

Analyse de concepts formels guidée par des connaissances de domaine : Application à la découverte de ressources génomiques sur le Web

Nizar Messai

▶ **To cite this version:**

Nizar Messai. Analyse de concepts formels guidée par des connaissances de domaine : Application à la découverte de ressources génomiques sur le Web. Informatique [cs]. Université Henri Poincaré - Nancy I, 2009. Français. NNT : . tel-00446548

HAL Id: tel-00446548
https://theses.hal.science/tel-00446548

Submitted on 12 Jan 2010

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Nancy-Université
Université Henri Poincaré

Département de formation doctorale en informatique **École doctorale IAEM Lorraine**
UFR STMIA

Analyse de concepts formels guidée par des connaissances de domaine : Application à la découverte de ressources génomiques sur le Web

THÈSE

présentée et soutenue publiquement le 20 Mars 2009

pour l'obtention du

Doctorat de l'université Henri Poincaré – Nancy 1

(spécialité informatique)

par

Nizar Messai

Composition du jury

Rapporteurs :	Jean Diatta	Professeur, Université de La Réunion
	Marianne Huchard	Professeur, Université Montpellier 2
Examinateurs :	François Brucker	Maître de conférences, Université Paul Verlaine-Metz
	Claude Godart	Professeur, Université Henri Poincaré - Nancy1
	Amedeo Napoli	Directeur de recherches, CNRS, Nancy
	Henri Soldano	Maître de conférences, Université Paris 13
Invité :	Marie-Dominique Devignes	Chargée de recherches, CNRS, Nancy

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Loria

Mis en page avec la classe thloria.

Table des matières

Table des figures **vii**

Liste des tableaux **xi**

1 Introduction générale **1**

1.1 Contexte de travail 1

1.1.1 Les ressources Web 1

1.1.2 La découverte de ressources biologiques sur le Web 4

1.2 Objectif de la thèse et principales contributions 6

1.3 Organisation du manuscrit 7

2 Contexte de l'étude : découverte de ressources biologiques sur le Web **9**

2.1 Introduction 9

2.2 Description de l'existant 10

2.2.1 Principales caractéristiques des sources de données biologiques 10

2.2.2 Méthodes existantes pour l'identification et l'exploitation des sources de données biologiques 10

2.2.3 Limites 13

2.3 BioRegistry : un annuaire sémantique de sources de données biologiques 13

2.3.1 Prise en compte des connaissances de domaine dans la modélisation de l'annuaire BioRegistry 13

2.3.2 Peuplement de l'annuaire BioRegistry 16

2.3.3 Interrogation de l'annuaire BioRegistry 17

2.4 Découverte de sources de données biologiques "guidée par des connaissances" . . . 18

3 Analyse de Concepts Formels : fondement théorique, applications et extensions **21**

3.1 Introduction 22

3.2 Théorie des treillis : Notions de base 22

3.2.1 Ensemble ordonné 22

3.2.2 Treillis . . . 23
3.2.3 Fermeture . . . 24
3.2.4 Connexion de Galois . . . 24
3.3 Analyse de concepts formels . . . 25
3.3.1 Origine et contexte philosophique . . . 25
3.3.2 Contexte formel . . . 25
3.3.3 Connexion de Galois dans un contexte formel . . . 25
3.3.4 Concept formel . . . 26
3.3.5 Treillis de concepts . . . 27
3.3.6 Algorithmes de construction de treillis de concepts . . . 28
3.3.7 Implications dans un contexte formel . . . 33
3.4 ACF et découverte de ressources . . . 35
3.4.1 Principales motivations . . . 36
3.4.2 ACF et Recherche d'Information . . . 36
3.5 ACF et données complexes . . . 39
3.5.1 Contexte formel multivalué . . . 39
3.5.2 Échelonnage conceptuel . . . 40
3.6 Les extensions de l'ACF . . . 42
3.6.1 Analyse de Concepts Formels Flous . . . 43
3.6.2 Analyse de Concepts Logiques . . . 46
3.6.3 Extensions Relationnelles de l'ACF . . . 48
3.6.4 Extension aux objets symboliques . . . 53
3.6.5 Autres extensions . . . 54

4 Découverte de ressources par treillis de concepts dans BioRegistry 55
4.1 Introduction . . . 55
4.2 Représentation du contenu de BioRegistry en ACF . . . 56
4.2.1 Classification flexible du contenu de BioRegistry . . . 58
4.3 Exploitation du contenu de BioRegistry . . . 58
4.3.1 Navigation . . . 58
4.3.2 Interrogation . . . 59
4.4 Formalisation : l'algorithme BR-Explorer . . . 61
4.4.1 Définitions . . . 61
4.4.2 L'algorithme BR-Explorer . . . 63
4.4.3 Déroulement de BR-Explorer sur un exemple . . . 63
4.4.4 Expressivité des requêtes dans BR-Explorer . . . 66
4.4.5 Correction et complétude de BR-Explorer . . . 67

4.4.6 Complexité de BR-Explorer . . . 68
4.5 Raffinement de requête à partir de ressources sémantiques . . . 68
4.5.1 Raffinement de requête par généralisation . . . 70
4.5.2 Raffinement de requête par spécialisation . . . 71
4.5.3 Raffinement mixte . . . 72
4.5.4 Choix du type de raffinement . . . 72
4.6 Dépendances entre attributs : hiérarchie d'attributs dans BR-Explorer . . . 73
4.6.1 Le besoin d'exprimer les dépendances entre attributs . . . 73
4.6.2 Formalisation . . . 74
4.6.3 Application des hiérarchies d'attributs à un treillis de concepts . . . 75
4.6.4 Hiérarchies d'attributs pour la découverte de ressources guidée par des connaissances . . . 77
4.6.5 Connaissances globales ou connaissances locales . . . 82

5 Étude des contextes multivalués 85
5.1 Introduction . . . 85
5.2 Besoin d'extension aux données complexes . . . 86
5.2.1 Données réelles et contextes multivalués . . . 86
5.2.2 Limites des approches existantes . . . 87
5.2.3 Principe général et étapes de l'approche proposée . . . 88
5.3 Étude des contextes numériques . . . 88
5.3.1 Contextes numériques . . . 88
5.3.2 Similarité entre les valeurs dans un contexte multivalué . . . 89
5.3.3 Représentation équivalente des contextes multivalués . . . 90
5.3.4 Opérateurs de dérivation . . . 90
5.3.5 Ordres partiels . . . 92
5.3.6 Connexion de Galois par similarité . . . 93
5.3.7 Concepts formels multivalués et treillis de concepts multivalués . . . 93
5.3.8 Variation de la précision dans les treillis de concepts multivalués . . . 94
5.3.9 Évolution du treillis de concepts multivalués . . . 96
5.3.10 Treillis extrêmes et liens avec l'échelonnage conceptuel . . . 98
5.4 Études des contextes symboliques . . . 100
5.4.1 Contextes symboliques . . . 100
5.4.2 Similarité entre les valeurs dans un contexte symbolique . . . 100
5.4.3 Ordres Partiels, opérateurs de dérivation et connexion de Galois par similarité 102
5.4.4 Concepts formels multivalués et treillis de concepts multivalués . . . 104

5.4.5 Variation de la granularité des concepts multivalués et évolution du treillis de concepts multivalués . . . 105
5.5 Généralisation aux contextes plurivalués et aux contextes hétérogènes . . . 108
5.5.1 Contexte plurivalué numérique . . . 109
5.5.2 Contexte plurivalué symbolique . . . 110
5.5.3 Contexte hétérogène . . . 112

6 Découverte de ressources par treillis de concepts multivalués 113
6.1 Introduction . . . 113
6.2 Classification à granularité variable . . . 113
6.3 Navigation . . . 114
6.3.1 Navigation statique . . . 114
6.3.2 Navigation dynamique : les zooms avant/arrière dans les treillis de concepts multivalués . . . 114
6.4 Interrogation . . . 116
6.4.1 Les requêtes multivaluées . . . 116
6.4.2 Pertinence des objets par rapport à une requête multivaluée . . . 118
6.4.3 Stratégie de recherche des objets pertinents . . . 119
6.4.4 Choix du treillis de concepts multivalués à interroger . . . 120
6.4.5 Illustration dans le cas de contextes numériques . . . 121
6.4.6 Illustration dans le cas de contextes symboliques . . . 124

7 Mise en oeuvre 129
7.1 Introduction . . . 129
7.2 Le prototype BR-Explorer . . . 129
7.2.1 Architecture générale du système BR-Explorer . . . 129
7.2.2 Le noyau du système BR-Explorer . . . 130
7.2.3 L'interface utilisateurs du système BR-Explorer . . . 130
7.2.4 Expérimentation . . . 131
7.3 Le système SimBA . . . 131
7.4 Architecture générale du système SimBA . . . 131
7.4.1 Expérimentation . . . 132

8 Conclusion et perspectives 135
8.1 Conclusion générale . . . 135
8.2 Perspectives . . . 136

Bibliographie 137

A Exécution du Système BR-Explorer **151**

A.1 Visualisation du treillis de concepts 151

A.2 Interface de requête de BR-Explorer 152

A.3 Visualisation du résultat de l'exécution de BR-Explorer sur une requête 153

A.4 Visualisation du voisinage de la requête dans le treillis 154

A.5 Interface de définition de hiérarchies d'attributs dans BR-Explorer 155

Table des figures

1.1 L'évolution du nombre des sites Web entre août 1995 et octobre 2008 : étude faite par NETCRAFT. 2
1.2 L'évolution du nombre des sources de données biologiques disponibles sur le Web entre 1999 et 2009. Cette étude est effectuée en s'appuyant sur les recensements annuels des sources de données disponibles effectués par la revue Nucleic Acids Research 5

2.1 Le modèle de métadonnées BioRegistry. 15
2.2 Le modèle relationnel de la base de données de l'annuaire BioRegistry. 16
2.3 Déroulement du processus d'extraction automatique des métadonnées relatives aux sources de données dans l'annuaire BioRegistry. La métadonnées avec une astérisque correspondent aux métadonnées du DCMI. 17
2.4 L'architecture orientée services (SOA). 18
2.5 Role de l'annuaire BioRegistry dans un système de découverte de sources de données biologiques guidée par des connaissances de domaine. 19

3.1 Le treillis de concepts correspondant au contexte formel $\mathbb{K} = (G, M, I)$ donné dans la table 3.1 28
3.2 Les étapes de la construction incrémentale du treillis de concepts correspondant au contexte formel donné dans la table 3.2. L'ajout d'une nouvelle ligne au treillis construit est symbolisée par "$\xrightarrow{(+\ \mathrm{n})}$" où n désigne le numéro de la ligne considérée. Les modifications engendrées par un tel ajout sont indiquées en rouge sur le treillis. 32
3.3 Les treillis de concepts correspondant aux contextes formels donnés dans les tables 3.4 : $\underline{\mathfrak{B}}(\mathbb{K}_1)$ à gauche et $\underline{\mathfrak{B}}(\mathbb{K}_2)$ à droite. 33
3.4 Liens lien entre les concepts des treillis $\underline{\mathfrak{B}}(\mathbb{K}_1)$ et $\underline{\mathfrak{B}}(\mathbb{K}_2)$ et ceux du treillis $\underline{\mathfrak{B}}(\mathbb{K})$. 34
3.5 Les implications minimales non redondantes à support non nul déduites directement à partir du treillis de concepts. 35
3.6 Le treillis de concepts correspondant au contexte formel donné dans le tableau 3.9. 43
3.7 Taxonomie d'enrichissement - Hiérarchie d'attributs 49
3.8 Treillis monovalués : Touristes (à gauches), Destinations (à droite). 52
3.9 Treillis résultant après enrichissement relationnel. 53
3.10 Treillis de données multivaluées : Intersection (à gauche), Union (à droite). 54

4.1 Le treillis de concepts correspondant au contexte formel donné dans la table 4.2 . 57
4.2 Le treillis de concepts résultant de l'insertion de la requête. 65
4.3 Le treillis de concepts correspondant au contexte formel donné dans la table 4.4. 69
4.4 *TaxoBR* : une partie de la taxonomie des organismes. 70

4.5 Représentation graphique des exemples de dépendances entre attributs : (a)- $m_1 \succ m_2$, (b)- $m_1 \succ (m_2 \sqcup m_3)$ équivalente à $(m_1 \succ m_2) \sqcap (m_1 \succ m_3)$, (c)- $(m_1 \sqcap m_2) \succ m_3$ (équivalente à $(m_1 \succ m_3) \sqcap (m_2 \succ m_3)$), (d)- $(m_1 \sqcap m_2) \succ (m_3 \sqcup m_4)$ (équivalente à $(m_1 \succ m_3) \sqcap (m_2 \succ m_3) \sqcap (m_1 \succ m_4) \sqcap (m_2 \succ m_4)$). 74
4.6 La représentation graphique de deux exemples de hiérarchies d'attributs. (a) La hiérarchie d'attributs obtenue par la définition des dépendances entre attributs de la requête cherchant les restaurants *italiens près du musée du Louvre* et *ayant une terrasse* [Carpineto and Romano, 2005] : $\mathcal{HA}$:*Near the Louvre museum* $\succ$ *Italian* $\succ$ *with a "dehors"*. (b) La hiérarchie d'attributs obtenue par la définition des dépendances entre attributs de la requête cherchant les sources de données de BioRegistry contenant des données relatives aux *mammifères* et ayant un contenu *complet* et *à jours* : $\mathcal{HA}$: *Mammals* $\succ$ *(Complete* $\sqcup$ *Updated)*. 75
4.7 La hiérarchie de concepts $\mathfrak{B}_{\mathcal{HA}_1}(\mathbb{K})$ résultant de l'application de $\mathcal{HA}_1$ au treillis $\underline{\mathfrak{B}}(\mathbb{K})$ donné dans Figure 4.1. 77
4.8 Première étape de navigation dans le treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$ guidée par des connaissances exprimées par $\mathcal{HA}$: $Mammals \succ (Complete \sqcup Updated)$. 78
4.9 Déroulement de l'algorithme BR-Explorer avec prise en compte de préférences entre attributs de la requête $Q = (X,\{$*Mammals, Complete, Updated*$\})$. Les préférences sont données par $\mathcal{HA}$: *Mammals* $\succ$ *(Complete* $\sqcup$ *Updated)*. 81
4.10 Déroulement de l'algorithme BR-Explorer sur le treillis resultant de l'insertion de Q= $(\{x\},\{$*Mammifères, Complet, À jour*$\})$ dans le treillis $\underline{\mathfrak{B}}_{\mathcal{HA}_1}(G, M, I)$. 83

