utilisation. On parle de contexte formel en FCA ou de Chu space en génie logiciel.

La classification

- Exemples :
  - Classification des mots du dictionnaire selon les types {NOM, VERBE INT,VERBE TRANS., ADJECTIF}
  - Classification des langages du 1er ordre : token = stuctures maths M (L-structures), types = formules ϕ du langage, M $\models_c \phi$ ssi ϕ est vraie dans M. L'ensemble des types associés à un M est l'ensemble des formules de L vraies dans M $\rightarrow$ théorie de M.

Les infomorphismes

- Problème majeur : un type de classification n'est pas unique. La question est, comment relier ces différents types?
- Intérêt : les systèmes distribués
- On se donne deux classifications A et B où les objets peuvent être différents (eg, modélisation des composants d'un syst. distribué) ou les mêmes (différentes vues ou différents langages pour un même objet).

Les infomorphismes

Définition : un infomorphisme $f: A \rightleftarrows B$, consiste en une paire de classifications $<A,B>$ et une paire de fonctions contravariante $f = <f^{sup}, f^{inf}>$ satisfaisant la propriété :

$$f^{inf}(b) \models_A \alpha \quad \text{ssi} \quad b \models_B f^{sup}(\alpha)$$

Fournit un mécanisme pour relier des systèmes d'information ayant des structures semblables

Les infomorphismes

$$\begin{tikzcd}
\mathtt{typ(A)} \arrow[r, "f^{sup}"] \arrow[d, dash, "\models_A"] & \mathtt{typ(B)} \arrow[d, dash, "\models_B"] \\
\mathtt{tok(A)} & \mathtt{tok(B)} \arrow[l, "f^{inf}"]
\end{tikzcd}$$

Représentation d’un infomorphisme

Exemples d’infomorphismes

- Ontologie : les classes sont les types et les instances ou sous-classes, les tokens avec la relation de classification “instanceOf”
- ***P***(A) ensemble des parties d’un ensemble A, instances : les éléments de A,

  types : les sous-ensembles de A et avec la relation d’appartenance comme relation de classification.

La théorie IF

- Si une classification modélise un composant, alors les types de la classification représentent toutes les propriétés pertinentes du composant pour le modèle.
- Si les tokens représentent toutes les instances possibles, alors la classification définit une **théorie** du composant (i.e. les relations entre types qui sont vérifiées par tous les tokens).

La théorie IF (suite)

- Séquents (Gentzen) (Γ, Δ) composés d'en-sembles de types.
- Formulation : $\Gamma \vdash \Delta$
- Définition : Soit A une classification et (Γ, Δ) un séquent de A. Un token a de A satisfait le séquent si a est de certains types de Δ cha-que fois que a est de tous les types de Γ.

La théorie IF (suite)

- Si chaque token a est contraint par (Γ, Δ) alors on a $\Gamma \vdash_A \Delta$ et Th(A)=<typ(A), $\vdash_A$> est la théorie générée par A.
- Les contraintes représentent les régularités du système (partie statique).

entrainement	$\alpha \vdash \beta$	α entraîne β
nécessité	$\vdash \alpha$	Nécessairement du type α
exhaustivité	$\vdash \alpha, \beta$	Toute instance est de l'un des 2 types
incohérence	$\alpha \vdash$	Aucune instance est de type α
incompatibilité	$\alpha, \beta \vdash$	Aucune instance n'est simultanément des 2 types

Séquents particuliers

La théorie IF (suite)

- Exemples
  - Les types a et b sont une partition du type g si
    - α |- γ et β |- γ (sous-types)
    - γ |- α, β (recouvrement)
    - α, β |- (disjonction)
  - Notation :

$$\forall\Gamma \rightarrow \exists\Delta$$

La théorie régulière

- Une théorie T est dite régulière si pour tout $\alpha \in typ(T)$ et pour des sous-ensembles arbitraires Γ, Δ, Γ', Δ', $\Sigma' \subseteq typ(T)$, les propriétés suivantes sont vérifiées :
  - Identité $\alpha \mid\!-_T \alpha$
  - Affaiblissement si $\Gamma \mid\!-_T \Delta$ alors $\Gamma, \Gamma' \mid\!-_T \Delta, \Delta'$
  - Coupure globale si $\Gamma, \Gamma' \mid\!-_T \Delta, \Delta'$ pour chaque partition (Γ', Δ') de Σ', alors $\Gamma \mid\!-_T \Delta$

La logique locale

- *Une logique locale* $L = \langle tok(L), typ(L), \models_L, \vdash_L, N_L \rangle$ *consiste en une théorie régulière* $th(L) = \langle typ(L), \vdash_L \rangle$, *une classification* $cla(L) = \langle tok(L), typ(L), \models_L \rangle$ *et un sous-ensemble* $N_L \in tok(L)$ de tokens *normaux satisfaisant toutes les contraintes de* $th(L)$.
- *Une logique L est valide si* $N_L = tok(L)$

Les infomorphismes logiques

- Une fois que les structures locales sont définies, elles doivent être reliées de manière à permettre à l'information de s'écouler entre les composants du système distribué.
- Notion d'infomorphisme logique

