

TP

Une petite base de connaissances graphique avec Cogui

Présentation de Cogui. Cogui est un outil de construction de bases de connaissances graphiques. Le formalisme sous-jacent est celui des *graphes conceptuels*, qui se traduit naturellement en logique. On y retrouve les composants classiques des bases de connaissances : vocabulaire composé de concepts et relations (d'arité quelconque ≥ 1), faits, règles (équivalentes aux règles existentielles), contraintes négatives. Cogui permet de vérifier que la base de connaissances est conforme au vocabulaire et dispose d'assistants permettant d'effectuer de petits tests (application de règles, test de la consistance de la base, calcul des réponses à une requête conjonctive, ...).

Sujet. On veut décrire des scènes construites à partir de figures géométriques, puis pouvoir retrouver les scènes possédant certaines caractéristiques. Nous nous intéressons dans un premier temps à la construction du vocabulaire. Dans le monde des graphes conceptuels, le vocabulaire est composé d'ensembles partiellement ordonnés de *types de concept* et de *types de relations*, ces derniers étant munis de *signatures* ; en outre, on peut déclarer que des types de concepts sont *disjoints*.

Aller sur le site de Cogui : <http://www.lirmm.fr/cogui/3/>

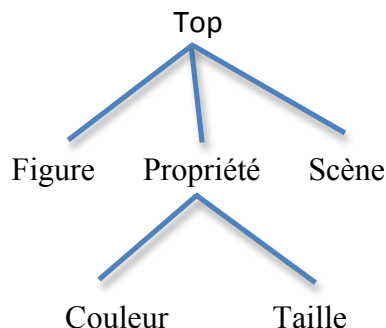
Lancer Cogui (cf. Download en fin de la page d'accueil du site si Cogui n'est pas déjà installé).
Vérifier que vous avez la dernière version : Menu Help / Check for Updates

Créer un projet Cogui : File/New Project --> Empty Cogui Project

Jeter un oeil sur la documentation de Cogui (en cours de mise à jour) : au moins Beginnings/Getting Started

Question 1 : ensemble des types de concept

On se donne l'ensemble de types de concept ci-dessous (avec t_1 spécialisation de t_2 ssi t_1 est un descendant de t_2 sur le schéma).



1-a Compléter en ajoutant des types correspondant aux figures suivantes :
ellipse, cercle, polygone, rectangle, carré, losange.

1-b Indiquer quels types doivent être déclarés disjoints.

Créer cet ensemble de types de concept en utilisant la vue hiérarchique ou la vue graphe (aller dans le Vocabulary du projet pour avoir l'ensemble de types initial composé seulement du type Top).

Pour obtenir la représentation graphe : clic droit sur un type de concept pour avoir le menu contextuel, puis « Graphical Editor » pour afficher la représentation graphe.

- Pour créer un nouveau type : menu contextuel/insert concept type
- Pour tracer un arc dans la vue graphe, se placer sur le sommet du type de départ et tracer l'arc en maintenant enfoncée la touche Maj ou Ctrl (selon le système).

Ajouter les types disjoints (clic droit dans le panneau « Disjoint Types » sous la vue hiérarchique des types de concept) ; si vous indiquez une liste de n types, ces types seront considérés comme *deux à deux* disjoints.

Question 2 : ensemble des types de relation

Construire un ensemble partiellement ordonné de relations, chacune munie de sa signature, permettant de représenter les notions suivantes :

- La position d'une figure par rapport à une autre : *à droite*, *à gauche*, *sur* ("directement sur"), *sous* ("directement sous"), *touche* (les bords des figures se touchent), *proche*, *entre* (une figure peut être entre 2 figures)
- Le fait qu'une certaine figure a une certaine propriété : on appellera cette relation *aProp*
- Le fait qu'une scène *contient* une figure, ou qu'une figure en contient une autre.

Créer cet ensemble de types de relation. Penser à utiliser la suggestion automatique de signature. Valider le vocabulaire (barre d'outils du haut, la coccinelle « check vocabulary »). Cogui émet un avertissement si un sous-ensemble de relations de même arité n'a pas de plus grand élément : toutefois, ce n'est pas une erreur.

Remarque : toujours **valider le vocabulaire**, avant de travailler sur les autres éléments de la base de connaissances.

Question 3 : expressivité du vocabulaire

Pour les connaissances ci-dessous, dites si vous avez pu représenter cette connaissance dans le vocabulaire, et si oui, par quelle notion :

1. pour tous x et y , x est sur y ssi y est sous x [le cas à gauche de / à droite de est similaire]
2. pour tous x , y et z , si x est entre y et z alors x est proche de y et de z
3. pour tous x et y , si x est sur y alors x touche y
4. tout losange est une figure
5. il n'existe pas de figure qui soit à la fois un cercle et un carré
6. toute figure a une couleur.

Question 4 : règles

4-a Parmi les connaissances ci-dessus que vous n'avez pas pu coder dans le vocabulaire, lesquelles sont représentables par des règles de graphes conceptuels ?

Remarque : Les ordres partiels sur les types de concept et les relations peuvent se traduire par des règles. Mais les intégrer directement au vocabulaire permet de les prendre en compte dans l'homomorphisme (opération de base dans tous les raisonnements) ; ce qui est plus efficace.

Créer un répertoire Rules dans votre projet (se placer sur le projet, menu contextuel : new folder), puis y mettre les règles de la question 3 (c'est-à-dire (1), y compris gauche/droite, (2) et (6)).

[Pour créer une règle : menu contextuel depuis Rules, puis new ...]

Ajouter une règle disant que la relation proche est symétrique et une autre disant que la relation contient est transitive.

