

Ingénierie Des Connaissances

Analyse Relationnelle de Concepts

Théo Chambon

2014 – 2015



**UNIVERSITÉ
DE MONTPELLIER**

PARTIE 1 : Analyse de Cartes de Pizza

La première étape afin de construire un treillis puis d'en analyser les données est la sélection de données pertinentes.

Pour cela, ai sélectionné deux sites parmi les propositions : dominos et les pizzas de charlotte.

Une fois ces sites sélectionnés, il a fallu choisir quelques une des pizzas, en tentant de garder à la fois quelques pizzas présentant des similarités, et des pizzas différentes les unes des autres, afin de pouvoir ensuite obtenir différentes catégories.

De plus, l'ensemble des pizzas étaient disponibles en plusieurs tailles, et avec plusieurs pâtes, ce qui était difficilement compatible avec les données recherchées. Ainsi, en sélectionnant une pizza, on sélectionne donc la pizza, sa taille, et son type de pâte.

Voici donc la liste des pizzas sélectionnées, leur composition, taille et prix, pour chacune des deux enseignes :

- Domino's :
 - Orientale Moyenne Pate Normale,
 - Margherita Grande pate fine
 - 4 fromages Moyenne Pate Epaisse Base Blanche
 - Magic Saumon Grande Pate Fine

- Pizzas de Charlottes :
 - Calzone Grande,
 - Chorizo Petite,
 - Australienne Moyenne
 - Méditerranéenne petite,
 - grecque grande,
 - Saumon Moyenne,
 - Fruits de mer Grande

On voit apparaître trois grandes catégories dans ces pizzas : viande, végétarienne et poisson.

A partir de ces données, il a fallu définir les différents attributs que l'on va utiliser. Les principaux attributs seront les différents ingrédients, les tranches de prix, les types de pâtes, et les tailles. De plus, il est intéressant de regrouper ces différents attributs par catégories, par exemple les viandes, les légumes, les pâtes...

Certains attributs, comme le prix par exemple, pourrait être des attributs valués. Mais cela n'est pas pris en charge par le logiciel Conexp, on va donc directement créer différentes fourchettes de prix

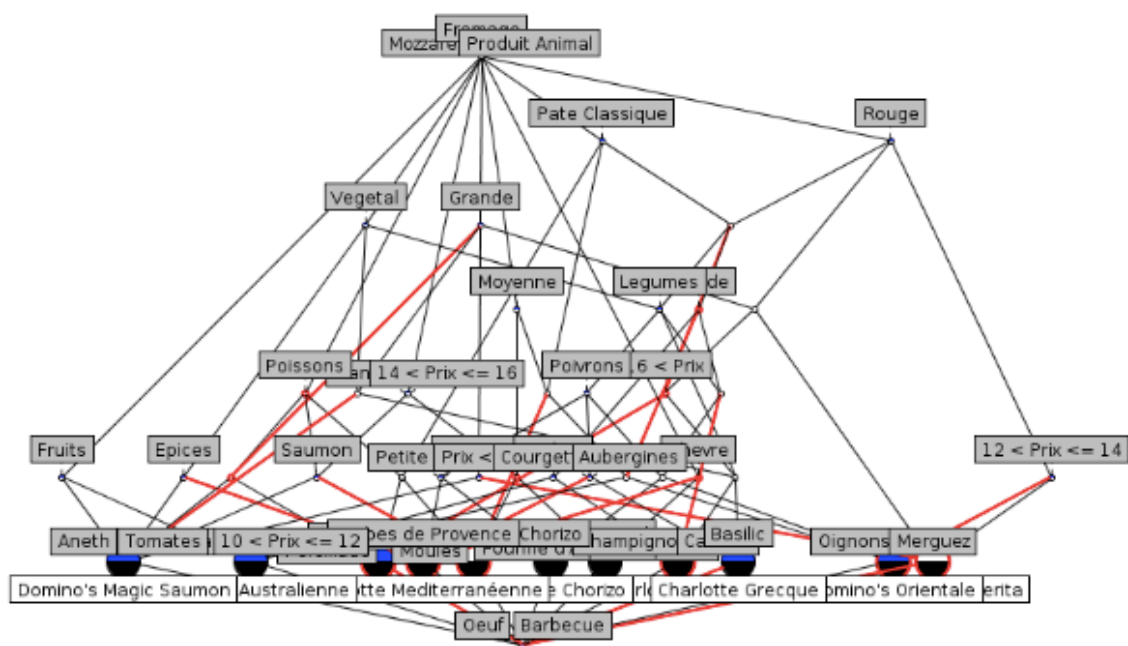
Ainsi, on va donc obtenir une taxonomie des différents attributs sélectionnés.