Les infomorphismes logiques

- Définition :

Soient deux logiques consistantes L =< tok(L), typ(L), $|=_L$, $|-_L$> et L' =<tok(L'), typ(L'), $|=_{L'}$, $|-_{L'}$>, un infomorphisme logique L $\rightleftarrows$ L' consiste en une paire contravariante de fonctions f =< f^{sup}, f^{inf}> avec f^{sup} : *typ*(*L*) $\rightarrow$ *typ*(*L'*) et f^{inf} : *tok*(*L'*) $\rightarrow$ *tok*(*L*) tel que:

1. *f est l'infomorphisme des classications f* : *cla*(*L*) $\rightleftarrows$ *cla* (*L'*)
2. *pour tous les* (Γ, Δ) $\subseteq$ *th*(*L*), Γ $|-_L$ Δ *est une contrainte de th*(*L*) si $f^{sup}[\Gamma]$ $|-_{L'}$ $f^{sup}[\Delta]$ est une *contrainte de th*(*L'*).

La propagation des logiques

- Proposition :

  Soit L une logique locale construite sur la classification C et soit f : A $\rightleftarrows$ C un infomorphisme.

1. *Si L est complète, alors son image inverse sous f notée $f^{-1}[L]$ est complète.*
2. *Si f est token surjective et que L est valide, alors $f^{-1}[L]$ est valide.*

La notion de canal (IF)

- *Définition : Un canal IF consiste en deux classifications A_1 et A_2 connectées par une classification centrale C au moyen de deux infomorphismes f_1 et f_2.*

$$
\begin{tikzcd}[column sep=huge, row sep=huge]
\mathtt{typ}(A_1) \arrow[r, "f_1^{sup}"'] \arrow[d, dash, "\models_{A1}"] & \mathtt{typ}(C) \arrow[d, dash, "\models_{C}"] & \mathtt{typ}(A_2) \arrow[l, "f_2^{sup}"] \arrow[d, dash, "\models_{A2}"] \\
\mathtt{tok}(A1) & \mathtt{tok}(C) \arrow[l, "f_1^{inf}"] \arrow[r, "f_2^{inf}"'] & \mathtt{tok}(A2)
\end{tikzcd}
$$

Le canal central

- Le concept de canal IF modélise le contexte dans lequel la transmission de l'information a lieu.
- Le canal central se comporte comme un médium pour la communication entre A_1 et A_2.
- Il autorise des flots n-aires, mais la plupart des applications n'utilisent que le canal binaire.

Exemple

- Système de représentation géographique
  - typ(A1) : symboles de cartographie (ex.: ligne rouge et jaune=autoroute)
  - tok(A1) : éléments de la carte
  - typ(A2) : types d'entités du monde réel
  - tok(A2) : entités physiques du monde réel
- La contrainte (relation d'infomorphisme) garantit que la carte décrit correctement le monde réel.

Exemple (suite)

- Le canal central contient le processus cartographique (jeu de règles qui assure qu’une ligne rouge indique une route par ex.)

La logiques distribuée

- Puisque les logiques locales sont des concepts inclusifs combinant les concepts des classifications et des théories, elles recouvrent une connaissance plus générale que des simples classifications.
- Il y a donc un intérêt à considérer les logiques IF distribuées via des canaux IF.

Les logiques distribuées

- *Définition*

  Soit un canal binaire C = $\{ f_1 : A_1 \leftrightarrows C, f_2 : A_2 \leftrightarrows C \}$ *muni d'une logique L sur la classification centrale C, la logique distribuée* $DLog_C(L)$ *de C générée par L est telle que:*

$$DLog_C(\mathrm{L}) = F^{-1}[\mathrm{L}]$$

Les logiques distribuées

$$\begin{tikzcd} & C & \\ A_1 \arrow[ur, "f_1"] \arrow[r, "\sigma_1"'] & A_1 + A_2 \arrow[u, "F"] & A_2 \arrow[ul, "f_2"'] \arrow[l, "\sigma_2"] \end{tikzcd}$$

Co-produit

Application aux hiérarchies de buts

- Problème de fond : coordination d’ontologies distribuées.
- Des services distribués sont représentés par des hiérarchies fonctionnelles (ontologies) réparties.
- L’exécution d’un service particulier peut nécessiter l’utilisation d’un service distant $\rightarrow$ on se ramène à trouver la hiérarchie fonctionnelle qui se rattache logiquement à la hiérarchie locale.
- Intérêt : automatisation de la composition des services

Les contraintes de IF

- IF représente les hiérarchies de buts distribuées comme des logiques locales.
- Les ensembles de contraintes modélisent la sémantique relationnelle des hiérarchies.
- Les logiques requises dans le partage des hiérarchies doivent être consistantes.

Principe de base

- IF est (entre autres) utilisé en partages d'onto-logies [Schorl03][Kalfo03][Kent00] pour identifier les types communs entre différentes ontologies.
- Extension de l'étude en identifiant les liaisons entre buts appartenant à des services distants utilisant la même structure formelle.
- En onto mapping, $\vdash$ représente la subsumption de concepts, alors que dans cette approche elle représente la dépendance fonctionnelle.