5.1 Le treillis de concepts multivalués $\underline{\mathfrak{B}}_{0.2}(G, M, W, I)$ correspondant au contexte formel multivalué (G, M, W, I) donné en table 5.3. 94
5.2 Les treillis de concepts multivalués correspondant au contexte multivalué (G, M, W, I) donné en table 5.3 pour les seuils de variation $\theta = 0.3$ (gauche) et $\theta = 0.5$ (droite). 95
5.3 Les treillis de concepts multivalués correspondant au contexte multivalué (G, M, W, I) donné en table 5.3 pour les seuils de variation $\theta = 0.8$ et $\theta = 0.9$. 96
5.4 L'évolution du treillis de concepts multivalués correspondant au contexte formel multivalué (G, M, W, I) donné dans la table 5.3, en terme de nombre de concepts multivalués, en fonction du seuil de variation θ. 97
5.5 L'évolution du treillis de concepts multivalués correspondant au contexte formel multivalué (G, M, W, I) donné à la table 5.3, en terme du nombre de concepts multivalués, en fonction du seuil de variation θ (θ varie entre 0.2 et 0.4) : la phase de création 97
5.6 L'évolution du treillis de concepts multivalués correspondant au contexte formel multivalué (G, M, W, I) donné à la table 5.3, en terme de nombre de concepts multivalués, en fonction du seuil de variation θ (θ varie entre 0.8 et 1) : la phase de fusion 98
5.7 Treillis de concepts multivalués $\underline{\mathfrak{B}}_0(G, M, W, I)$ (gauche) et treillis de concepts formels $\underline{\mathfrak{B}}(G, M, I)$ correspondant à l'échelle donné à la table 5.4 (droite). 99
5.8 Treillis de concepts multivalués $\underline{\mathfrak{B}}_1(G, M, W, I)$ (gauche) et treillis de concepts formels $\underline{\mathfrak{B}}(G, M, I)$ correspondant à l'échelle donné à la table 5.5 (droite). 99
5.9 Fragments de hiérarchies contenant les valeurs des attributs du contexte multivalué donné à la table 5.6. 101
5.10 $\underline{\mathfrak{B}}_2(G, M, W, I)$ correspondant au contexte symbolique (G, M, W, I) donné dans la table 5.6. 106

5.11 $\underline{\mathfrak{B}}_0(G, M, W, I)$ correspondant au contexte symbolique (G, M, W, I) donné dans la table 5.6. 107
5.12 $\underline{\mathfrak{B}}_4(G, M, W, I)$ correspondant au contexte symbolique (G, M, W, I) donné à la table 5.6. 107
5.13 $\underline{\mathfrak{B}}_{max}(G, M, W, I)$ correspondant au contexte symbolique (G, M, W, I) donné dans la table 5.6. 108
5.14 Le treillis de concepts plurivalués correspondant au contexte plurivalué numérique donné à la table 5.8 pour les seuils $\theta_{DT50} = 30$, $\theta_{Aquatox} = 100$ et $\theta_{DJA} = 0.1$. . . 111

6.1 Un exemple de zoom avant/arrière des concepts multivalués dans un processus de navigation dynamique. 115
6.2 Le treillis de concepts multivalués correspondant au contexte réduit donné à la table 6.1 pour un seuil de similarité $S = 2$. 117
6.3 Treillis de concepts multivalués $\underline{\mathfrak{B}}_{0.1}(G, M, W, I)$. 121
6.4 Treillis de concepts multivalués résultant de l'insertion de la requête $Y = \{D_{0.94}, M_{0.83}, DS_{0.65}\}$ dans $\underline{\mathfrak{B}}_{0.1}(G, M, W, I)$. 122
6.5 Recherche des objets pertinents dans le treillis obtenu 122
6.6 Interrogation du treillis $\underline{\mathfrak{B}}_2(G, M, W, I)$ avec la requête $Y = \{MeSH_{|\{Trans.fact.\}}, NCBI_{|\{Vertebrates\}}\}$. 125
6.7 Treillis de concepts multivalués, $\underline{\mathfrak{B}}_1(G, M, W, I)$, obtenu pour un seuil $S = 1$ à partir du contexte symbolique donné à la table 5.6. 127
6.8 Interrogation du treillis $\underline{\mathfrak{B}}_1(G, M, W, I)$ avec la requête $Y = \{MeSH_{|\{Trans.fact.\}}, NCBI_{|\{Vertebrates\}}\}$. 128

7.1 Un treillis de concepts multivalués correspondant au contexte multivalué formé par 48 sources de données dans BioRegistry pour un seuil $S = 0.5$. 133
7.2 Un treillis de concepts multivalués correspondant au contexte multivalué formé par 48 sources de données dans BioRegistry pour un seuil $S = 0.1$. 134

A.1 Visualisation du treillis de concepts correspondant au contexte formel extrait de BioRegistry. 151
A.2 Interface de définition de requêtes de BR-Explorer. 152
A.3 Visualisation du résultat de l'exécution de BR-Explorer sur la requête formée par l'ensemble de métadonnées {Drosophila, Eugenes, HCV}. 153
A.4 Visualisation du voisinage (à une distance égale à 1) de la requête formée par l'ensemble de métadonnées {Drosophila, Eugenes, HCV}. 154
A.5 Définition de hiérarchies d'attributs dans BR-Explorer. 155

Liste des tableaux

3.1 Un contexte formel représentant les planètes du système solaire. 26
3.2 Contexte formel représentant les planètes du système solaire avec renommage des attributs et des objets. 30
3.3 Les ensembles de rectangles maximaux calculés par l'algorithme de Chein à partir du contexte formel donné dans le tableau 3.2. Les rectangles barrés correspondent à des rectangles non maximaux. Les rectangles qui les contiennent sont indiqués entre parenthèses et l'iteration durant laquelle ils ont été calculés est donnée par le nombre de *. Par exemple, "~~1 × adg~~ **(12 × adg)" est interprété comme suit : le rectangle "1 × adg" n'est pas maximal, il est remplacé par "12 × adg" retrouvé à la deuxième itération de l'algorithme. 30
3.4 Découpage du contexte formel en deux parties. 33
3.5 Mesures réelles relatives aux planètes du système solaire. 40
3.6 Échelle conceptuelle de l'attribut "*Distance au soleil*" et le treillis correspondant. 41
3.7 Échelle conceptuelle de l'attribut "*Diamètre*" et le treillis correspondant. 41
3.8 Échelle conceptuelle de l'attribut "*Satellite*" et le treillis correspondant. 41
3.9 Contexte monovalué résultant de l'échelonnage conceptuel du contexte multivalué des planètes du système solaire. 42
3.10 Un exemple de contexte flou et le treillis de concepts flous correspondant selon la méthode de Ben Yahia et Jaoua. 45
3.11 L-contexte donné par les planètes et leurs propriétés. 47
3.12 Un exemple de contexte logique (gauche) et le treillis de concepts logiques correspondant (droite). Les lettres h, f et c sont les abbreviations respectives de *homme*, $femme$ et *chauve*. 48
3.13 Le contexte formel et du treillis correspondant donné dans [Carpineto and Romano, 2004a]. 49
3.14 Contexte formel enrichi par la taxonomie de la figure 3.7 et le treillis correspondant 50
3.15 Contextes Monovalués 52
3.16 Relation : A_Visité 52
3.17 Exemple de tableau de données multivaluées 54

4.1 Un contexte multivalué correspondant à une partie des données de BioRegistry . 56
4.2 Contexte formel représentant une partie du corpus BioRegistry. 57
4.3 Ajout de la requête $Q = (X, \{Mammals, Complete, Updated\})$ au contexte $\mathbb{K} = (G, M, I)$. 65
4.4 Un contexte formel représentant une partie de BioRegistry. 69
4.5 Table de vérité des expressions "$m_1 \sqcap m_2$", "$m_1 \sqcup m_2$" et "$m_1 \succ m_2$" en fonction de la présence de m_1 et m_2. L'expression est vraie lorsque la case correspondante contient "×". 82

5.1 Mesures réelles des caractéristiques des planètes du système solaire 89
5.2 Tableau des planètes du système solaire 91
5.3 Tableau des quatres premières planètes du système solaire. 91
5.4 Contexte formel résultat de l'échelonnage plan du contexte multivalué donné à la table 5.3. 98
5.5 Contexte formel résultat de l'échelonnage du contexte multivalué donné à la table 5.3 en remplaçant les valeurs par "×". 99
5.6 Un exemple de contexte symbolique représentant un extrait des données de Bio-Registry. 100
5.7 Le nombre de concepts multivalués extraits à partir du contexte donné dans la table 5.6 en fonction du seuil de similarité S. 108
5.8 Un extrait des mesures (approximatives) des caractéristiques relatives au pesticide appelé *glyphosate*. Ces mesures sont données par différentes sources qui ont étudié cette matière active. La dernière ligne du tableau représente les valeurs données par les experts de l'INRA Colmar. DT50 : *temps de demi vie au champ*, Aquatox : *toxicité* et DJA : *dose journalière admise*. 109
5.9 Un exemple de contexte plurivalué symbolique représentant une partie des données de BioRegistry 111

6.1 Le contexte multivalué réduit aux objets du concept multivalué ({*Exint, SpliceDB, TRANSCompel*}, $\{NCBI_{|\{Vertebarates,Eukaryotes,Mammals\}}, NAR_{|\{1.2\}}\}$). 116

Chapitre 1

Introduction générale

Sommaire

1.1 Contexte de travail 1
1.1.1 Les ressources Web 1
1.1.2 La découverte de ressources biologiques sur le Web 4
1.2 Objectif de la thèse et principales contributions 6
1.3 Organisation du manuscrit 7

1.1 Contexte de travail

1.1.1 Les ressources Web

État et évolution

Dans les standards du Web, une ressource est toute entité pouvant être identifiée par un URI (Uniform Resource Identifier) [1] ou par un IRI (Internationalized Resource Identifier) [2]. Des exemples de ressources Web peuvent être des documents électroniques, des images, du son, des bogues, des wikis, des services, etc. Grâce à cette définition, générale et ouverte, le Web s'est transformé en un espace de partage et d'échange d'information où chacun est capable de faire part de l'information dont il dispose sans se soucier des normes ni des contraintes de conformité à un modèle ou à un format préalablement définis. Parallèlement à l'apparition du Web au début des années 90, le monde était en train de subir une "informatisation" à large échelle. Ce changement était supporté par le développement électronique en plein essor donnant des solutions de plus en plus efficaces quant au stockage et à la gestion des données. Tous ces facteurs réunis ont poussé à une véritable "explosion" du Web. En octobre 2008, le nombre de sites Web accessibles est de 182 226 259 contre 130 en juin 1993 et 19 823 296 en août 2000. L'évolution du nombre de sites accessibles sur le Web entre 1995 et 2008 est donnée dans Figure 1.1 [3]. Étant donné qu'un site Web peut être composé de plusieurs ressources Web (entre 250 et 300 en moyenne [4]), le nombre de ressources Web est estimé aujourd'hui à plus de 50 milliards. Les efforts collectifs et totalement indépendants de conception, de création et de déploiement des ressources Web à

1. http ://tools.ietf.org/html/rfc3986
2. http ://tools.ietf.org/html/rfc3987
3. L'étude est faite par NETCRAFT : http ://news.netcraft.com/archives/2008/10/29/october_2008_web_server_survey.html
4. http ://www.boutell.com/newfaq/misc/sizeofweb.html

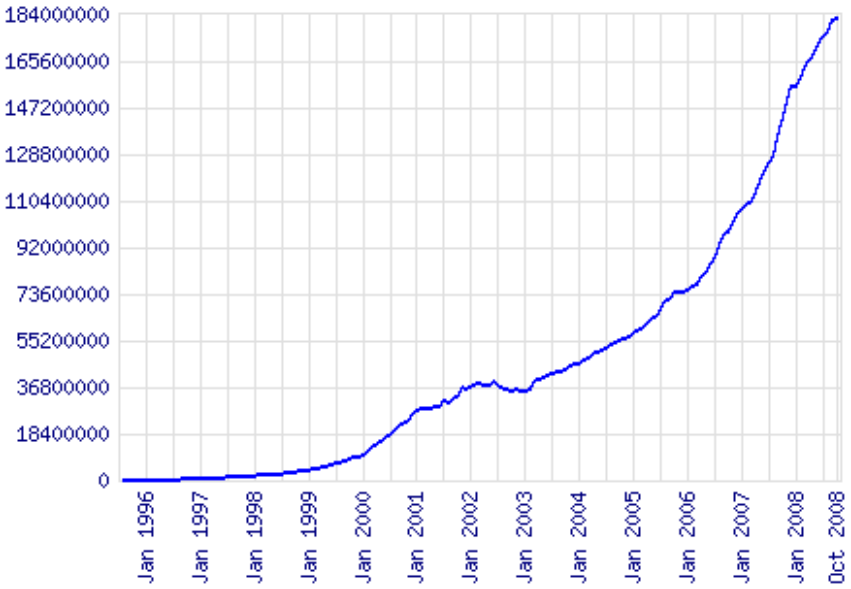


FIGURE 1.1 – L'évolution du nombre des sites Web entre août 1995 et octobre 2008 : étude faite par NETCRAFT.

l'échelle mondiale sont à l'origine de l'hétérogénéité de ces ressources à tous les niveaux (structure, contenu, mode d'exploitation, etc.). Cette hétérogénéité a fait du Web une source d'information d'une richesse inégalable [5].