Une fois tous les attributs obtenus, et ordonnés sous forme de taxonomie, on peut construire le tableau des attributs, ici à l'aide du logiciel Conexp.

	Prix < 10	10 < Prix <= 12	12 < Prix <= 14	14 < Prix <= 16	16 < Prix	Petite	Moyenne	Grande	Blanche	Rouge	Barbecue	Pate Fine	Pate Cla...	Pate Epa
Domino's Orientale			X				X			X			X	
Domino's Margherita			X				X	X		X		X		X
Domino's 4 Fromages				X					X					
Domino's Magic Saumon				X	X			X	X			X		
Charlotte Calzone										X			X	
Charlotte Chorizo	X				X	X	X			X			X	
Charlotte Australienne		X					X			X			X	
Charlotte Méditerranéenne	X					X	X			X			X	
Charlotte Grecque				X	X		X	X	X	X			X	
Charlotte Saumon					X		X						X	
Charlotte Fruits de Mer					X		X	X	X	X			X	

Tableau représentant les pizzas et leurs attributs

Sur ce tableau (on peut en voir une partie ci-dessus), on peut ainsi voir la liste des attributs d'une pizza, et pour chaque pizza quels sont les attributs qu'elle possède. On peut voir aussi que chaque pizza possédant du fromage aura l'attribut père "Fromage" coché, ainsi que l'attribut lui même (c'est à dire quel fromage). Il en va de même pour les viandes par exemple.



Treillis

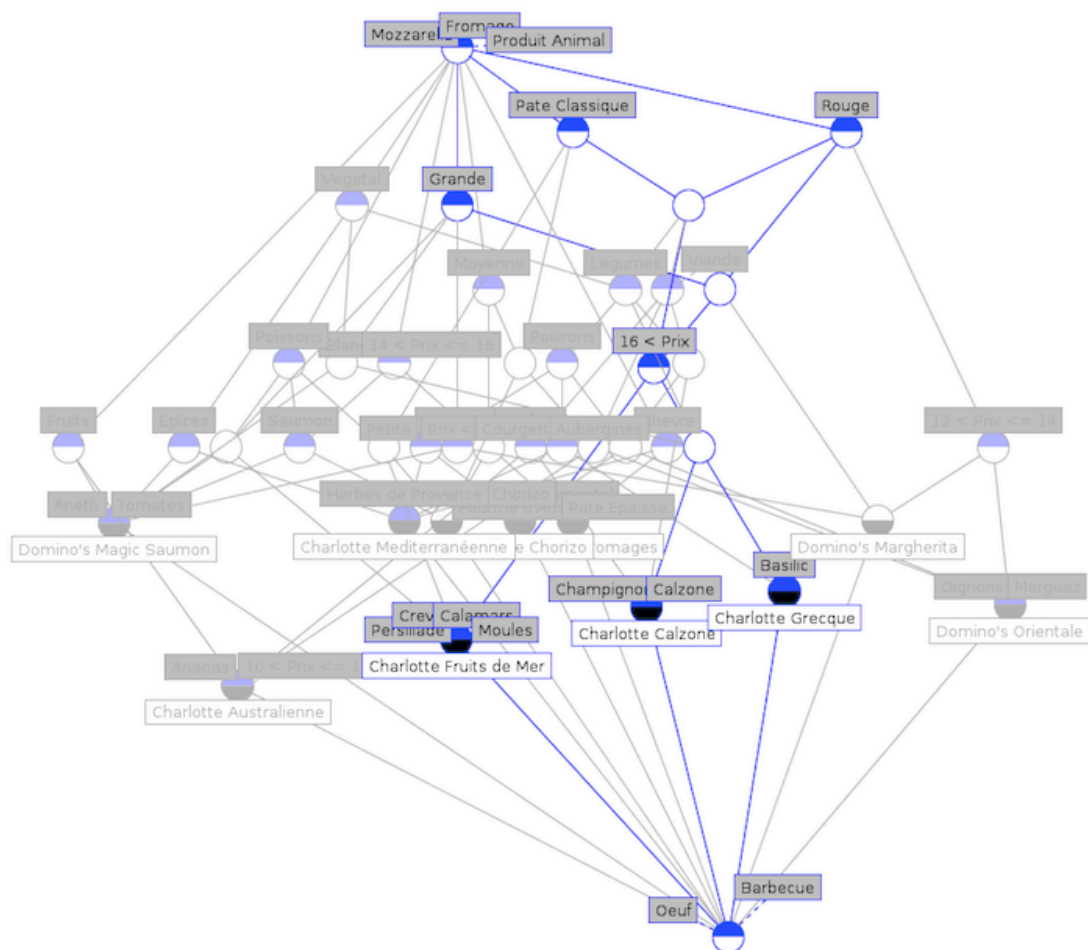
Une fois la saisie terminée, on peut donc générer le treillis ci dessus.