Remarque : vous pouvez réutiliser une règle existante en la copiant : sélectionner le nom de la règle à copier et utiliser le menu contextuel (clic droit, copy/paste).

Question 5 : faits et règles

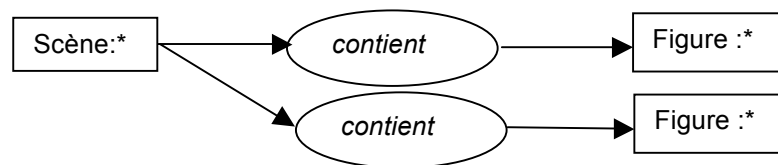
5-a En utilisant le vocabulaire que vous avez construit, dessiner le graphe conceptuel F correspondant à la connaissance suivante :

- La scène n°2 contient un polygone, un grand carré jaune et un petit cercle rouge, le polygone étant posé sur le carré, qui contient le cercle.

Créer ce graphe fait (le mettre dans le dossier Facts du projet) et le saturer en utilisant « Apply rules » (Menu Tools/Reasoning ou les legos dans la barre d'outils du haut). Il est possible de sauvegarder ce graphe : aller dans Factory Cogui (sous la fenêtre du graphe, cliquer sur la flèche verte : vous obtenez la traduction logique du graphe, et à droite « new objects to drag » : prenez le graphe et tirez-le dans la fenêtre Files/Facts à l'endroit voulu). Au besoin, pour afficher la fenêtre « Factory Cogui » : menu Window/Cogui/Factory.

Question 6 : requêtes

6-a On considère le graphe Q1 ci-dessous :



Combien y-a-t-il d'homomorphismes de Q1 dans F* (le graphe saturé de la question précédente) ?

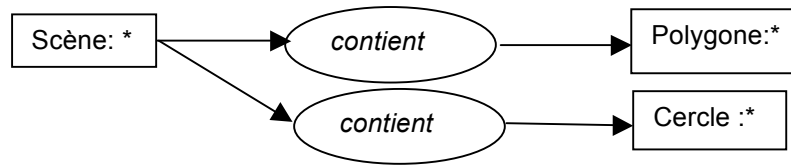
- Créer Q1 et ajouter des variables réponses *x, *y et *z (clic droit sur les sommets concepts et ajout de variables associées à ces sommets).
- Utiliser « Query Wizard » (Menu Tools/Reasoning ou icône (?) dans la barre d'outils du haut) pour vérifier le nombre d'homomorphismes de Q1 dans F*. Visualiser les homomorphismes en parcourant les graphes images (onglet « graph ») ou les réponses avec l'onglet « table ». Les homomorphismes sont appelés « projections » dans Cogui.

- **Si vous vous perdez dans les fenêtres :** Menu Window/ Reset Windows pour retrouver la disposition par défaut

6-b La requête Q1 est censée correspondre à la question « y-a-t-il une scène qui contient deux figures ? » (forme booléenne) ou « quelles sont toutes les scènes qui contiennent deux figures, et quelles sont ces figures » ? Pensez-vous qu'elle traduit effectivement cette question ?

- Vérifier que Q1 est redondant, ce qui explique pourquoi les homomorphismes trouvés ne sont peut-être pas ceux que l'on attendait : Menu contextuel sur le fichier Q1 / Inspect (ou bien l'oeil dans la barre d'outils lorsque Q1 est affichée). Dans la fenêtre de l'inspecteur, vous avez l'information que Q1 est redondant et vous pouvez visualiser les redondances.

6-c On considère maintenant la requête Q2 ci-dessous supposée représenter la question « y-a-t-il une scène contenant deux figures, l'une étant un polygone et l'autre un cercle ? »



- Combien y-a-t-il d'homomorphismes de Q2 dans F* ?
- Q2 traduit-elle effectivement la question ? Pourquoi ?

- Créer Q2.
- Utiliser à nouveau « Query Wizard » pour vérifier le nombre d'homomorphismes de Q2 dans F.

Question 7 : requêtes (suite)

- Faites une copie F' de F et minimiser F' de façon à ce qu'il ne contienne plus d'information qui puisse être inférée par application de règles (notamment : il est inutile de dire que la scène contient le cercle, puisque la relation contient est transitive).
- Créer la requête Q3 : « y-a-t-il une scène qui contient une figure rouge contenue dans un rectangle, lui-même sous une figure » ?
- Q3 a-t-elle une réponse dans F' ? (utiliser Query Wizard sans sélectionner de règles)
- Q3 a-t-elle une réponse dans la saturation de F' avec les règles ?

Question subsidiaire

Dans la question 6-a : on voudrait pouvoir indiquer que l'on cherche deux objets. Comment faire ?

Une solution avec Cogui :

- On transforme la requête en une règle (« si Q1(x,y,z) alors ReponseQ1(x,y,z) », la requête devenant ReponseQ1(x,y,z)).
- Cogui permet d'associer à toute règle un script qui contrôle la façon d'appliquer la règle (ici, on n'appliquerait la règle que si les deux sommets de type Figure ont des images différentes par l'homomorphisme de son hypothèse) : afficher la règle, aller dans l'onglet script et sélectionner le petit entonnoir, ce qui insère le script ci-dessous :

```

boolean before(boolean applicable)
{
  return true;
}
  
```

La règle ne sera appliquée pour un certain homomorphisme que si la fonction before retourne vrai. Ici, on peut indiquer `return X != Y ;` pour vérifier que les variables X et Y aient des images différentes.

A rendre (d'ici le vendredi 12 avril)

- un document (.pdf) avec les réponses aux questions. Vous pouvez faire des images de toutes les vues graphes par le menu contextuel « Export Image ».
- un zip de votre projet Cogui.