Cette situation bien qu'avantageuse, dans le sens où beaucoup d'informations dont on a besoin sont accessibles et prêtes à être exploitées, a intensifié les difficultés liées à l'identification et à l'exploitation des ressources Web. En effet, face à une source d'information de telle envergure, les centaines de millions d'utilisateurs qui l'exploitent tous les jours ont besoin de disposer de moyens efficaces qui assurent le meilleur appariement entre leurs besoins et les ressources susceptibles de répondre à ces besoins. Cet appariement est connu depuis l'apparition du Web sous le nom de découverte de ressources pertinentes.

Découverte de ressources sur le Web

D'une manière générale, la découverte de ressources consiste à identifier de manière efficace les ressources satisfaisant un ensemble de critères préalablement définis. Elle recouvre essentiellement la Recherche d'Information (RI) au sens large, la Recherche de Documents, la découverte de services Web, la découverte de ressources ayant un format particulier telles que les ressources multimedia (image, son, vidéo, etc) ou encore la découverte de ressources dans un domaine particulier tel que la biologie. Les critères préalablement définis, appelés *requêtes*, se ramènent à une description de la ressource "idéale" à identifier. Les ressources pertinentes pour une requête donnée sont celles dont la description est la plus proche des critères spécifiés dans la requête. Ainsi, le processus de découverte de ressources n'est autre que le rapprochement entre une description de la ressource "idéale" à identifier et un ensemble de descriptions des ressources disponibles sur le Web. Ce rapprochement nécessite donc de disposer de l'ensemble des descriptions des ressources

5. "*The web is becoming a universal repository of human knowledge and culture which has allowed unprecedent sharing of ideas and information in a scale never seen before [Baeza-Yates and Ribeiro-Neto, 1999]*"

disponibles. De ce fait, la mise en place d'une approche de découverte de ressources se ramène principalement aux trois étapes suivantes [Lichtner, 1994, Yuwono et al., 1995].

1. La description des ressources : cette étape, connue sous le nom "*indexation*" ou "*annotation*" des ressources, consiste à associer à chaque ressource une fiche descriptive contenant un certain nombre d'informations qui reflète au mieux le contenu de la ressource et qui a pour rôle de faciliter son identification par la suite. Ces fiches peuvent être créées manuellement ou, ce qui est souvent le cas, automatiquement par l'intermédiaire de logiciels appelés spiders, robots, ou crawlers. Selon l'approche considérée, l'ensemble des fiches descriptives est appelé corpus de recherche, annuaire ou catalogue.
2. L'identification des ressources pertinentes : cette étape constitue la découverte de ressources proprement dite. Elle peut être effectuée de deux manières possibles : par interrogation ou par navigation. La découverte par interrogation consiste à formuler une requête (critères de la découverte), effectuer l'appariement entre la requête et les descriptions des ressources disponibles et retourner en réponse un ensemble de ressources classées selon leur pertinence. La découverte par navigation consiste à parcourir l'ensemble des descriptions des ressources afin d'identifier les ressources pertinentes. Ce mode est motivé par le fait qu'il est "plus facile, pour un être humain, de reconnaître une ressource pertinente que de la décrire" par l'intermédiaire d'une requête. La navigation peut être facilitée par la classification des descriptions des ressources dans des structures hiérarchiques.
3. L'interaction avec les ressources pertinentes : cette étape consiste à accéder aux ressources effectives -et non pas à leurs fiches descriptives- afin d'exploiter leurs contenus. Dans la plupart des approches de découverte de ressources, cette étape est réduite à la présentation du chemin d'accès des ressources pertinentes aux utilisateurs. Dans ce cas, l'interaction effective avec les ressources est une étape qui intervient une fois que le processus de découverte de ressources s'est déroulé avec succès.

Le défi principal auquel les approches de découverte de ressources doivent faire face aujourd'hui est double. D'une part, l'espace de recherche à gérer est d'une échelle sans précédent et qui ne cesse de croître. D'autre part, les millions d'utilisateurs expriment des demandes en information très variées et leurs exigences ont beaucoup évolué de "*donnez-moi ce que je dis*" à "*donnez-moi ce dont j'ai besoin*"[6] [Belkin, 2008]. Pour répondre à ces exigences, les approches de découverte de ressources doivent disposer de moyens pour "comprendre" les besoins exprimés à travers les requêtes pour être en mesure de les orienter vers les ressources convenables. Ces moyens reposent sur les connaissances de domaine et cela aboutit à la découverte de ressources guidée par des connaissances.

Découverte de ressources guidée par des connaissances

La découverte de ressources guidée par des connaissances de domaine consiste à prendre en compte l'aspect sémantique dans le processus de découverte de ressources afin d'améliorer ses résultats. Il s'agit de comprendre le besoin exprimé dans une requête et de l'orienter vers les ressources les plus à mêmes de répondre à ce besoin. Cela nécessite d'associer une sémantique à la fois aux requêtes et aux descriptions des ressources en s'appuyant sur des connaissances partagées qui forment un référentiel commun permettant d'établir le lien entre les deux sémantiques. Les connaissances partagées se présentent sous la forme de ressources sémantiques telles que les ontologies de domaine, les taxonomies, les thésaurus, etc. Ces ressources sémantiques ont considérablement évolué en nombre et en qualité depuis l'apparition du

6. user expectations have expanded considerably, moving from "give me what I said" to "give me what I want"

Web sémantique [Berners-Lee et al., 2001]. Suite à cette évolution, de plus en plus d'approches de découverte de ressources sont conçues en tenant compte de l'aspect sémantique à différents niveaux [Pisanelli et al., 2002, Baziz, 2005]. Tout d'abord lors de la création (ou préparation) du corpus de la découverte, les ressources sont décrites en se référant à des ontologies de domaine. A chaque ressource est associée une description qui renvoie à des concepts d'une ou de plusieurs ontologies [Guarino et al., 1999, Baziz et al., 2006]. Ensuite, les requêtes sont analysées, en s'appuyant sur les mêmes ontologies, afin de leur associer aussi des concepts d'ontologies [Klapaftis and Manandhar, 2005, Singhal, 2008]. L'appariement se fait par la suite grâce à des méthodes de calcul de proximité entre concepts dans une ontologie [Hliaoutakis et al., 2006].

Le scénario idéal serait de disposer d'une ontologie de domaine universelle et suffisamment riche pour couvrir toutes les ressources existantes. Cependant, le nombre de ressources Web et leur hétérogénéité, d'une part, et la complexité et la diversité des données qu'elles peuvent contenir, d'autre part, font qu'une telle ontologie est impossible à mettre en place. L'illustration est donnée par le projet Cyc [7] [Lenat, 1995, Matuszek et al., 2006] lancé depuis 1984 avec l'objectif de développer une ontologie globale et une base de connaissances générale et toujours loins de ses ambitions initiales. De ce fait, on distingue deux familles de ressources sémantiques utilisées dans les approches de découverte de ressources. La première famille est constituée des ressources sémantiques générales qui ont pour objectif de couvrir le maximum de ressources du Web sans aller dans les détails des domaines particuliers. L'exemple le plus connu de cette famille est WordNet [8]. La deuxième famille est constituée des ressources sémantiques spécifiques à des domaines particuliers tels que la médecine (UMLS [9]), la biologie (MeSH thesaurus [10], NCBI taxonomy [11] et GO [12]). Cette répartition des ressources sémantiques conduit à la répartition des approches de découverte de ressources guidée par des connaissances en approches générales et approches spécifiques.

Dans le cas où l'intérêt porte sur des ressources relatives à un domaine particulier tel que la biologie, de nouvelles difficultés dues à la complexité du domaine posent des défis supplémentaires aux approches de découverte de ressources. Il convient donc d'adopter et à défaut de définir des approches qui prennent en compte la particularité du domaine. Dans la suite, nous nous intéressons à la découverte de ressources biologiques sur le Web.

1.1.2 La découverte de ressources biologiques sur le Web

Caractéristiques et évolution des ressources biologiques sur le Web

Les ressources biologiques sur le Web sont les ressources Web ayant un contenu relatif au domaine biologique. Ces ressources sont de différents types incluant de simples documents textuels, des articles scientifiques comportant des résultats de recherche, des bases de données contenant d'importants volumes de données. Les données volumineuses stockées dans certaines ressources biologiques peuvent être des résultats d'expérimentations ou d'études cliniques, des descriptions d'objets biologiques complexes tels que les gènes, etc. A l'image des ressources du Web, ces ressources particulières, que nous appelons "*sources de données biologiques*", ont considérablement évolué depuis une dizaine d'années. Leur nombre est passé d'environ 200 en 1999 à près de 1200

7. http ://www.cyc.com/cyc
8. http ://wordnet.princeton.edu/
9. Unified Medical Language System http ://umlsinfo.nlm.nih.gov/
10. Medical Subject Headings thesaurus http ://www.nlm.nih.gov/mesh/meshhome.html
11. National Center for Biotechnology Information http ://www.ncbi.nlm.nih.gov/
12. Gene Ontology http ://www.geneontology.org/

en 2009. L'évolution chronologique du nombre de sources de données biologiques disponible sur le Web est schématisée dans Figure 1.2.

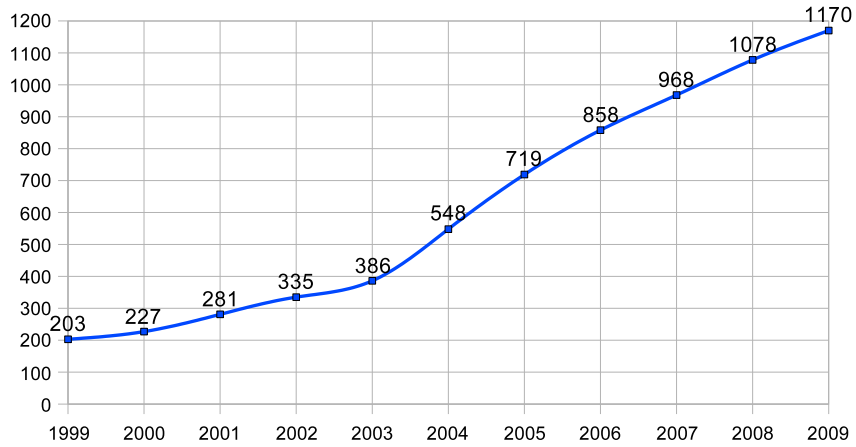


FIGURE 1.2 – L'évolution du nombre des sources de données biologiques disponibles sur le Web entre 1999 et 2009. Cette étude est effectuée en s'appuyant sur les recensements annuels des sources de données disponibles effectués par la revue Nucleic Acids Research

L'évolution des sources de données biologiques accessibles sur le Web est la conséquence de l'apparition de la bioinformatique qui a fédéré les efforts des biologistes et des informaticiens. En effet, parallèlement aux expérimentations biologiques de plus en plus avancées, des solutions informatiques de plus en plus efficaces sont proposées dans le but de faciliter la gestion, le stockage des données manipulées et l'interprétation des résultats obtenus. Cependant, en l'absence de standards de description et d'un schéma unique à ces sources, l'identification et la localisation de sources susceptibles de contenir des données précises demandées par un utilisateur, notamment un biologiste, n'est pas une tâche aisée, d'où la sous-exploitation de ces données. Ce problème est causé par plusieurs facteurs dont les deux principaux sont les suivants.

1. Nature des objets biologiques : les données contenues dans les sources décrivent des objets biologiques complexes ainsi que leurs relations très variées.
2. Développement non concerté des sources de données : les sources de données biologiques sont développées indépendamment par différents auteurs dans différents contextes. De ce fait, elles sont caractérisées par une hétérogénéité à plusieurs niveaux incluant le format, la structuration des données, le mode d'exploitation, etc.

Défis de la découverte des sources de données biologiques sur le Web

Les difficultés de la découverte de sources de données biologiques se manifestent à toutes les étapes du processus de découverte. La première difficulté est rencontrée dès la mise en place de l'espace de la découverte (corpus). En effet, les sources sont souvent structurées sous la forme de bases de données ou d'entrepôts de données. Les données structurées sous cette forme sont difficilement accessibles (voire même inaccessibles) aux outils d'indexation automatique qui ont pour rôle de créer les fiches descriptives des sources à partir des données qu'elles contiennent en vue de faciliter leur identification par la suite. Par conséquent, le corpus obtenu n'est pas assez riche pour permettre l'identification des sources pertinentes. A ce niveau, l'obtention d'un corpus riche et complet nécessite d'abord la mise en place de modèles spécifiques à la description des

sources de données biologiques conçus et validés par des experts du domaine et conformes aux standards d'annotation et ensuite, la description des contenus des sources conformément à ces modèles en se référant à des ressources sémantiques.

La deuxième difficulté se rencontre lors de la formulation du besoin sous la forme d'une requête du fait de la complexité du domaine. Il y a deux manières de faire face à cette difficulté. La première façon consiste à proposer différentes formes d'assistance telle que la reformulation et l'enrichissement des requêtes, l'expression des préférences, etc. La deuxième façon consiste à proposer les moyens d'effectuer une découverte progressive telle que la navigation dans le corpus décrivant l'ensemble des sources [Baeza-Yates and Ribeiro-Neto, 1999]. Dans ce cas, une classification conceptuelle facilitant cette découverte doit être préalablement effectuée.

La troisième difficulté est rencontrée au niveau de l'identification des sources de données capables d'apporter des réponses satisfaisantes aux besoins exprimés. Selon les choix effectués dans les étapes précédentes, des mesures de pertinence doivent être définies.

1.2 Objectif de la thèse et principales contributions

Les travaux de recherche effectués dans le cadre de cette thèse portent sur l'exploitation des connaissances de domaine dans un processus de découverte de ressources biologiques sur le Web. L'objectif principal est de faciliter l'identification des ressources pertinentes en vue d'une exploitation ultérieure des données qu'elles contiennent. Afin d'atteindre cet objectif, la solution proposée devra être capable de surmonter les défis posés par la découverte de ressources biologiques sur le Web.