QUESTIONS

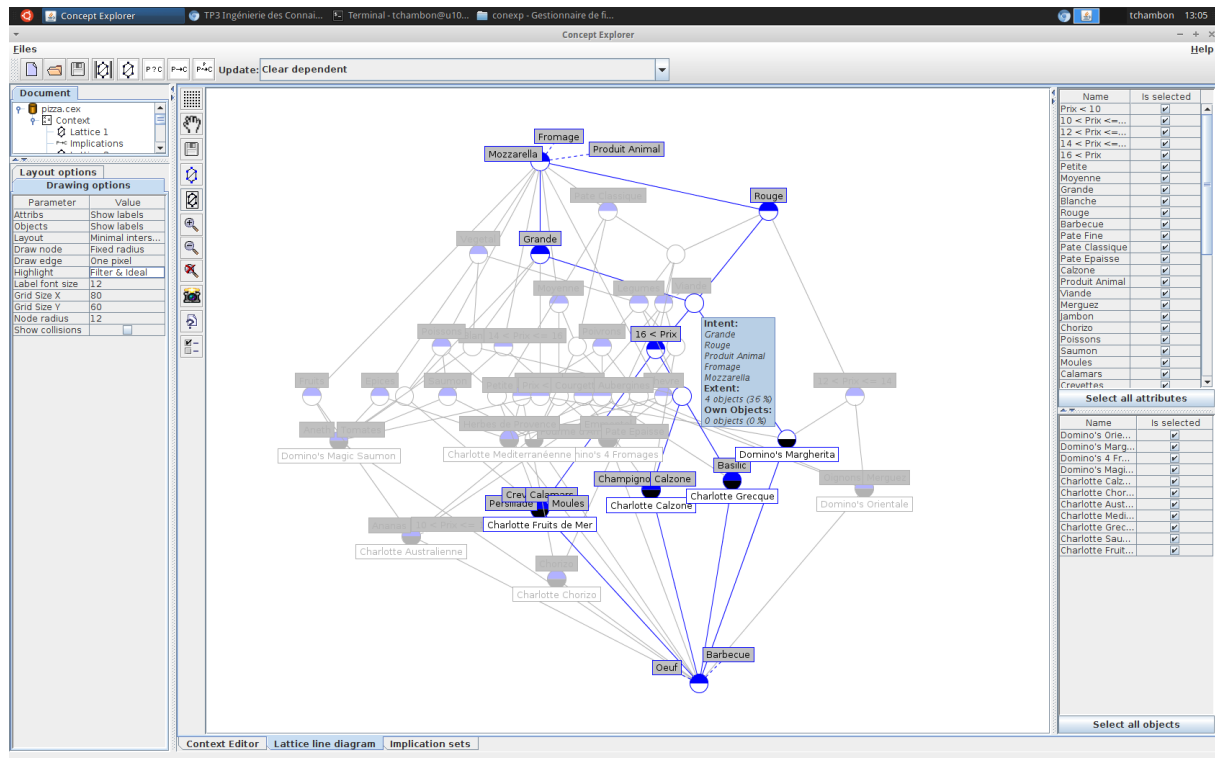
1/ Comme observé précédemment, on retrouve trois grandes catégories principales chez les deux pizzaiolos sélectionnés : les pizzas à la viandes, celles au poissons, et les végétariennes.