Les dépendances entre les buts

Dépendances fonctionnelles dans le style λ-calcul :

Un type de but γ_i influence fonctionnellement γ_j si la seule façon de réaliser γ_j est d'avoir d'abord réalisé γ_i avec la notation:

$$\boxed{\gamma_i < \gamma_j}$$

(basé sur une approche similaire en Software Engineering [Bal99])

Les contextes formels

Γ_1

	γ_1	γ_5	γ_6	γ_7
γ_1	1	1	1	1
γ_5	0	1	0	1
γ_6	0	0	1	1
γ_7	0	0	0	1

Γ_2

	γ_1	γ_5	γ_6	γ_8	γ_9	γ_3	γ_2
γ_1	1	1	1	0	1	1	1
γ_5	0	1	0	0	1	1	1
γ_6	0	0	1	0	1	1	1
γ_8	0	0	0	1	1	1	1
γ_9	0	0	0	0	1	1	1
γ_3	0	0	0	0	0	1	1
γ_2	0	0	0	0	0	0	1

Γ_3

	γ_4	γ_2
γ_4	1	1
γ_2	0	1

La construction des hiérarchies

Contexte formel

Γ_1

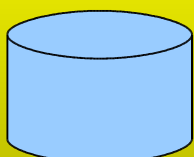
	γ_1	γ_5	γ_6	γ_7
γ_1	1	1	1	1
γ_5	0	1	0	1
γ_6	0	0	1	1
γ_7	0	0	0	1

Γ_2

	γ_1	γ_5	γ_6	γ_8	γ_9	γ_3	γ_2
γ_1	1	1	1	0	1	1	1
γ_5	0	1	0	0	1	1	1
γ_6	0	0	1	0	1	1	1
γ_8	0	0	0	1	1	1	1
γ_9	0	0	0	0	1	1	1
γ_3	0	0	0	0	0	1	1
γ_2	0	0	0	0	0	0	1

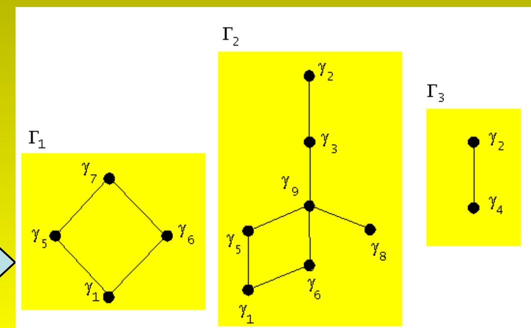
Γ_3

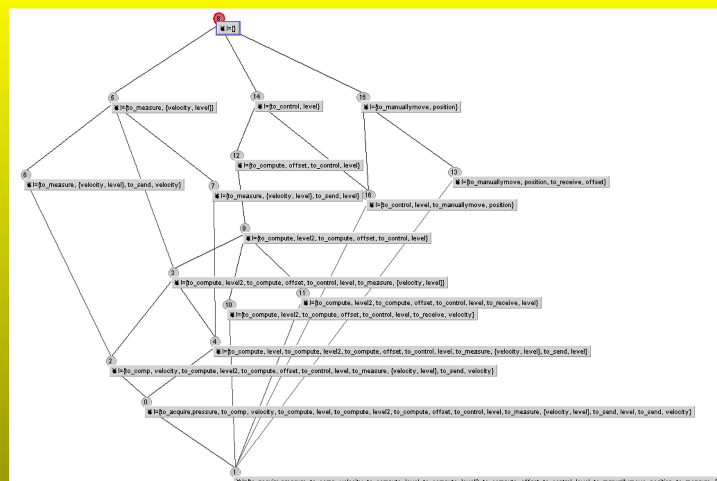
	γ_4	γ_2
γ_4	1	1
γ_2	0	1



KB

Treillis épuré





Treillis de Galois des types de buts

Ontologie des buts

$\gamma_1 \sqsubseteq \gamma_5$	$\gamma_1 \sqsubseteq \gamma_5$	$\gamma_8 \sqsubseteq \gamma_9$	$\gamma_4 \sqsubseteq \gamma_2$
$\gamma_1 \sqsubseteq \gamma_6$	$\gamma_1 \sqsubseteq \gamma_6$	$\gamma_9 \sqsubseteq \gamma_3$	
$\gamma_5 \sqsubseteq \gamma_7$	$\gamma_5 \sqsubseteq \gamma_9$	$\gamma_3 \sqsubseteq \gamma_2$	
$\gamma_6 \sqsubseteq \gamma_7$	$\gamma_6 \sqsubseteq \gamma_9$		
$\gamma_5 \, O \, \gamma_6$	$\gamma_5 \, O \, \gamma_6$		
$\gamma_7 = \gamma_5 + \gamma_6$	$\gamma_9 = \gamma_5 + \gamma_6 + \gamma_8$		

Les considérations ontologiques

Objectif double :

- fournir un modèle de connaissances pour un processus physique du monde réel.
- disposer d'un signifiant partagé à travers un système distribué.