Dans cette thèse, nous proposons une approche de découverte de ressources guidée par des connaissances de domaine qui tient compte de la particularité des sources de données biologiques sur le Web tout en restant générique et adaptable à d'autres contextes. Cette approche considère les connaissances de domaine de différentes manières à la fois indépendantes et complémentaires durant toutes les étapes du processus de découverte de ressources. Le point de départ est un annuaire de sources de données biologiques, BioRegistry, qui fournit une annotation sémantique des contenus des sources en faisant référence aux ressources sémantiques utilisées. Ceci correspond à l'étape de description des ressources dans le processus de découverte de ressources. Notre contribution peut être répartie en trois parties.

Utilisation des treillis de concepts pour la classification et la découverte des sources de données Il s'agit d'abord de s'appuyer sur la théorie de l'Analyse de Concepts Formels (ACF) pour organiser l'ensemble des sources de données en fonction de leurs annotations dans l'annuaire BioRegistry. La structure obtenue est appelée treillis de concepts formels et facilite la découverte des sources de données pertinentes. La découverte peut être effectuée soit de manière progressive en naviguant dans le treillis, soit de manière directe en interrogeant le treillis de concepts par l'intermédiaire d'une requête, soit en combinant les deux modes (navigation et interrogation).

Utilisation des connaissances de domaine pour guider le processus de découverte Les connaissances de domaine sont utilisées pour améliorer les performances du processus de la découverte de deux manières. La première consiste à effectuer l'enrichissement sémantique de requêtes pour qu'elles soient plus cohérentes avec les annotations des sources de données et permettent par la suite une meilleure identification des sources pertinentes. La deuxième consiste à exprimer des préférences qui vont guider le processus de l'identification des sources pertinentes par navigation ou par interrogation.

Utilisation de connaissances de domaine pour guider la construction du treillis de concepts Il s'agit d'étendre l'ACF pour pouvoir prendre en compte la complexité du domaine biologique et fournir une meilleure classification des sources de données biologiques en s'appuyant sur la sémantique de leurs contenus. Nous définissons d'abord les treillis de concepts multivalués et la manière dont les connaissances de domaine sont utilisées pour les construire. Ensuite, nous exploitons leur potentiel comme support de découverte de ressources guidée par des connaissances de domaine.

1.3 Organisation du manuscrit

La suite de ce mémoire est organisée comme suit. Le chapitre 2 contient une description détaillée de l'annuaire de sources de données biologiques BioRegistry, les étapes de sa mise en place et des éléments sur son exploitation. Le chapitre 3 contient un état de l'art sur l'ACF, ses extensions et ses principales applications à la découverte de ressources et à la découverte de connaissances. Le chapitre 4 détaille l'approche de découverte de ressources que nous proposons. Le chapitre 5 présente l'extension de l'ACF aux données complexes. Le chapitre 6 contient l'application de cette extension à la découverte de ressources guidée par des connaissances et illustre ses principaux apports. Le chapitre 7 présente les implémentations résultant de la mise en oeuvre de l'approche proposée de découverte de ressources dans le domaine de la biologie. Finalement, le chapitre 8 conclut ce travail et donne les principales perspectives.

Chapitre 2

Contexte de l'étude : découverte de ressources biologiques sur le Web

Sommaire

2.1 Introduction 9
2.2 Description de l'existant 10
2.2.1 Principales caractéristiques des sources de données biologiques 10
2.2.2 Méthodes existantes pour l'identification et l'exploitation des sources de données biologiques 10
2.2.3 Limites 13
2.3 BioRegistry : un annuaire sémantique de sources de données biologiques 13
2.3.1 Prise en compte des connaissances de domaine dans la modélisation de l'annuaire BioRegistry 13
2.3.2 Peuplement de l'annuaire BioRegistry 16
2.3.3 Interrogation de l'annuaire BioRegistry 17
2.4 Découverte de sources de données biologiques "guidée par des connaissances" 18

2.1 Introduction

L'un des défis de la bioinformatique aujourd'hui est de permettre aux biologistes d'accéder efficacement aux données gisant dans les centaines de sources de données réparties à travers le monde. Le grand nombre de sources, leur hétérogénéité et la complexité des objets biologiques auxquels elles font référence rendent souvent difficile la mise en relation d'une requête avec la source appropriée. Dans ce chapitre nous décrivons l'annuaire de sources de données biologiques BioRegistry qui constitue le point de départ de l'approche de découverte de ressources développée dans le cadre de cette thèse.

La suite du chapitre est organisée comme suit. Dans la section 2.2, nous revenons sur les principales caractéristiques des sources de données biologiques disponibles sur le web et nous dressons un bref état de l'art des approches proposées pour l'identification et l'accès à ces sources. Dans la section 2.3, nous présentons les étapes de la mise en place de BioRegistry et nous discutons les choix effectués et leur impact sur la méthode de découverte de ressources à mettre en place.

Finalement, dans la section 2.4, nous décrivons l'architecture globale de l'approche de découverte de ressources bâtie autour de BioRegistry.

2.2 Description de l'existant

2.2.1 Principales caractéristiques des sources de données biologiques

Les caractéristiques principales des quelques 1200 sources de données biologiques disponibles sur le web [Galperin and Cochrane, 2009, Hernandez and Kambhampati, 2004] sont les suivantes :

- Les sources contiennent des données pouvant être de nature très diverse. Cette diversité est due à la complexité du domaine biologique et à l'existence de plusieurs sous-domaines à la fois spécifiques et évolués.
- Les données contenues dans les sources sont codées sous différents modèles et formats, souvent hétérogènes. Cette hétérogénéité est due à la mise en place et l'évolution indépendante de chaque source.
- Les sources de données sont autonomes et leur mise en ligne se fait de façon indépendante. De ce fait, chaque source peut évoluer de manière indépendante. L'évolution d'une source peut varier des simples mises-à-jour de son contenu au changement complet de son schéma, de la structuration et du format des données qu'elle contient, etc.
- Les modes et les capacités d'interrogation offertes par les interfaces des sources de données sont différents suivant les sources.

Ces caractéristiques illustrent l'hétérogénéité des sources de données biologiques à tous les niveaux. Cette hétérogénéité soulève différents problèmes qui compliquent l'identification et l'accès aux sources et par suite l'exploitation des données qu'elles contiennent.

2.2.2 Méthodes existantes pour l'identification et l'exploitation des sources de données biologiques

Afin d'assurer une meilleure exploitation du contenu des sources de données, il est indispensable de pouvoir les identifier et de les associer aux requêtes qui portent sur leur contenu. De ce fait, le problème de découverte des sources pertinentes ne se posait pas en tant que tel mais plutôt en tant qu'une phase d'orientation d'une (partie de) requête vers une source choisie -de manière automatique- parmi un ensemble réduit de sources. Dans ce cadre, de nombreux efforts ont été consacrés jusqu'à présent aux problèmes posés par la standardisation de l'accès à ces sources, le traitement des requêtes en vue de leur distribution (répartition) sur les sources pertinentes, l'intégration des réponses, etc. Ces tâches nécessitent d'une part de concevoir des scénarios appropriés qui tiennent compte des spécificités des schémas des sources, des formats des données qu'elles contiennent, de leur modes d'interrogation, etc., et d'autre part, de disposer d'une interopérabilité entre les sources. Ces efforts ont abouti à la mise en place de différentes approches dites "*approches d'intégration ou de médiation*" ayant pour objectif d'unifier l'accès à diverses sources de données -préalablement choisies- afin d'exploiter leurs contenus et d'apporter des réponses aux requêtes posées.

L'évolution importante des sources de données biologiques durant les dernières années s'ajoute aux difficultés liées à l'hétérogénéité des sources et à la complexité de leur contenu, dépassent ainsi rapidement les capacités de fédération de cette première famille d'approches (les systèmes d'intégration ou de médiation). Dans cette nouvelle situation, l'identification des sources pertinentes est passée du stade de simple étape au sein du processus d'exploitation d'un ensemble

réduit de sources au stade de processus indépendant qui précède l'exploitation des sources et l'oriente vers la (les) source(s) à exploiter parmi la totalité des sources accessibles sur le Web. Ce fonctionnement caractérise une deuxième famille d'approches pour l'identification et l'accès aux sources pertinentes. Cette famille inclue les catalogues et les portails.

Un bref état de l'art des principaux systèmes de ces deux familles est donné dans les sections suivantes.

Les systèmes d'intégration et de médiation

L'objectif principal d'un système d'intégration est de permettre d'accéder à des informations contenues dans des sources de données biologiques de manière simple et automatisée [Mougin, 2006]. Le fonctionnement général d'un système d'intégration pour une requête donnée suit les étapes suivantes :

- identification des sources de données pertinentes pour la requête ;
- accès aux sources pertinentes de façon centralisée et transparente pour les utilisateurs ;
- agrégation automatique des résultats obtenus à partir des différentes sources afin de fournir une réponse globale et homogène.

Les systèmes d'intégration incluent principalement les entrepôts de données, les systèmes de médiation et les systèmes d'intégration navigationnelle. Un état de l'art détaillé de ces approches est donné dans [Mougin, 2006]. Ici nous nous contentons de rappeler les principes de ces trois types d'approches et de citer les principaux systèmes représentatifs de chaque type dans le domaine qui nous intéresse, celui des données biologiques.

Les entrepôts de données. Ces systèmes regroupent des données extraites de multiples sources dans un entrepôt centralisé [Widom, 1995]. Des adaptateurs sont définis pour chaque source afin de garantir la cohérence de le schéma de la source avec le schéma global de l'entrepôt. Ensuite, les données des sources sont entièrement ou en partie stockées dans l'entrepôt et les utilisateurs peuvent soumettre des requêtes directement sur ces données locales au lieu d'interroger chaque source indépendamment. Parmi les systèmes implémentés sous la forme d'entrepôt de données nous pouvons citer GUS [Davidson et al., 2001] et Biozon [Birkland and Yona, 2006]. GUS (Genomics Unified Schema) est défini comme une plateforme de bases de données génomiques, incluant les sources GenBank [13] et SwissProt [14] et les termes GO (Gene Ontology). Biozon est un entrepôt de données qui intègre des données récoltées dans une vingtaine de sources différentes, telles que Swiss-Prot et KEGG [15].

Les systèmes de médiation. Ils réalisent une intégration virtuelle au moyen du médiateur qui comporte un schéma global des différentes sources. Le médiateur interagit avec celles-ci au travers d'adaptateurs qui se chargent d'interroger les sources et de rendre le résultat au médiateur qui s'occupe de fusionner les différentes réponses de manière homogène et globale [Rousset et al., 2002]. Parmi les systèmes de médiation existants nous pouvons citer TAMBIS [Goble et al., 2001] et SEMEDA [Kohler et al., 2003]. TAMBIS (Transparent Access to Multiple Bioinformatics Information Sources) est un système de médiation qui intègre 15 sources de données dont SwissProt. Le système de médiation SEMEDA (SEmantic MEtaDAtabase) intègre 6 sources de données dont SwissProt et KEGG.

13. http ://www.ncbi.nlm.nih.gov/Genbank/
14. http ://www.expasy.org/sprot/
15. http ://www.genome.jp/kegg/

Les systèmes d'intégration navigationnelle. Ils permettent une intégration virtuelle et n'offrent pas de schéma global mais uniquement un langage de requêtes unique pour accéder à différentes sources distribuées. Un système d'intégration navigationnelle s'appuie sur un ensemble de descriptions des sources qu'il inclut et sur un générateur de chemins indiquant l'ordre d'accès à ces sources. Le fonctionnement de ce type de système consiste à transformer la requête en chemins qui peuvent chacun mener à des résultats en exploitant les références croisées définies entre sources et les liens internes. Les principaux systèmes existants sont SRS [Etzold and Argos, 1993] et NEntrez [Schuler et al., 1996]. SRS (Sequence Retrieval System) unifie l'accès à plus de 140 sources de données spécifiques à la biologie moléculaire incluant SwissProt et PubMed [16] et permet de parcourir et d'exploiter les références croisées entre ces sources de données. NEntrez unifie l'accès à toutes les sources du NCBI : GenBank [17], OMIM [18], etc.

Les portails et les catalogues

Les portails thématiques et les catalogues (ou annuaires) de sources de données biologiques constituent la deuxième famille d'approches qui s'intéressent à l'identification des sources pertinentes sans assurer les moyens de les interroger.

Les portails sont des sites Web qui proposent une indexation thématique d'un ensemble de sources de données. Leur fonctionnement peut varier du simple parcours de la liste des sources qu'ils intègrent afin d'identifier celles qui doivent être interrogées à des méthodes d'identification qui s'appuient sur une description préalable des sources. L'un des portails les plus connus en biologie est ExPAZy [19] qui référence plus d'un millier de documents et de sources de données biologiques.

Les catalogues de sources de données biologiques constituent une solution plus complète et plus riche que les portails. En effet, les catalogues répertorient souvent un grand nombre de sources de données décrites de manière détaillée par l'intermédiaire d'un ensemble de métadonnées. Ces métadonnées sont par la suite utilisées pour l'identification des sources pertinentes pour répondre à une requête donnée. Ainsi, la richesse et la bonne structuration de ses métadonnées dans un catalogue est une condition nécessaire pour assurer et faciliter un appariement efficace entre les requêtes et les sources pertinentes. En plus de leur autonomie en tant que systèmes de découverte de sources de données, les catalogues peuvent venir compléter les systèmes d'intégration dans la phase d'identification des sources pertinentes à interroger.

Le catalogue le plus complet et le mieux maintenu aujourd'hui est sans doute le catalogue du NAR [Galperin and Cochrane, 2009]. Dans sa dernière mise-à-jour, en janvier 2009, ce catalogue recense près de 1200 sources. Chaque source de données référencée dans le catalogue NAR est décrite par son titre, son url, une courte description textuelle et sa catégorie NAR. Bien que la catégorisation des sources facilite la navigation dans le catalogue, l'interrogation du catalogue reste limitée aux simples recherches textuelles sur le titre et la description des sources.