2/ A l'aide du treillis, on peut mettre en places de nouvelles catégories, en sélectionnant un ou plusieurs noeuds portant sur les caractéristiques nous intéressant pour ces catégories. Ainsi, par exemple, on va pouvoir créer des catégories de prix, en sélectionnant un certain prix sur le treillis.

Par exemple, en sélectionnant le noeud correspondant à un prix supérieur à 16€, on obtient les pizzas de cette catégorie, les pizzas chères :

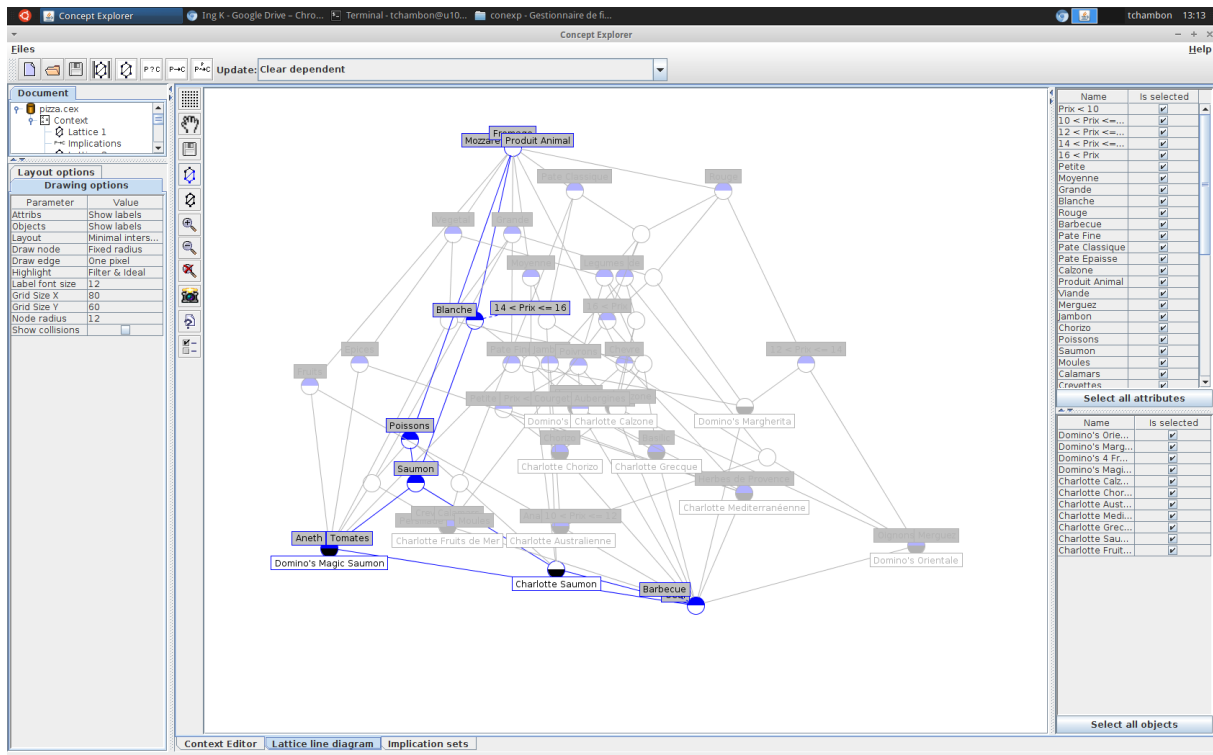


On peut aussi sélectionner des noeuds regroupant plusieurs attributs différents, et ainsi créer de nouvelles catégories. Par exemple, en sélectionnant le noeud mis en avant ci dessous, on obtient les pizzas à la sauce tomate et à la viande, de grande taille. Cela peut par exemple être une nouvelle catégorie intéressante.