L'ontologie inclut 3 relations de base :

- la relation partie fonctionnelle de : $\sqsubseteq_I$
- la relation de recouvrement entre types de buts : O
- la somme (fusion) de buts : +

Les relations ontologiques

$\gamma_1 \sqsubseteq \gamma_5$

$\gamma_1 \sqsubseteq \gamma_6$

$\gamma_5 \sqsubseteq \gamma_7$

$\gamma_6 \sqsubseteq \gamma_7$

$\gamma_5 \, \mathrm{O} \, \gamma_6$

$\gamma_7 = \gamma_5 + \gamma_6$

Γ_1

$\gamma_1 \sqsubseteq \gamma_5$

$\gamma_1 \sqsubseteq \gamma_6$

$\gamma_5 \sqsubseteq \gamma_9$

$\gamma_6 \sqsubseteq \gamma_9$

$\gamma_8 \sqsubseteq \gamma_9$

$\gamma_9 \sqsubseteq \gamma_3$

$\gamma_3 \sqsubseteq \gamma_2$

$\gamma_5 \, \mathrm{O} \, \gamma_6$

$\gamma_9 = \gamma_5 + \gamma_6 + \gamma_8$

Γ_2

$\gamma_4 \sqsubseteq \gamma_2$

Γ_3

La formalisation

- *Une hiérarchie fonctionnelle $\mathcal{F}$ partie-de est décrite par le tuple : $\mathcal{F} = (\Gamma, \sqsubseteq_I, O, +)$, où Γ est un ensemble fini de types de buts, $\sqsubseteq_I$ est un ordre partiel sur Γ, O est la relation d'overlap et +, la relation de fusion sur les types de buts.*
- Raisonner sur les buts est basé sur leurs propriétés extensionnelles (leur contexte physique c_i).

- Une contrainte γ_i |- γ_k traduit le fait que le contexte physique de γ_i fait partie intégrante du contexte du but γ_k .
- Le contexte commun de deux buts est le contexte de leur overlap
- Tout contexte de la fusion de deux buts est un contexte de l'un des deux buts
- On peut alors construire une logique locale

La logique locale est définie par le tuple L = (C, *Γ,* |=, |-), *où* (C, *Γ,* |=) *est une classification IF entre types de buts et instances de contextes et* (*Γ*, |-) est *la théorie locale donnée par la plus petite relation de conséquence close par l'identité, l'affaiblissement et la coupure globale, tel que pour tout* γ_i , γ_j , $\gamma_k \in \Gamma$ *on ait :*

1. γ_i |- γ_j *ssi* $\gamma_j \sqsubseteq_{I}^{def} \gamma_i$
2. γ_i, γ_j |- γ_k *ssi* $\gamma_k =_{def} \gamma_i$ *O* γ_j
3. γ_k |- γ_i , γ_j *ssi* $\gamma_k = \gamma_i + \gamma_j$

Le processus IF

- Etant donné des hiérarchies fonctionnelles réparties on peut construire une théorie IF centrale résultant de l’union disjointe des hiérarchies
- La logique associée distribuée sur les classi-fications de chaque composant extrait les sé-quents des services distribués s’ils partagent des contextes physiques.

Le processus IF

- On modélise la recherche de dépendances en utilisant des classifications pour chaque service (F_0, origine et candidat(s) potentiel(s) F_i.
- Pour construire le canal IF $\rightarrow$ extraire de chaque hiérarchie potentielle le contexte physique qui correspond au contexte(s) recherché(s). Cas du contrôle distribué.

contexte recherché :

$c_k = (r, \mu(r), \{\varphi_{\mu(r)}\})$

contexte correspondant :

$c^{(i)}_j = (r, \mu(r), \{\varphi^{(i)}_{\mu(r)}\})$

Le processus IF (suite)

- Formalisation de l'équivalence par une classification M reliant les types (α, β, ...) avec toutes les instances possibles

typ(M)

$\models$

tok(M)

$\models_M$	α	β
a	0	0
b	0	1
c	1	0
d	1	1

Le processus IF (suite)

- On introduit les transposées des classifications F_0 et $F^{(k)}_i$ (service i sur système k) pour raisonner sur les contextes.

$$
\begin{tikzcd}
\mathrm{typ}(F^{\perp}_0) \arrow[d, dash] & \mathrm{typ}(M) \arrow[l, "\zeta_0^{sup}"'] \arrow[r, "\zeta^{(k)\,sup}_i"] \arrow[d, dash] & \mathrm{typ}(F^{(k)\perp}_i) \arrow[d, dash] \\
\mathrm{tok}(F^{\perp}_0) \arrow[r, "\zeta_0^{inf}"] & \mathrm{tok}(M) & \mathrm{tok}(F^{(k)\perp}_i) \arrow[l, "\zeta^{(k)\,inf}_i"']
\end{tikzcd}
$$

Le processus IF (suite)

L’alignement partiel de contextes est réalisé via la classification M où des relations telles que $\zeta_0^{sup}(\alpha) = c_r$ et $\zeta^{(k)}{}_i^{sup}(\alpha) = c^{(k)}{}_q$ signifient que c_r et $c^{(k)}{}_q$ sont des types identiques dans les classifications $F_0{}^\perp$ et $F^{(k)}{}_i{}^\perp$.