Afin d'améliorer les performances dans l'identification des sources, certains catalogues s'appuient sur des descriptions plus détaillées des sources à l'aide de métadonnées décrivant davantage les sources et leurs contenus. Le catalogue le plus connu de cette catégorie était DB-

16. http ://www.ncbi.nlm.nih.gov/pubmed/
17. http ://www.ncbi.nlm.nih.gov/Genbank/
18. http ://www.ncbi.nlm.nih.gov/omim/
19. http ://www.expasy.ch/

CAT [Discala et al., 2000]. Il contenait 509 sources (au 1^{er} mars 2004) décrites par leurs noms, domaines, descriptions, auteurs, les références qu'elles contiennent, etc. Contrairement au catalog NAR qui est bien maintenu et où la validité des sources est testée toutes les semaines, DBCAT a été abandonné suite à des problèmes de maintenance.

2.2.3 Limites

Malgré les nombreux efforts effectués dans le but de garantir une couverture suffisante et une meilleure exploitation des sources de données biologiques, les approches proposées présentent encore de nombreuses limites notamment en ce qui concerne l'identification des sources pertinentes. Contraints par les difficultés de définir un schéma global et un mode d'accès unique à des sources hétérogènes et des contenus très variés, les systèmes d'intégration disponibles ne prennent en compte qu'un nombre réduit de sources de données. En conséquence, les techniques développées au sein de ces systèmes pour la répartition des requêtes sur les sources qu'ils intègrent telles que l'analyse sémantique des requêtes et des réponses ne sont pas généralisable à l'échelle de toutes les sources du Web.

D'un autre coté, les portails et catalogues existants, qui font référence à un nombre plus élevé de sources comparés aux systèmes d'intégration, ne disposent pas des moyens nécessaires à l'identification efficace des sources pertinentes. Ces systèmes sont handicapés par l'absence d'une indexation à la fois riche et à jour des sources de données qu'ils référencent.

Pour aboutir à une approche de découverte de ressources qui soit réellement efficace, il est indispensable de combiner les avantages des systèmes existants. Plus précisément, une telle approche doit pouvoir :

- faire référence à toutes les sources disponibles,
- annoter les sources à l'aide de métadonnées qui décrivent le contenu des sources de manière exhaustive et sémantique,
- assurer des techniques efficaces pour l'appariement entre les requêtes et les sources pertinentes en exploitant les connaissances disponibles,
- assurer la mise à jour régulière de la liste des sources référencées et des métadonnées qui les décrivent afin de prendre en compte l'apparition de nouvelles sources (ou disparition de sources existantes) et l'évolution du contenu des autres.

Ces caractéristiques essentielles pour une approche réussie et efficace ont motivé, inspiré et orienté les choix pris lors de la conception et mise en place de l'annuaire de sources de données biologique BioRegistry.

2.3 BioRegistry : un annuaire sémantique de sources de données biologiques

2.3.1 Prise en compte des connaissances de domaine dans la modélisation de l'annuaire BioRegistry

Le modèle de méta-données de BioRegistry

Les méta-données décrivent le contenu, la qualité et d'autres caractéristiques relatives à des données. Elle jouent un rôle essentiel dans les tâches d'indexation et d'interrogation d'ensembles de données. En 1995, un comité international d'experts a proposé un modèle standard de description de méta-données relatives aux ressources du Web : le Dublin Core Metadata Initiative (DCMI) [Dekkers and Weibel, 2003]. Ce standard est composé d'une quinzaine d'éléments :

identifier, title, creator, subject, description, publisher, contributor, date, type, format, language, source, rights, coverage, relation.

Bien que le modèle de méta-données du DCMI doive rester très simple, il inclut deux mécanismes permettant d'être plus précis dans la description : (i) des raffinements d'éléments ("element refinements") tels que "*created*" et "*modified*" qui raffinent l'élément "*date*" pour exprimer une date de création ou de modification ; (ii) des schémas d'encodage tels que "*vocabulary encoding schemes*" permettant de spécifier qu'un terme est extrait d'un vocabulaire contrôlé ou "*syntax encoding schemes*" permettant de spécifier qu'une valeur est formatée selon certaines règles (par exemple une date dans le format W3CDTF : YYYY-MM-DD).

Le modèle des méta-données du DCMI étant très général, son utilisation pour décrire des ressources d'un domaine particulier nécessite souvent des raffinements voire des extensions. Ceci est le cas des sources de données biologiques. En effet, leur spécificité a nécessité la proposition d'un modèle hiérarchique qui étend le modèle du DCMI pour l'organisation des méta-données à attacher à ces sources. Le modèle de métadonnées BioRegistry [Smaïl-Tabbone et al., 2005], schématisé dans la figure 2.1, comporte 3 sections (chaque élément souligné correspond à un élément du DCMI). La première section correspond aux méta-données utilisées pour décrire chaque source de données biologiques. Elle comporte quatre catégories de méta-données pouvant être associées à une source de données :

- Identification de la source : nous retrouvons ici plusieurs champs du DCMI tels que title, publisher, description, coverage/temporal...
- Thèmes couverts par la source : nous retrouvons ici le champ subject du DCMI mais aussi un champ relatif aux organismes couverts par une source (par exemple, la source nommée Mouse Genome DB contient des données sur l'organisme Mouse).
- Qualité de la source : cette catégorie, absente du DCMI, est cruciale pour documenter la qualité d'une source biologique par rapport au mode de validation de ses entrées, la compatibilité par rapport aux standards, la couverture (nombre de gènes, nombre de contigs...) et l'existence de références croisées avec d'autres sources de données.
- Disponibilité de la source : cette catégorie regroupe les champs concernant les adresses des différents sites donnant accès à une source de donnée, ainsi que les contraintes d'accès pour les mondes académique et industriel (gratuité, authentification...).

La seconde section est consacrée à la documentation des ontologies de domaines utilisées (identifiant, version, localisation...). La troisième section modélise les relations pouvant exister entre plusieurs sources de données.

Prise en compte de la sémantique dans la modélisation de BioRegistry

La modélisation de BioRegistry est faite conformément aux recommandations du DCMI en utilisant autant que possible des types de données standards pour les différents champs de notre modèle ("syntax encoding schema") et surtout en s'appuyant sur des vocabulaires standards ou ontologies de domaines lorsqu'ils existent ("vocabulary encoding schema"). Ainsi, le thésaurus MeSH peut être utilisé pour renseigner les sujets traités par une source de données, la taxonomie NCBI peut être utilisée pour renseigner les organismes couverts par une source, etc. De plus, le modèle de BioRegistry est ouvert et peut intégrer d'autres ontologies de domaines qui pourront coexister grâce au système d'association d'un terme d'indexation avec la référence de l'ontologie dont il est issu (cf. éléments Subjects et Organisms dans la figure 2.1).

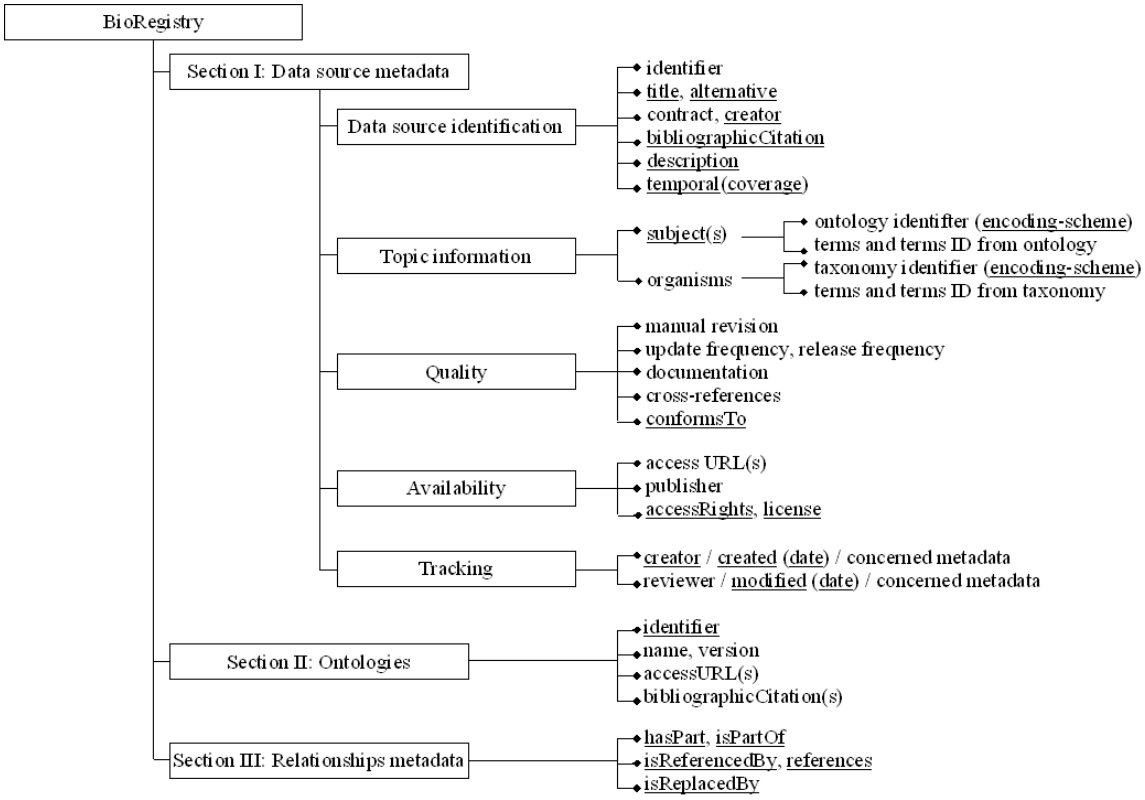


FIGURE 2.1 – Le modèle de métadonnées BioRegistry.

2.3.2 Peuplement de l'annuaire BioRegistry

Le modèle de données BioRegistry a été utilisé pour concevoir l'annuaire de sources de données biologique BioRegistry. L'annuaire BioRegistry constitue en quelque sorte une instance concrète du modèle détaillé dans la section précédente. Cette tâche est confrontée d'une part à la disponibilité des données nécessaires à la description des sources de données et d'autre part à l'existence de ressources sémantiques. Le catalogue NAR a été choisi comme point de de départ pour la création d'une telle instance puis qu'il référence de manière exhaustive les sources de données disponibles et effectue des mises à jours régulières pour prendre en compte l'apparition de nouvelles sources. Un premier ensemble de métadonnées est extrait directement à partir du catalogue NAR. Ensuite, en suivant certains liens du catalogue, d'autres métadonnées sont récupérées à partir de ressources externes tels que les articles scientifiques décrivant les sources. Les métadonnées associées à l'ensemble des sources sont enregistrées dans une base de données relationnelle. Le modèle relationnel de cette base est donné dans la figure 2.2.

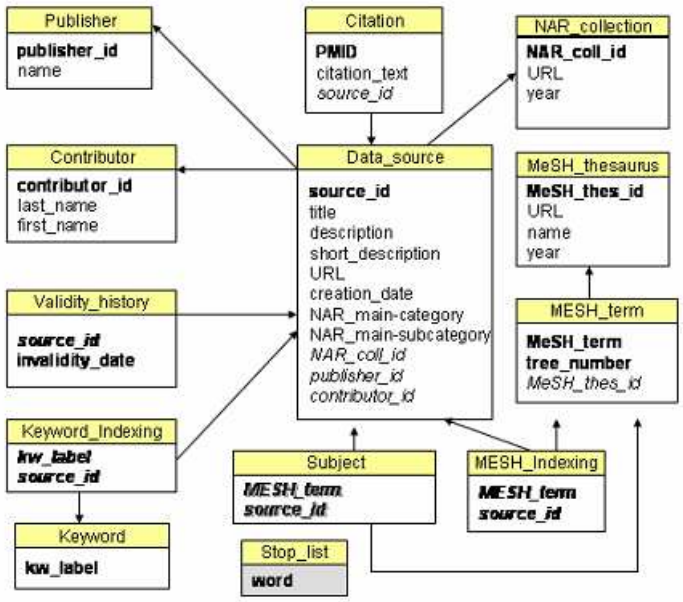


FIGURE 2.2 – Le modèle relationnel de la base de données de l'annuaire BioRegistry.

Les métadonnées sont collectées et stockées dans la bases de données de manière automatique. Le processus de collecte consiste d'abord à parcourir la liste des sources de données dans le catalogue NAR pour extraire pour chaque source les métadonnées "*title*", "*short description*" et "*url*" (voir la table "Data_source" dans le modèle relationnel figure 2.2) ainsi que le lien vers la description détaillée de la source ("NAR Database summary"). L'analyse de cette description détaillée permet d'extraire les métadonnées "*description*", "*NAR_main_category*", "*NAR_main_subcategory*" ainsi que le lien vers la citation correspondante à la source (l'article scientifique décrivant qui la décrit). A cette étape, un ensemble de termes MeSH est extrait à partir de la description de la source de données. L'accès à la citation permet de récupérer son identifiant PubMed. Cet identifiant est par la suite utilisé dans la base de données bibliographiques Medline pour extraire des métadonnées relatives à la citation, en particulier les termes MeSH indexant cette citation et qui seront stockés dans la table "Subject". La séquence d'opé-

rations dans un processus de collecte de métadonnées est schématisée dans la figure 2.3. Les métadonnées collectées pour chaque source sont réparties sur les tables de la base de données conformément au modèle relationnel donné dans la figure 2.2.