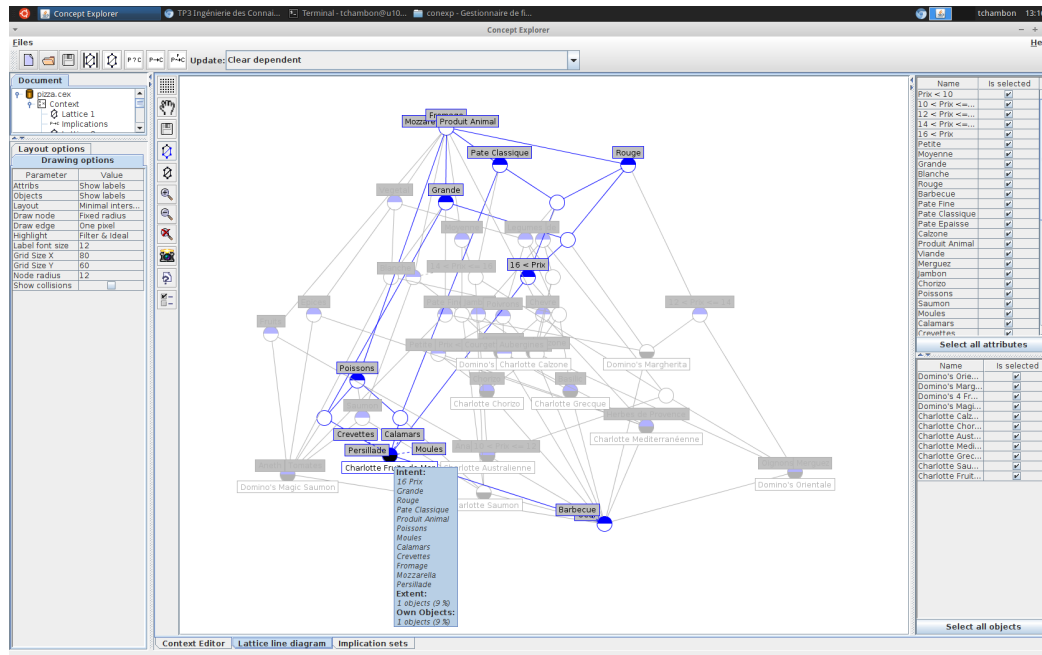
3/ A l'aide du treillis, on peut tout d'abord voir en sélectionnant une pizza quels sont les noeuds qui y correspondent. Ensuite, il suffit de sélectionner ses noeuds pour obtenir des pizzas proche de la pizza choisie, car les deux pizzas partageront ainsi tout les attributs de ce ou ces noeuds, et seront donc proches. Deux pizzas étant proches si elles partagent une partie importante de leurs attributs.

Par exemple, si je cherche une pizza proche de la pizza Magic Saumon de chez Domino's, en sélectionnant le noeud Saumon, j'obtiendrais les pizzas assez proche (et au saumon) de l'autre pizzaiolo.



Les différences seront observables en regardant les noeuds reliés à une pizza sans l'être à l'autre. Cela permet de mettre en avant les attributs non partagés.

Pour les pizzas végétariennes, il faudra appliquer la même procédure, mais avec le noeud viandes, ou encore Produit Animal pour les végétaliens (mais toutes les pizzas ayant du fromage cela ne donnera aucun résultats avec les données actuelles).



Fruits de mer => Prix > 16

Relations : a_taille(Domino's Margherita, Grande), a_ingrédient(Domino's Margherita, Mozzarella), a_ingrédient(Domino's Margherita, Pate Classique)
a Prix(Charlotte Fruits de Mer, >16), a Prix(<10), a Taille(, Petite)

Petite Pizza = Pizza ET aPourTaille.Petit ET aPourPrix.<10

PARTIE 2 : Comparaison de Logiciels Wikis

La première étape de ce TP a été de choisir un jeu de données intéressant et représentatif afin que l'étude de ces données soit intéressante. Les données choisies sont une partie de la liste des Wikis, une partie des Licences logicielles, et une partie des systèmes de gestion de bases de données. Ces listes de données étant très fournies, j'ai donc choisi de conserver seulement les attributs intéressants dans notre cas, et quelques exemples pour chacune des listes, afin de ne pas avoir des tableaux illisibles, mais de manière à conserver des relations intéressantes entre les objets et un échantillon le plus représentatif possible.