Le processus IF (suite)

- On doit satisfaire l'équation de l'infomorphisme, ce qui permet de calculer les tokens de $F_0^{\perp}$ et $F^{(k)}{}_i^{\perp}$.
- On introduit les partitions conjonctives $\wedge F_0^{\perp}$ et $\wedge F^{(k)}{}_i^{\perp}$ qui vont permettre de classifier les types de buts en fonction des conjonctions de contex-tes physiques.

$$\kappa_0 : F_0^{\perp} \rightleftarrows \wedge F_0^{\perp} \text{ et } \kappa^{(k)}{}_i : F^{(k)}{}_i^{\perp} \rightleftarrows \wedge F^{(k)}{}_i^{\perp}$$

Le processus IF (suite)

- L'étape suivante consiste à construire la classification C du canal IF qui traduit la notion de flot d'information en fonction des identifications de contextes.

$$g_0 : \wedge F_0{}^{\perp} \rightleftarrows C \text{ et } g^{(k)}_i : \wedge F^{(k)}_i{}^{\perp} \rightleftarrows C$$

- C est la colimite du processus
- typ(C) représente l'union disjointe des types de $\wedge F_0{}^{\perp}$ et de $\wedge F^{(k)}_i{}^{\perp}$
- tok(C) sont des paires de buts qui relient un jeton de $\wedge F_0{}^{\perp}$ avec un jeton de $\wedge F^{(k)}_i{}^{\perp}$ seulement s'ils sont transmis par les infoms ζ_0 et $\zeta^{(k)}_i$.

Le processus IF (suite)

- On en déduit la logique centrale LogC :

$$\{c^{(k)}_q , c^{(l)}_p , c^{(m)}_r , \ldots \} \mid\!- \{c_s , c_t , \ldots\}$$

La logique distribuée DlogC(L) sur la somme $F_0 + F^{(k)}_i$ est l'image inverse de LogC.

- Interopérabilité traduite par des contraintes du type :

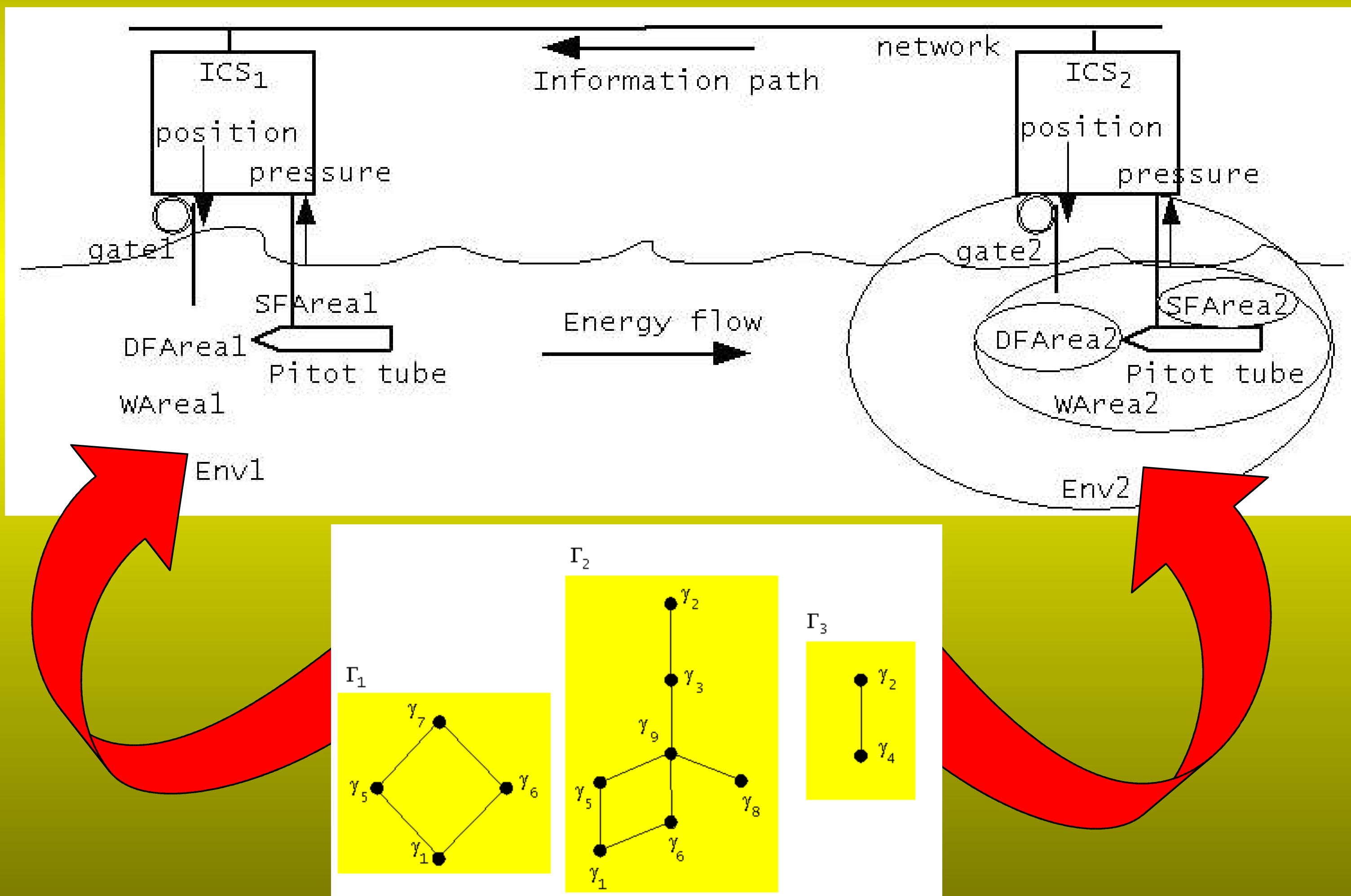
$$\gamma^{(k)}_j , \gamma^{(l)}_h , \ldots \mid\!- \gamma_m \text{ pour la somme } F_0, F^{(k)}_i$$