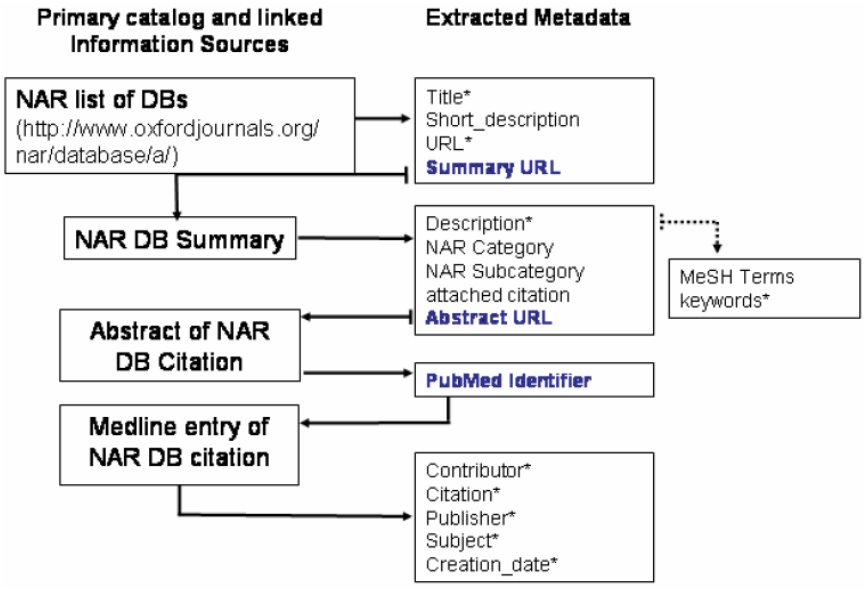


FIGURE 2.3 – Déroulement du processus d'extraction automatique des métadonnées relatives aux sources de données dans l'annuaire BioRegistry. La métadonnées avec une astérisque correspondent aux métadonnées du DCMI.

2.3.3 Interrogation de l'annuaire BioRegistry

L'organisation des métadonnées dans BioRegistry sous la forme d'une base de données relationnelle permet un premier mode d'interrogation de l'annuaire par l'intermédiaire de requêtes SQL. Différentes combinaisons de métadonnées peuvent ainsi être obtenues grâce à l'expressivité du langage SQL. L'expression des requêtes est facilitée par la mise en place de formulaires avec assistance de saisie à l'aide de champs à choix multiples et de listes de selections. Malgré l'expressivité de SQL et les facilités d'interrogation par formulaire, ce mode d'interaction avec l'annuaire BioRegistry reste purement syntaxique et nécessite beaucoup d'expertise pour aboutir à des résultats satisfaisants. Les limites de ce mode sont de plus accentuées par la multiplicité et l'hétérogénéité des sources de données biologiques qui conduisent à une méconnaissance sur les contenus et les propriétés de sources de données, ce qui empêche de formuler correctement une requête sur l'annuaire des sources.

Une meilleure interaction avec l'annuaire BioRegistry nécessite la prise en compte des connaissances de domaine en considérant les ressources sémantiques utilisées dans la phase d'annotation (telles que MeSH thesaurus). La prise en compte de l'aspect sémantique permet de mieux associer les requêtes avec les sources pertinentes et d'établir une classification de l'ensemble des sources de données en fonction de la similarité de leurs contenus. La possibilité de naviguer dans une telle classification peut conduire à découvrir de nouvelles sources pertinentes. La prise en compte de ces aspects nous conduit à définir l'approche de découverte de sources de données guidée par des connaissances de domaine.

2.4 Découverte de sources de données biologiques "guidée par des connaissances"

L'approche de découverte de sources de données biologiques guidée par des connaissances de domaine définie autour de l'annuaire BioRegistry adopte et étend le modèle d'architecture orientée service (SOA) [Josuttis, 2007] utilisée pour la découverte de services web. La figure 2.4 donne la forme générale du modèle SOA.

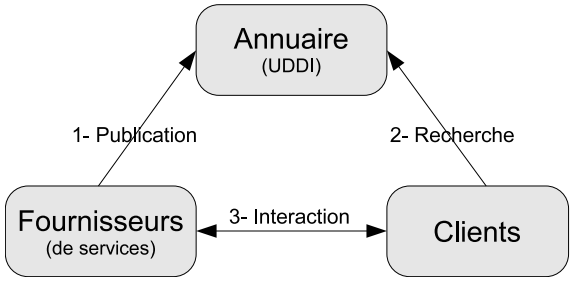


FIGURE 2.4 – L'architecture orientée services (SOA).

L'extension du modèle SOA est effectuée principalement pour assurer la prise en des connaissances de domaines à la fois dans la phase de description des sources de données et dans la phase de l'identification des sources pertinentes. L'architecture étendue est donnée dans la figure 2.5. Dans cette architecture, l'annuaire BioRegistry joue un rôle similaire à l'annuaire UDDI (Universal Description Discovery and Integration) des services Web, les sources de données biologiques jouent le rôle des fournisseurs de services Web et les utilisateurs des sources jouent le rôle des clients demandeurs de services Web.

La correspondance entre les deux modèles existe aussi entre les opérations entre les entités de chaque modèle.

1. L'annotation des sources de données correspond à la publication des services Web. Une première différence réside dans le fait que l'annotation est une opération propre à l'annuaire BioRegistry qui n'implique pas une participation des sources alors que la publication d'un service est opération déclenchée par le fournisseur de ce services. La deuxième différence, qui illustre l'aspect sémantique de BioRegistry, consiste à tenir compte des ressources sémantiques lors de l'annotation des sources.
2. La recherche des sources contenant de données pertinentes correspond à la recherche de services offrant certaines fonctions. De la même manière que pour l'annotation, la recherche des sources de données bénéficie de la prise en compte de la sémantique dans BioRegistry et de la possibilité de navigation dans la classification de l'ensemble des sources.
3. L'interaction entre les utilisateurs et les sources de données identifiées lors de la phase de recherche consiste à accéder aux données contenues dans ces sources. Cette interaction est assurée par les interfaces conçues par les auteurs des sources. Cette étape correspond à l'interaction entre les clients et les fournisseurs de services Web qui consiste à invoquer ces services à travers des requêtes exprimées dans un language particulier, WSDL (Web Services Description Language), en respectant les paramètres d'exécution de chaque service.

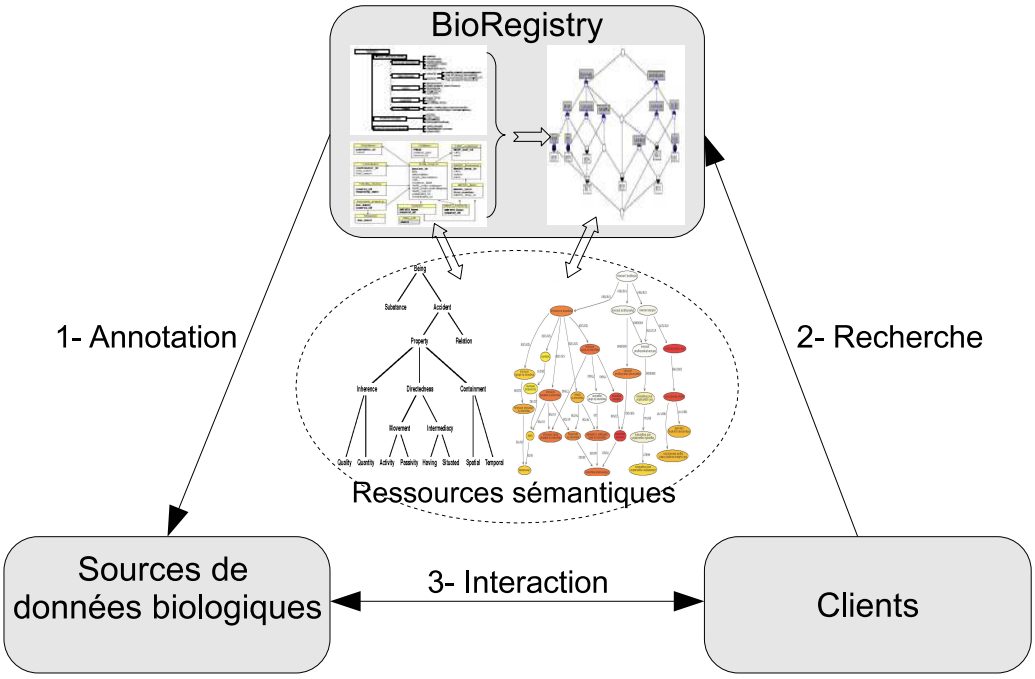


FIGURE 2.5 – Role de l'annuaire BioRegistry dans un système de découverte de sources de données biologiques guidée par des connaissances de domaine.

Chapitre 3

Analyse de Concepts Formels : fondement théorique, applications et extensions

Sommaire

3.1 Introduction 22
3.2 Théorie des treillis : Notions de base 22
3.2.1 Ensemble ordonné 22
3.2.2 Treillis 23
3.2.3 Fermeture 24
3.2.4 Connexion de Galois 24
3.3 Analyse de concepts formels 25
3.3.1 Origine et contexte philosophique 25
3.3.2 Contexte formel 25
3.3.3 Connexion de Galois dans un contexte formel 25
3.3.4 Concept formel 26
3.3.5 Treillis de concepts 27
3.3.6 Algorithmes de construction de treillis de concepts 28
3.3.7 Implications dans un contexte formel 33
3.4 ACF et découverte de ressources 35
3.4.1 Principales motivations 36
3.4.2 ACF et Recherche d'Information 36
3.5 ACF et données complexes 39
3.5.1 Contexte formel multivalué 39
3.5.2 Échelonnage conceptuel 40
3.6 Les extensions de l'ACF 42
3.6.1 Analyse de Concepts Formels Flous 43
3.6.2 Analyse de Concepts Logiques 46
3.6.3 Extensions Relationnelles de l'ACF 48
3.6.4 Extension aux objets symboliques 53
3.6.5 Autres extensions 54

3.1 Introduction

L'Analyse de Concepts Formels (ACF), appelée aussi Analyse Formelle de Concepts (AFC), est un formalisme qui constitue un pont entre les mathématiques, en particulier la théorie des ensembles ordonnés, et les applications d'analyse de données. Il s'agit de la restructuration, commencée par Wille et finalisée par Ganter et Wille, de la théorie des treillis définie par G. Birkhoff en adéquation avec la philosophie de la pensée humaine définie par C.S. Peirce. Dans ce chapitre nous dresserons un état de l'art qui couvre à la fois les notions de base et les définitions mathématiques relatives à la théorie des ensembles ordonnés et les applications en analyse de données qui s'appuient sur l'ACF. La suite du chapitre est organisée comme suit. Dans la section 3.2, nous rappelons les notions de base la théorie des treillis nécessaires à la compréhension de l'ACF. Dans la section 3.3 nous détaillons les notions et les définitions relatives à l'ACF et nous donnons un survol des principaux algorithmes développés dans le cadre de l'ACF. Dans la section 3.4, nous évoquons les principales applications de découverte de ressources qui s'appuient sur l'ACF. Dans la section 3.5 nous détaillons la manière dont les résultats de l'ACF sont appliqués en présence de données complexes. Finalement, dans la section 3.6, nous citons les principales approches qui ont tenté d'étendre les résultats à différentes formes de données complexes.

3.2 Théorie des treillis : Notions de base

3.2.1 Ensemble ordonné

Définition 1 (Relation binaire) *Une **relation binaire** R entre deux ensembles M et N est un ensemble de couples d'éléments (m, n) tels que $m \in M$ et $n \in N$, i.e. un sous ensemble de $M \times N$. $(m, n) \in R$ (aussi noté par mRn) signifie que l'élément m est en relation R avec l'élément n. Si $M = N$, on parle de relation binaire sur M. R^{-1} est la relation inverse de R, i.e. la relation entre N et M telle que $nR^{-1}m \Leftrightarrow mRn$.*

Définition 2 (Relation d'ordre (partiel)) *Une relation binaire R sur un ensemble E est dite **relation d'ordre partiel** (ou simplement relation d'ordre) sur E si elle vérifie les conditions suivantes pour tous $x, y, z \in E$:*

1. *$(x, x) \in R$ (R est reflexive)*
2. *si $(x, y) \in R$ et $x \neq y$ alors $(y, x) \notin R$ (R est antisymétrique)*
3. *si $(x, y) \in R$ et $(y, z) \in R$ alors $(x, z) \in R$ (R est transitive)*

Une relation d'ordre R est souvent notée par $\leq$ (R^{-1} est notée par "$\geq$") et on dit que "*x est plus petit que y*" lorsque $x \leq y$.

Définition 3 (Ensemble ordonné) *Un **ensemble partiellement ordonné** (ou simplement ensemble ordonné) est un couple $(E, \leq)$ où E est un ensemble et "$\leq$" est une relation d'ordre sur E.*

Dans un ensemble ordonné $(E, \leq)$, deux éléments x et y de E sont dits **comparables** lorsque $x \leq y$ ou $y \leq x$, autrement ils sont dits **incomparables**. Pour deux éléments comparables et différents, $x \leq y$ et $x \neq y$, on note $x < y$. Un sous ensemble de $(E, \leq)$ dans lequel tous les elements sont comparables est appelé **chaîne**. Un sous ensemble de $(E, \leq)$ dans lequel tous les éléments sont incomparables est appelé **anti-chaîne**.

Définition 4 (Successeur, prédécesseur, couverture) *Soient $(E, \leq)$ un ensemble ordonné et $x, y \in E$. y est dit **successeur** de x lorsque $x < y$ et il n'existe aucun élément $z \in E$ tel que $x < z < y$. Dans ces cas, x est dit **prédécesseur** de y et on note $x \prec y$. Lorsque x est un **prédécesseur** de y on dit que x **couvre** y (et que y est couvert par x). La **couverture** de x est formée par tout ses successeurs.*

Tout ensemble ordonné, $(E, \leq)$, peut être représenté graphiquement par un diagramme appelé "**diagramme de Hasse**" (ou diagramme de couverture) et obtenu comme suit :

1. Tout élément de E est représenté par une petit cercle dans le plan
2. Si $x, y \in E$ et $x \prec y$ alors le cercle correspondant à y doit être au-dessus de celui correspondant à x et les deux cercles sont reliés par un segment.

À partir d'un tel diagramme on peut lire la relation d'ordre comme suit : $x < y$ si et seulement s'il existe un chemin ascendant qui relie le cercle correspondant à x à celui correspondant à y.

Définition 5 (Principe de dualité des ensembles ordonnés) *Soit $(E, \leq)$ un ensemble ordonné. La relation inverse "$\geq$" de "$\leq$" est aussi une relation d'ordre sur E. "$\geq$" est appelée **duale** de "$\leq$" et $(E, \geq)$ est appelé le dual de l'ensemble ordonné $(E, \leq)$.*

Le diagramme de Hasse de $(E, \geq)$ peut être obtenu à partir de celui de $(E, \leq)$ par une simple réflexion horizontale. De plus, il possible de dériver les propriétés duales de $(E, \geq)$ à partir des propriétés de $(E, \leq)$.