Voici donc le jeu de données que j'ai choisi de conserver :

- Wikis
 - Software Licence
 - Data Backend
 - Open Source
 - Multilingual
- Licences
 - Licence
 - Link with a different licence
- SGBD
 - Maintainer
 - License

Wikis :

	Software license ^[1]	Open source	Multilingual	Data backend
BlueSpice for MediaWiki	GPL	Yes	Yes	MySQL, ApacheDerby
BrainKeeper	Proprietary	No	Yes	MySQL
eXo	LGPL	Yes	Yes	PostgreSQL, MySQL, Oracle,

Platform				Apache Derby, HSQLDB
FlexWiki	GPL	Yes		File system,Microsoft SQL Server
Ikiwiki	GPL	Yes ^[11]	Yes	Oracle

Licences :

License	Link with code using a different license
GNU Affero General Public License	Yes
GNU General Public License	No
GNU Lesser General Public License	Yes
Open Software License	Yes

SGBD :

Maintainer	License
Apache Derby	Apache License
Microsoft SQL Server	Proprietary
MySQL	GPL or Proprietary
Oracle	Proprietary
SQLite	Public domain

Une fois ces données sélectionnées, il a fallu les coder dans une famille de contextes, à l'aide de l'application RCAExplore.

Tout d'abord, il a fallu créer les objets et leurs attributs, en se basant sur les données extraites précédemment.

	OpenSou...	MultiLang...
BlueSpic...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
BrainKee...	<input type="checkbox"/>	<input checked="" type="checkbox"/>
eXoPlatfo...	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
FlexWiki	<input checked="" type="checkbox"/>	<input type="checkbox"/>
IkiWiki	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

	Link with ...
GNU AGPL	<input checked="" type="checkbox"/>
GPL	<input type="checkbox"/>
LGPL	<input checked="" type="checkbox"/>
Proprietary	<input checked="" type="checkbox"/>
Apache Li...	<input type="checkbox"/>
Public Do...	<input checked="" type="checkbox"/>