Overview du processus

$$
\begin{tikzcd}
 & & & & \mathbf{c} & & & \\
F_0 \arrow[r, "f_0"] & \wedge F_0^{\perp} \arrow[urrr, "g_0"] & F^{\perp}_0 \arrow[l, "\kappa_0"'] & & M \arrow[ll, "\zeta_0"'] \arrow[r, "\zeta^{(k)}_i"] & F^{(k)\perp}_i \arrow[r, "\kappa^{(k)}_i"] & \wedge F^{(k)\perp}_i \arrow[ull, "g^{(k)}_i"'] & F^{(k)}_i \arrow[l, "f^{(k)}_i"']
\end{tikzcd}
$$

Un exemple de scénario

Le système de contrôle distribué



γ_1 = (to_acquire, {(pressure, 1, liquid_volume)})

γ_2 = (to_act_upon, {(offset, 1, actuator)})

γ_3 = (to_compute, {(offset, 1, actuator)})

γ_4 = (to_receive, {(offset, 1, actuator)})

γ_5 = (to_compute, {(level, 1, channel_part)})

γ_6 = (to_compute, {(velocity, 1, channel_part)})

γ_7 = (to_send, {(level, 1, channel_part), (velocity, 1, channel_part)})

γ_8 = (to_receive, {(level, 1, channel_part), (velocity, 1, channel_part)})

γ_9 = (to_compute, {(level, 2, {channel_part, channel_part})})

Types de buts

Théories locales

	downstream node		upstream node
Γ_1	$\gamma_2, \gamma_3 \vdash \gamma_1$ $\gamma_4 \vdash \gamma_2, \gamma_3$ $\vdash \gamma_4$	Γ'_1	$\gamma'_2, \gamma'_3 \vdash \gamma'_1$ $\gamma'_4 \vdash \gamma'_2, \gamma'_3$ $\vdash \gamma'_4$
Γ_2	$\gamma_2, \gamma_3 \vdash \gamma_1$ $\gamma_6 \vdash \gamma_2, \gamma_3, \gamma_5$ $\gamma_6 \vdash \gamma_7$ $\gamma_7 \vdash \gamma_9$ $\vdash \gamma_9$	Γ'_2	$\gamma'_2, \gamma'_3 \vdash \gamma'_1$ $\gamma'_6 \vdash \gamma'_2, \gamma'_3, \gamma'_5$ $\gamma'_6 \vdash \gamma'_7$ $\gamma'_7 \vdash \gamma'_9$ $\vdash \gamma'_9$
Γ_3	$\gamma_8 \vdash \gamma_9$ $\vdash \gamma_9$	Γ'_3	$\gamma'_8 \vdash \gamma'_9$ $\vdash \gamma'_9$

Contextes physiques

$$
\begin{array}{lclclcl}
c_1 & = & (pressure, 1, SFA), & c'_1 & = & (pressure, 1, SFA') \\
c_2 & = & (pressure, 1, DFA), & c'_2 & = & (pressure, 1, DFA') \\
c_3 & = & (velocity, 1, FA), & c'_3 & = & (velocity, 1, FA') \\
c_4 & = & (level, 1, FA), & c'_4 & = & (level, 1, FA') \\
c_5 & = & (velocity, 1, ExtEnt), & c'_5 & = & (velocity, 1, ExtEnt) \\
c_6 & = & (level, 1, ExtEnt), & c'_6 & = & (level, 1, ExtEnt) \\
c_7 & = & (level, 2, FA, ExtEnt), & c'_7 & = & (level, 2, FA', ExtEnt) \\
c_8 & = & (offset, 1, gate), & c'_8 & = & (offset, 1, gate') \\
c_9 & = & (position, 1, gate), & c'_9 & = & (position, 1, gate')
\end{array}
$$