3.2.2 Treillis

Définition 6 (Majorant, minorant, supremum, infimum) *Soient $(E, \leq)$ un ensemble ordonné et S un sous ensemble de E. Un element $a \in E$ est dit **majorant** de S lorsque $a \geq s \; \forall s \in S$. De façon duale, $a \in E$ est dit **minorant** de S lorsque $a \leq s \; \forall s \in S$.*

*Le plus petit majorant (respectivement minorant) de S, s'il existe, est appelé **supremum** ou borne supérieure (respectivement **infimum** ou borne inférieure) de S et noté $\bigvee S$ (respectivement $\bigwedge S$). Dans le cas ou $S = \{x, y\}$, $\bigvee S$ et $\bigwedge S$ sont aussi notés par $x \vee y$ et $x \wedge y$ respectivement.*

Dans tout ensemble ordonné, lorsque le supremum (respectivement l'infimum) existe, il est unique.

Définition 7 (Treillis, treillis complet) *Un **treillis** est un ensemble partiellement ordonné $(E, \leq)$ tel que $x \vee y$ et $x \wedge y$ existent pout tout couple d'éléments $x, y \in E$. Un treillis est dit **complet** si $\bigvee S$ et $\bigwedge S$ existent pour tout sous ensemble S de E. En particulier, un treillis complet admet un élément maximal (top) noté par $\top$ et un élément minimal (bottom) noté par $\bot$.*

Tout treillis fini est un treillis complet.

Définition 8 (Demi-treillis) *Un ensemble ordonné $(E, \leq)$ est un **sup-demi-treillis** (respectivement **inf-demi-treillis**) si tout couple d'éléments $x, y \in E$ admet un supremum $x \vee y$ (respectivement un infimum $x \wedge y$).*

3.2.3 Fermeture

Définition 9 (Fermeture) *On appelle* ***opérateur de fermeture*** *sur un ensemble ordonné, $(E, \leq)$, toute application $\varphi : E \to E$ qui vérifie les propriétés suivantes pour tout $x, y \in E$:*

- *$x \leq \varphi(x)$ (φ est extensive),*
- *si $x \leq y$ alors $\varphi(x) \leq \varphi(y)$ (φ est monotone croissante),*
- *$\varphi(x) = \varphi(\varphi(x))$ (φ est idempotente).*

Un élément $x \in E$ est dit ***fermé*** *pour φ si et seulement si $x = \varphi(x)$.*

Définition 10 (Système de fermeture) *Un* ***système de fermeture*** *(dit aussi système de fermés) sur un ensemble E est un ensemble de parties de E contenant E et fermé pour l'intersection. Formellement, $\mathfrak{C} \subseteq \mathfrak{P}(E)$ est un système de fermeture si les conditions suivantes sont vérifiées :*

1. *$E \in \mathfrak{C}$*
2. *si $\mathfrak{A} \subseteq \mathfrak{C}$ alors $\bigcap \mathfrak{A} \in \mathfrak{C}$ (l'intersection de toute suite de fermés de $\mathfrak{A}$ appartient à $\mathfrak{C}$)*

Les deux notions, *opérateur de fermeture* et *système de fermeture*, sont fortement liées. En effet, tout système de fermeture peut être considéré comme l'ensemble de tous les fermés d'un opérateur de fermeture. Inversement, il est possible de définir un opérateur de fermeture sur tout système de fermeture. Cette relation est exprimée dans le théorème suivant.

Théorème 1 *Si $\mathfrak{C}$ est un système fermé sur E alors l'application $\varphi_{\mathfrak{C}}$ suivante*

$$\varphi_{\mathfrak{C}} : X \mapsto \bigcap \{A \in \mathfrak{C} \mid X \subseteq A\}$$

définit un opérateur de fermeture sur E ($\varphi_{\mathfrak{C}} : E \to E$). Inversement, l'ensemble

$$\mathfrak{C}_{\varphi} = \{\varphi(X) \mid X \subseteq E\}$$

de tous les fermés pour un opérateur de fermeture φ est un système fermé.

La notion de fermeture est aussi liée à la définition des treillis. En effet, l'ensemble des fermés dans un ensemble E étant donné un opérateur de fermeture sur E forme un treillis complet. Inversement, tout treillis complet est isomorphe au treillis des fermés d'un opérateur de fermeture et on a le théorème suivant.

Théorème 2 *Si $\mathfrak{C}$ est un système fermé alors $(\mathfrak{C}, \subseteq)$ est un treillis complet avec $\bigwedge X = \bigcap X$ et $\bigvee X = \varphi_{\mathfrak{C}}(\bigcup X)$ $\forall X \subseteq \mathfrak{C}$. Inversement, tout treillis complet est isomorphe au treillis des fermés d'un opérateur de fermeture.*

3.2.4 Connexion de Galois

Définition 11 (Connexion de Galois) *Soient $\varphi : P \to Q$ et $\psi : Q \to P$ deux applications entre deux ensembles ordonnés $(P, \leq_P)$ et $(Q, \leq_Q)$. φ et ψ forment une* ***connexion de Galois*** *entre $(P, \leq_P)$ et $(Q, \leq_Q)$ si elles vérifient les conditions suivantes pour tous $p, p_1, p_2 \in P$ et $q, q_1, q_2 \in Q$:*

1. *si $p_1 \leq_P p_2$ alors $\varphi(p_2) \leq_Q \varphi(p_1)$,*
2. *si $q_1 \leq_Q q_2$ alors $\psi(q_2) \leq_P \psi(q_1)$,*
3. *$p \leq_P \psi(\varphi(p))$ et $q \leq_Q \varphi(\psi(q))$.*

Les conditions données dans la définition précédente sont équivalentes à la formule suivante :

$$p \leq_P \psi(q) \Leftrightarrow q \leq_Q \varphi(p)$$

3.3 Analyse de concepts formels

3.3.1 Origine et contexte philosophique

L'*Analyse de Concepts Formel* (ACF) [Wille, 1982, Ganter and Wille, 1999] a été présentée comme un domaine de mathématiques appliquées qui consiste à restructurer la théorie des treillis [Birkhoff, 1967] afin de faciliter son utilisation dans des applications du monde réel et de permettre l'interprétation de ses notions en dehors du cadre théorique aussi bien par des mathématiciens que par des non-mathématiciens. L'objectif à travers la mise en place de l'ACF est d' "*atteindre une théorie structurée qui expose les pensées formelles selon des interprétations significatives et permettre ainsi des communications et des discussions critiques de leurs contenus*" [Wille, 1996]. Pour cela l'ACF a été centrée autour de la notion de *concept* qui, du point de vue philosophique, est considéré comme l'unité de base de la pensée humaine. De manière informelle, un concept peut être défini comme un groupement d'individus et de leurs propriétés communes. Toujours en liaison avec le point de vue philosophique, en particulier celui de *Pierce* [20] qui considère que "*dans tout processus de raisonnement ou d'argumentation, on ne peut étudier qu'une part de la realité*", l'ACF extrait les concepts à partir de contextes restreints [Dau and Klinger, 2005]. De tels contextes, appelé contextes formels, constituent le point de départ de l'ACF et sont définis dans la section suivante.

3.3.2 Contexte formel

Définition 12 (Contexte formel) *Un* ***contexte formel*** *est un triplet* $\mathbb{K} = (G, M, I)$ *où* G *est un ensemble d'objets,* M *est un ensemble d'attributs et* I *est une relation binaire entre* G *et* M *appelée relation d'incidence de* $\mathbb{K}$ *et vérifiant* $I \subseteq G \times M$*. Un couple* $(g, m) \in I$ *(noté aussi* gIm*) signifie que l'objet* $g \in G$ *possède l'attribut* $m \in M$*.*

Un contexte formel peut être représenté sous la forme d'un tableau où les lignes correspondent aux objets et les colonnes correspondent aux attributs. Les cases du tableau sont remplies comme suit : si le i$^{\text{ème}}$ objet g est en relation I avec le j$^{\text{ème}}$ alors la case intersection de la ligne i et la colonne j contient "$\times$" sinon la case est vide. La table 3.1 [Davey and Priestley, 2002] donne un exemple de contexte formel représentant les planètes du système solaire [21]. Dans cet exemple de contexte, Mercure possède les attributs : petite taille, proche du soleil et n'est pas un satellite.

3.3.3 Connexion de Galois dans un contexte formel

Définition 13 *Soit* $\mathbb{K}$ *un contexte formel. Pour tout* $A \subseteq G$ *et* $B \subseteq M$*, on définit :*

$$A' = \{m \in M | \forall g \in A, gIm\}$$

$$B' = \{g \in G | \forall m \in B, gIm\}$$

Intuitivement, A' est l'ensemble des attributs communs à tous les objets de A et B' est l'ensemble des objets possédant tous les attributs de B. Les applications $' : \mathfrak{P}(G) \rightarrow \mathfrak{P}(M)$ et

20. Charles Sanders Peirce (10 septembre 1839 - 19 avril 1914) est un sémiologue et philosophe américain. Il est considéré comme le fondateur du courant pragmatiste avec William James et, avec Ferdinand de Saussure, un des deux pères de la sémiologie (ou sémiotique) moderne. Ces dernières décennies, sa pensée a été l'objet d'un regain d'intérêt. Il est désormais considéré comme un innovateur dans de nombreux domaines, en particulier dans la méthodologie de la recherche et dans la philosophie des sciences.

21. Bien que Pluton n'est plus considérée comme planète, nous continuerons à la garder dans le contexte pour conserver l'exemple introduit dans [Davey and Priestley, 2002]

TABLE 3.1 – Un contexte formel représentant les planètes du système solaire.

Objet \ Attribut	Taille			Distance au soleil		Satellite	
	petite	moyenne	grande	proche	loin	oui	non
Mercure	×			×			×
Vénus	×			×			×
Terre	×			×		×	
Mars	×			×		×	
Jupiter			×		×	×	
Saturne			×		×	×	
Uranus		×			×	×	
Neptune		×			×	×	
Pluton	×				×	×	

$' : \mathfrak{P}(M) \to \mathfrak{P}(G)$ sont appelées opérateurs de dérivation entre l'ensemble des objets et l'ensemble des attributs dans un contexte formel. La composition de ces opérateurs produit deux opérateurs $'' : \mathfrak{P}(G) \to \mathfrak{P}(G)$ et $'' : \mathfrak{P}(M) \to \mathfrak{P}(M)$. Le premier opérateur permet d'associer à un ensemble d'objets A l'ensemble maximal d'objets dans G ayant les attributs communs aux objets de A. Cet ensemble est noté par A''. De façon duale, le second opérateur permet d'associer à un ensemble d'attributs B l'ensemble maximal d'attributs dans M communs aux objets ayant les attributs dans B. Cet ensemble est noté par B''.

Les opérateurs $'' : \mathfrak{P}(G) \to \mathfrak{P}(G)$ et $'' : \mathfrak{P}(M) \to \mathfrak{P}(M)$ définissent deux fermetures respectivement sur l'ensemble des parties de G, $\mathfrak{P}(G)$ (noté aussi par 2^G), et sur l'ensemble des parties de M, $\mathfrak{P}(M)$ (noté aussi par 2^M). Les ensembles A'' et B'' sont des fermés pour ces deux opérateurs respectifs. L'ensemble des fermés de $\mathfrak{P}(G)$ muni de l'inclusion est un treillis complet. De la même façon, l'ensemble des fermés de $\mathfrak{P}(M)$ muni de l'inclusion est un treillis complet. Les opérateurs de dérivation $' : \mathfrak{P}(G) \to \mathfrak{P}(M)$ et $' : \mathfrak{P}(M) \to \mathfrak{P}(G)$ forment une bijection entre les ensembles de fermés de $\mathfrak{P}(G)$ et $\mathfrak{P}(M)$ et définissent un isomorphisme entre les deux treillis respectifs : à chaque fermé A dans $\mathfrak{P}(G)$ correspond un unique fermé B dans $\mathfrak{P}(M)$ et vice versa. De cette façon, les opérateurs de dérivation $'' : \mathfrak{P}(G) \to \mathfrak{P}(G)$ et $'' : \mathfrak{P}(M) \to \mathfrak{P}(M)$ forment une connexion de Galois entre $(\mathfrak{P}(G), \subseteq)$ et $(\mathfrak{P}(M), \subseteq)$.

3.3.4 Concept formel

Les paires de fermés reliées par cette connexion de Galois, détaillée dans la section précédente, forment les concepts formels définis comme suit.

Définition 14 (Concept formel) *Soit $\mathbb{K} = (G, M, I)$ un contexte formel. Un **concept formel** est un couple (A, B) tel que $A \subseteq G$, $B \subseteq M$, $A' = B$ et $B' = A$. A et B sont respectivement appelées extension (extent) et intension (intent) du concept formel (A, B). L'ensemble des concepts formels associés au contexte formel $\mathbb{K} = (G, M, I)$ est noté par $\mathfrak{B}(G, M, I)$.*

Dans un contexte formel, un concept correspond à un rectangle maximal de la table formée par la relation binaire du contexte : tout objet de l'extension a tous les attributs de l'intension. Il est important de noter que cette notion de rectangle maximal est indépendante de l'ordre des lignes

et des colonnes. Ces ensembles maximaux d'objets et d'attributs correspondent à des fermés dans $\mathfrak{P}(G)$ et $\mathfrak{P}(M)$ respectivement. Un sous-ensemble B de M est l'intension d'un concept formel dans $\mathfrak{B}(G, M, I)$ si et seulement si $B'' = B$ (B est fermé pour $''$) et, de façon duale, un sous ensemble A de G est l'extension d'un concept formel dans $\mathfrak{B}(G, M, I)$ si et seulement si $A'' = A$ (A est fermé pour $''$).

Les concepts de $\mathfrak{B}(G, M, I)$ sont ordonnés par une relation de d'ordre hiérarchique entre concepts (appelée aussi relation de subsomption) notée par "$\leq$" et définie suit.