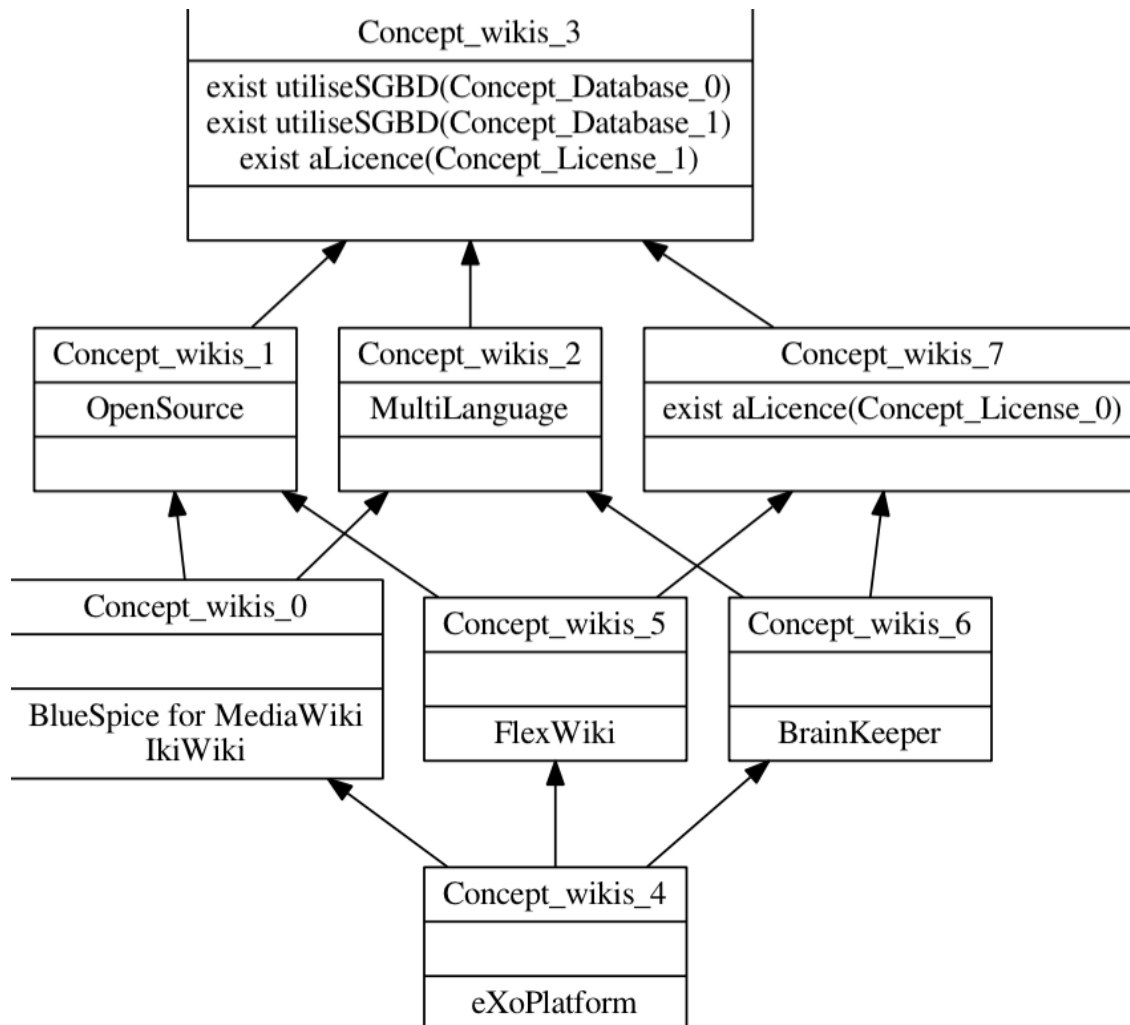
Une fois ces objets créés, il a fallu créer les relations entre ces objets, c'est à dire quels logiciels utilisent quelles licences et systèmes de gestions de bases de données.

wikis	Database	License	utiliseSGBD	sdbALicense	aLicence	
scaling operator:		exist	▼	description:		
	GNU AGPL	GPL	LGPL	Proprietary	Apache Li...	Public Do...
ApacheDerby	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Microsoft SQL Server	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
MySQL	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Oracle	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
PostGreSQL	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

wikis	Database	License	utiliseSGBD	sdbALicense	aLicence	
scaling operator:		exist	▼	description:		
	ApacheD...	Microsoft ...	MySQL	Oracle	PostGreSQL	
BlueSpice for MediaWiki	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
BrainKeeper	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
eXoPlatform	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
FlexWiki	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
IkiWiki	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Une fois tout cela créé, il est possible de générer la famille de treillis à l'aide de RCAExplore.

En analysant la famille de treillis, on peut voir de nouveaux concepts représentant des catégories d'objets. Par exemple, on voit apparaître des catégories simples, tels que les logiciels OpenSource et ceux qui sont multilingues, ces catégories sont simplement basées sur un seul attribut. Mais on va aussi voir des catégories combinant plusieurs concepts, ainsi on va voir apparaître le groupe des logiciels OpenSource et multilingues (IkiWiki et BlueSpice). Ou encore les logiciels OpenSource et ayant des licences du nouveau concept *Concept_Licence_0* (les Licences permettant d'utiliser du code ayant un lien avec du code utilisant une autre licence).



Les groupes apparaissant pour licences et les SBDG sont plus simple car uniquement basés sur les attributs, et pas sur les relations avec d'autres objets, ils sont donc moins intéressants à analysés.

Enfin, voici un exemple d'une partie de la logique de description correspondant à cette étude de cas :

Concept_Wiki_0 : OpenSource ^ Multilanguage ^ Concept_Wiki_3

Concept_Wiki_0(BrainKeeper)

Wiki(BlueSpice), Wiki(BrainKeeper), Licence(GPL), Licence(OpenSource), SGBD(Oracle), SGBD(MySQL)

utiliseSGBD(BrainKeeper, MySQL), SGBDaLicence(MySQL, GPL), aLicence(BrainKeeper, Proprietary)