	F_1				F_2							F_3	
	γ_1	γ_2	γ_3	γ_4	γ_1	γ_2	γ_3	γ_5	γ_6	γ_7	γ_9	γ_8	γ_9
$(pressure, 1, SFA)$	1	1	1	1	1	1	1	0	1	1	1	0	0
$(pressure, 1, DFA)$	1	1	1	1	1	1	1	0	1	1	1	0	0
$(velocity, 1, FA)$	0	1	0	1	0	1	0	0	1	1	1	0	0
$(level, 1, FA)$	0	0	1	1	0	0	1	0	1	1	1	0	0
$(velocity, 1, ExtEnt)$	0	0	0	0	0	0	0	1	1	1	1	0	0
$(level, 1, ExtEnt)$	0	0	0	0	0	0	0	1	1	1	1	0	0
$(level, 2, FA, ExtEnt)$	0	0	0	0	0	0	0	0	1	1	1	0	0
$(offset, 1, gate)$	0	0	0	0	0	0	0	0	0	1	1	1	1
$(position, 1, gate)$	0	0	0	0	0	0	0	0	0	0	1	0	1

Classifications IF

Infomorphismes

$$F_2$$

$$\begin{array}{llllllllllll}
\zeta_2^{\vee}(\gamma_1) & = & a & \zeta_2^{\vee}(\gamma_2) & = & a & \zeta_2^{\vee}(\gamma_3) & = & a & & & \\
\zeta_2^{\vee}(\gamma_5) & = & d & \zeta_2^{\vee}(\gamma_6) & = & d & \zeta_2^{\vee}(\gamma_7) & = & d & \zeta_2^{\vee}(\gamma_9) & = & d
\end{array}$$

$$F_1'$$

$$\begin{array}{llllllllllll}
\zeta_1'^{\vee}(\gamma_1') & = & a & \zeta_1'^{\vee}(\gamma_2') & = & c & \zeta_1'^{\vee}(\gamma_3') & = & b & \zeta_1'^{\vee}(\gamma_4') & = & d
\end{array}$$

$$F_2'$$

$$\begin{array}{llllllllllll}
\zeta_2'^{\vee}(\gamma_1') & = & a & \zeta_2'^{\vee}(\gamma_2') & = & a & \zeta_2'^{\vee}(\gamma_3') & = & a & \zeta_2'^{\vee}(\gamma_5') & = & d \\
\zeta_2'^{\vee}(\gamma_6') & = & d & \zeta_2'^{\vee}(\gamma_7') & = & d & \zeta_2'^{\vee}(\gamma_9') & = & d & & &
\end{array}$$

Diagramme global

$$
\begin{tikzcd}
 & & & C & & & \\
F_2 \arrow[r, "f_2"'] & \wedge F_2^{\perp} \arrow[urr, "g_2"] & F_2^{\perp} \arrow[l, "\kappa_2"'] & M \arrow[l, "\zeta_2"'] \arrow[r, "\zeta_i'"'] & {F_i'}^{\perp} \arrow[r, "\kappa_i'"'] & \wedge {F_i'}^{\perp} \arrow[ull, "g_i'"'] & F_i' \arrow[l, "f_i'"']
\end{tikzcd}
$$

Classification centrale

	$\{c'_1, c'_2, c'_3, c'_4\}$	$\{c_1, c_2, c_3, c_4\}$	$\{c_5, c_6, c_7, c_9\}$	$\{c_5, c_6\}$	$\{c_1, c_2, c_3, c_4, c_5, c_6, c_7, c_9\}$
(γ_5, γ'_2)	0	0	0	1	0
(γ_5, γ'_3)	0	0	0	1	0
(γ_5, γ'_4)	1	0	0	1	0
(γ_6, γ'_2)	0	1	0	1	1
(γ_6, γ'_3)	0	1	0	1	1
(γ_6, γ'_4)	1	1	0	1	1
(γ_7, γ'_2)	0	1	0	1	1
(γ_7, γ'_3)	0	1	0	1	1
(γ_7, γ'_4)	1	1	0	1	1
(γ_9, γ'_2)	0	1	1	1	1
(γ_9, γ'_3)	0	1	1	1	1
(γ_9, γ'_4)	1	1	1	1	1

	$\{c'_1, c'_2, c'_3, c'_4\}$	$\{c_1, c_2, c_3, c_4\}$	$\{c_5, c_6\}$	$\{c'_1, c'_2, c'_3, c'_4, c'_5, c'_6, c'_7, c'_9\}$
(γ_5, γ'_5)	0	0	1	0
(γ_5, γ'_6)	1	0	1	0
(γ_5, γ'_7)	1	0	1	0
(γ_5, γ'_9)	1	0	1	1
(γ_6, γ'_5)	0	1	1	0
(γ_6, γ'_6)	1	1	1	0
(γ_6, γ'_7)	1	1	1	0
(γ_6, γ'_9)	1	1	1	1
(γ_7, γ'_5)	0	1	1	0
(γ_7, γ'_6)	1	1	1	0
(γ_7, γ'_7)	1	1	1	0
(γ_7, γ'_9)	1	1	1	1
(γ_9, γ'_5)	0	1	1	0
(γ_9, γ'_6)	1	1	1	0
(γ_9, γ'_7)	1	1	1	0
(γ_9, γ'_9)	1	1	1	1

La logique résultante

- typ(C) : union disjointe de $\wedge F_2{}^{\perp}$ et $\wedge F'_i{}^{\perp}$
- tok(C) : couple d'instances (γ_k, γ'_l) qui sont appropriés s'il proviennent du même type de M.
- On en déduit la logique centrale Log_C :

$$\{c'_1 , c'_2 , c'_3 , c'_4 \} \vdash \{c_5 , c_6\}$$

- La logique distribuée $Dlog_C(L)$ sur la somme $F_2 + F'_i$ est l'image inverse de Log_C.

Les résultats

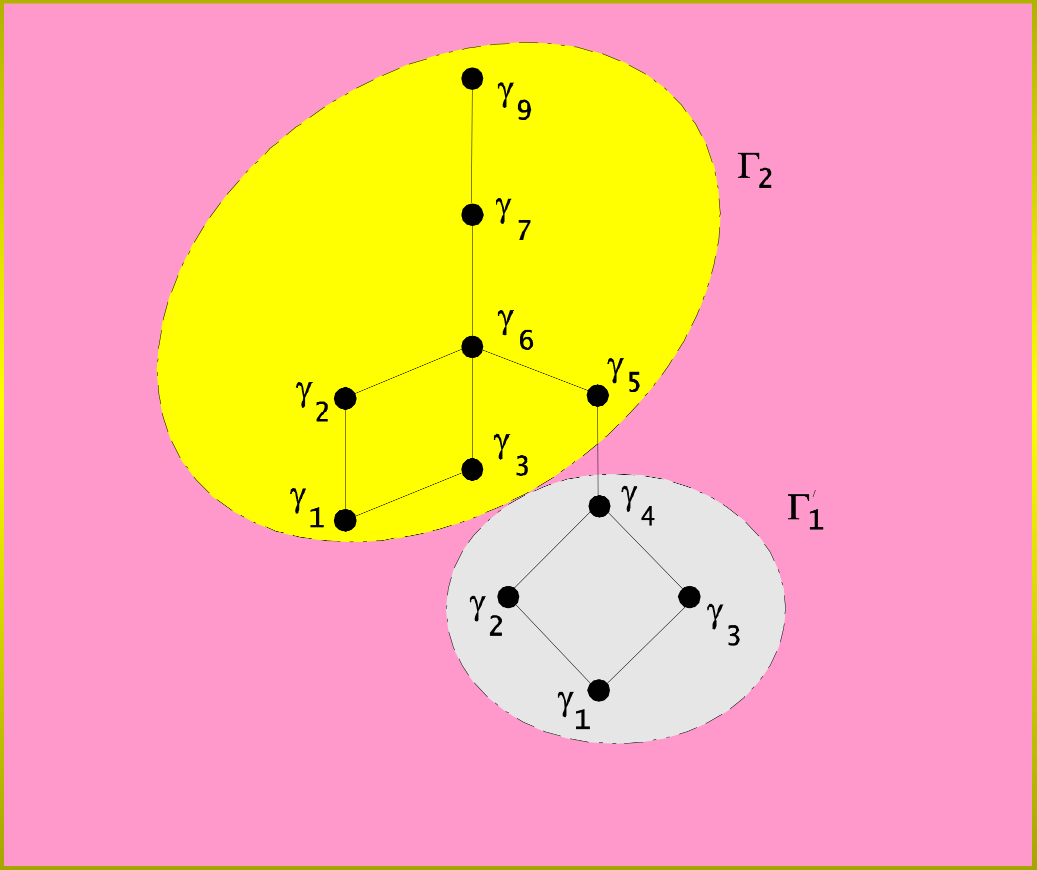
Interopérabilité traduite par les contraintes :

γ'_4 |- γ_5 pour la somme F_2, F'_1

γ'_6 |- γ_5 pour la somme F_2, F'_2

Seule la première contrainte satisfait toutes les conditions (but de type to_send(…))

Les résultats



Conclusion

- Interopérabilité des systèmes de contrôle démontrée logiquement à partir d'un jeu de relations.
- extension du processus de composition d'ontologies [kal03] pour l'automatiser
- Bases catégorielles de l'approche IF : cadre formel valide
- Limitation à la description de processus physiques

Futures directions

- Sujet d’investigation : étendre la représentation des connaissances à d’autres domaines.
- fournir un outil de conception graphique en cours).
- Évaluation et tests du processus de composition (SMA) (en cours)

Bibliographie

[Bal99] Ball T.: The concept of dynamic analysis. Procs. of the ACM Symposium on the Foundations of Software Engineering (ACM SIGSOFT) (1999) pp216-234

[Bar97] Barwise J., Seligman J.: Information Flow. Cambridge tracts in Theoretical Computer Science, 44, (1997) Cambridge University Press.

[Kalfo03] Kalfoglou Y., Schorlemmer M.: IF-Map: An ontology mapping method based on Information Flow theory. Journal of Data Semantics, S. Spaccapietra et al eds., 11, Springer Verlag, (2003) pp98-127

[Kent00] Kent R.E.: The information flow foundation for conceptual knowledge organization, Procs. of the 6th int. conf. of the int. society for knowledge organization, Ed. Ergon Verlag, (2000)

[Schorl03] Schorlemmer M., Kalfoglou Y.: A channel-Theoretic Foundation for Ontology coordination, Procs. of the 2nd European Workshop on Multi-Agent Systems (2004)