Définition 15 (Relation de "subsomption") *Soient $(A1, B1)$ et $(A2, B2)$ deux concepts formels de $\mathfrak{B}(G, M, I)$. $(A1, B1) \leq (A2, B2)$ si et seulement si $A1 \subseteq A2$ (ou de façon duale $B2 \subseteq B1$). $(A2, B2)$ est dit **super-concept** de $(A1, B1)$ et $(A1, B1)$ est dit **sous-concept** de $(A2, B2)$. La relation "$\leq$" est dite relation de subsomption.*

La relation "$\leq$" s'appuie sur deux inclusions duales, entre ensembles d'objets et entre ensembles d'attributs et peut ainsi être interprétée comme une relation de généralisation/spécialisation entre les concepts formels. Un concept est plus général qu'un autre concept s'il contient plus d'objets dans son extension. En contre partie, les attributs partagés par ces objets sont réduits. De façon duale, un concept est plus spécifique qu'un autre s'il contient moins d'objets dans son extension. Ces objets ont plus d'attributs en commun.

3.3.5 Treillis de concepts

Définition 16 (Treillis de concepts) *La relation "$\leq$" permet d'organiser les concepts formels en un treillis complet $(\mathfrak{B}(G, M, I), \leq)$ appelé **treillis de concepts** ou encore **treillis de Galois** [Birkhoff, 1967] et noté par $\underline{\mathfrak{B}}(G, M, I)$ ou $\underline{\mathfrak{B}}(\mathbb{K})$. Le supremum et l'infimum dans $\underline{\mathfrak{B}}(\mathbb{K})$ sont donnés par :*

$$\bigwedge_{j \in J} (A_j, B_j) = \left(\bigcap_{j \in J} A_j, \left(\bigcup_{j \in J} B_j \right)'' \right)$$

$$\bigvee_{j \in J} (A_j, B_j) = \left(\left(\bigcup_{j \in J} A_j \right)'', \bigcap_{j \in J} B_j \right)$$

Le diagramme de Hasse représentant le treillis de concepts correspondant au contexte formel donné dans la table 3.1 est donné dans la figure 3.1 (visualisé grâce au logiciel ConExp [22]). La notation des concepts est dite réduite : elle s'appuie sur l'héritage à la fois des attributs et des objets entre les concepts du treillis. Les attributs sont placés au plus haut dans le treillis : à chaque fois qu'un nœud N est étiqueté par un attribut m, tous les descendants de N dans le treillis héritent l'attribut m. De façon duale, les objets sont placés au plus bas dans le treillis : à chaque fois qu'un nœud N est étiqueté par un objet g, g est hérité "vers le haut" et tous les ancêtres de N le partagent. Ainsi l'extension A d'un concept (A, B) est obtenue en considérant tous les objets qui apparaissent sur les descendants du nœud N dans le treillis et son intension B est obtenue en considérant tous les attributs qui apparaissent sur les ancêtres du nœud N dans le treillis.

Le treillis de concepts est une représentation équivalente des données contenues dans un contexte formel qui met en avant les groupements possibles entre objets et attributs ainsi que les relations d'inclusion entre ces groupements. De plus, la représentation graphique du treillis de

22. http ://sourceforge.net/projects/conexp/

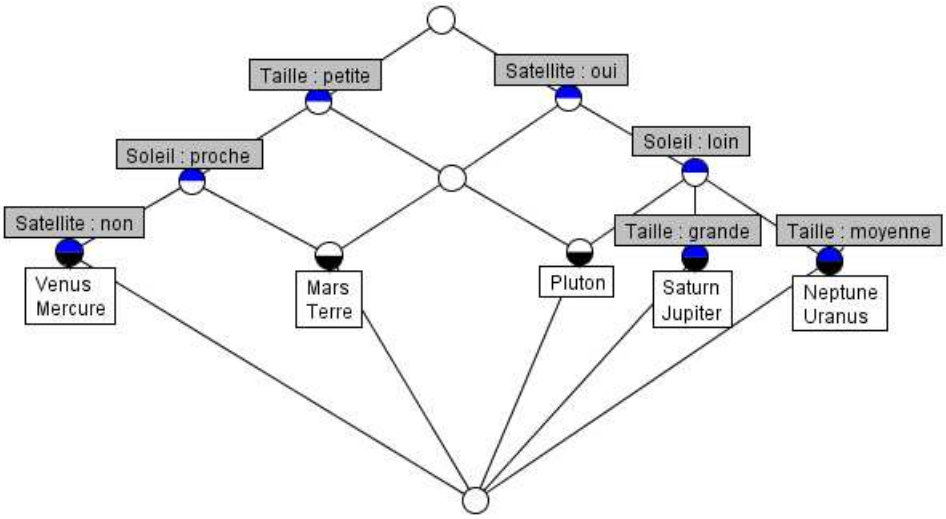


FIGURE 3.1 – Le treillis de concepts correspondant au contexte formel $\mathbb{K} = (G, M, I)$ donné dans la table 3.1

concepts, sous la forme d'un diagramme de Hasse, facilite la compréhension et l'interprétation de la relation entre les objets et les attributs d'une part et entre objets ou attributs d'autre part. L'avantage de cette représentation est qu'à partir d'un treillis de concepts il est toujours possible de retrouver le contexte formel correspondant et inversement.

3.3.6 Algorithmes de construction de treillis de concepts

La construction du treillis de concepts d'une relation binaire donnée peut être décomposée en trois parties [Guenoche, 1990, Guenoche and Mechelen, 1993] :

1. l'énumération des rectangles maximaux (les fermés),
2. la recherche de la relation d'ordre partiel entre ces rectangles
3. et la représentation graphique du treillis (construction du diagramme de HASSE correspondant au treillis).

Les deux premières étapes constituent le problème de calcul des concepts d'un treillis de concepts à partir d'un contexte formel et peuvent être exécutées simultanément ou de manière séquentielle (dans l'ordre donné plus haut). Cependant, la troisième étape fait partie du problème de visualisation de graphes. De ce fait, ces deux problématiques sont souvent traitées indépendamment ce qui a donné lieu à deux axes de recherche complémentaires : le premier consiste à proposer des algorithmes de plus en plus performants (complexité, temps d'execution, occupation mémoire, passage à l'échelle) pour le calcul de treillis de concepts à partir de contextes formels et le deuxième consiste à fournir des outils efficaces pour la visualisation de ces treillis.

Le problème de calcul des concepts d'un treillis de concepts à partir d'un contexte formel a fait l'objet de nombreux travaux de recherche [Chein, 1969, Norris, 1978, Ganter, 1984, Bordat, 1986, Guenoche, 1990, Ganter and Reuter, 1991, Guenoche and Mechelen, 1993, Kuznetsov, 1993, Dowling, 1993, Godin et al., 1995c, Nguifo and Njiwoua, 1998, Nourine and Raynaud, 1999, Lindig, 2000, Stumme et al., 2000, Stumme et al., 2002, Valtchev and Missaoui, 2001, Valtchev et al., 2002, der Merwe et al., 2004,

Martin and Eklund, 2008, Szathmary et al., 2008] qui ont abouti à la proposition d'une variété d'algorithmes dont les plus connus sont **Chein** [Chein, 1969], **Norris** [Norris, 1978] (la méthode dite de malgrange), **NextClosure** [Ganter, 1984, Ganter and Reuter, 1991], **Bordat** [Bordat, 1986], **Close-by-One** [Kuznetsov, 1993], **Godin** [Godin et al., 1995c], **Galois** [Carpineto and Romano, 1996, Carpineto and Romano, 2004a], **Nourine** [Nourine and Raynaud, 1999], **Lindig** [Lindig, 2000], **Titanic** [Stumme et al., 2000, Stumme et al., 2002], **Divide&Conquer** [Valtchev and Missaoui, 2001, Valtchev et al., 2002] et **AddIntent** [der Merwe et al., 2004]. Chacun de ces algorithmes se distingue des autres par plusieurs critères dont la stratégie de calcul des concepts, la recherche de l'ordre entre ces concepts, les structures de données utilisées pour le stockage des résultats intermédiaires et le résultat final. Les principaux algorithmes ont fait l'objet d'une comparaison détaillée dans [Kuznetsov and Obiedkov, 2002]. Cette comparaison a montré qu'aucun algorithme n'est meilleur que tous les autres sur tous les plans et que les performances d'un algorithme dépendent fortement des caractéristiques du contexte formel en entrée.

Dans la suite de cette section nous allons répartir les algorithmes en fonction de leurs stratégies d'acquisition de données à partir d'un contexte formel. Considérant ce critère, on distingue trois familles d'algorithmes : les algorithmes **batch** qui considèrent la totalité du contexte dès le départ, les algorithmes **incrémentaux** qui considèrent le contexte ligne par ligne et les algorithmes **d'assemblage** qui répartissent le contexte en deux et calculent les concepts correspondant à chaque moitié puis font l'assemblage.

Algorithmes batch

Les algorithmes batch constituent la première génération des algorithmes de construction de treillis. Il prennent en entrée le contexte formel tout entier et calculent les concepts formels et l'ordre entre ces concepts simultanément ou de manière séquentielle.

L'un des premiers algorithmes proposés de cette catégorie est l'algorithme de **Chein** [Chein, 1969] qui génère les concepts par niveaux. L'algorithme est itératif. Son point de départ est l'ensemble L_1 de couples (A, B) représentant les lignes du contexte formel (A contient un seul élément de G et $B = A'$). A chaque étape i, l'algorithme part d'un ensemble L_i et construit les éléments de L_{i+1}. Un élément $(A3, B3)$ de L_{i+1} est obtenu en combinant deux éléments $(A1, B1)$ et $(A2, B2)$ de L_i comme suit : $A3 = A1 \cup A2$ et $B3 = B1 \cap B2$. Les éléments de L_i inclues dans au moins un élément de L_{i-1} ne sont pas maximaux et sont donc supprimés. L'algorithme s'arrête lorsque L_{i+1} contient moins de deux éléments. Les éléments non supprimés après l'arrêt de l'algorithme sont les concepts du contexte formel considéré. Pour illustrer le fonctionnement de cet algorithme, nous considérons l'exemple de contexte formel donné dans la table 3.1 avec un renommage des attributs et des objets pour faciliter la lisibilité des concepts lors des étapes d'exécution de l'algorithme. Le contexte considéré est donné dans la table 3.2 et les traces d'exécution de l'algorithme sont données dans le la table 3.3. Les concepts formels calculés par l'algorithme à partir du contexte formel considéré sont les éléments non barrés dans le tableau 3.3. La complexité de l'algorithme de **Chein** est $O(|G|^3|M||L|)$ $|G|$ étant le nombre d'objets dans le contexte, $|M|$ le nombre d'attributs et $|L|$ le nombre de concepts formels dans le treillis obtenu.

Les autres algorithmes batch les plus connus sont les suivants. L'algorithme **NextClosure** [Ganter, 1984] constitue l'algorithme de référence pour le calcul des fermés dans un contexte formel. Il s'appuie sur l'ordre lexicographique entre ensembles d'attributs pour calculer les fermés. Le calcul des fermés peut être appliqué aux attributs (NextIntent) auquel cas on obtient les intensions des concepts formels ou bien aux objets (NextExtent) auquel cas on obtient les extensions des concepts formels. L'avantage majeur de cet algorithme est qu'il ne garde pas en

TABLE 3.2 – Contexte formel représentant les planètes du système solaire avec renommage des attributs et des objets.

	a	b	c	d	e	f	g
1	×			×			×
2	×			×			×
3	×			×		×	
4	×			×		×	
5			×		×	×	
6			×		×	×	
7		×			×	×	
8		×			×	×	
9	×				×	×	

TABLE 3.3 – Les ensembles de rectangles maximaux calculés par l'algorithme de Chein à partir du contexte formel donné dans le tableau 3.2. Les rectangles barrés correspondent à des rectangles non maximaux. Les rectangles qui les contiennent sont indiqués entre parenthèses et l'iteration durant laquelle ils ont été calculés est donnée par le nombre de *. Par exemple, "~~1 × adg~~ **(12 × adg)" est interprété comme suit : le rectangle "1 × adg" n'est pas maximal, il est remplacé par "12 × adg" retrouvé à la deuxième itération de l'algorithme.

L_1	L_2	L_3	L_4
~~1 × adg~~ **(12 × adg)	12 × adg	12349 × a	∅
~~2 × adg~~ **(12 × adg)	~~13 × ad~~ **(134 × ad)	3456789 × f	
~~3 × adf~~ **(34 × adf)	~~134 × ad~~ **(1234 × ad)		
~~4 × adf~~ **(34 × adf)	~~19 × a~~ **(129 × a)		
~~5 × cef~~ **(56 × cef)	1234 × ad		
~~6 × cef~~ **(56 × cef)	~~129 × a~~ ***(12349 × a)		
~~7 × bef~~ **(78 × bef)	34 × adf		
~~8 × bef~~ **(78 × bef)	~~35 × f~~ **(356 × f)		
9 × aef	~~356 × f~~ **(3567 × f)		
	~~3567 × f~~ **(35678 × f)		
	~~35678 × f~~ **(345678 × f)		
	~~39 × af~~ **(349 × af)		
	~~345678 × f~~ ***(3456789 × f)		
	~~349 × af~~		
	56 × cef		
	~~57 × ef~~ **(578 × ef)		
	~~578 × ef~~ **(5789 × ef)		
	~~5789 × ef~~ **(56789 × ef)		
	56789 × ef		
	78 × bef		

mémoire l'ensemble des fermés déjà calculés puisqu'il s'appuie sur un test de canonicité pour vérifier qu'un fermé n'a pas été généré. De ce fait, seul le dernier fermé généré doit être gardé en mémoire. La complexité théorique de l'algorithme **NextClosure** est $O(|G|^2|M||L|)$. L'algorithme **Bordat** [Bordat, 1986] construit les concepts en s'appuyant sur une structure d'arbre pour garder les résultats intermédiaire. Sa complexité est $O(|G||M|^2|L|)$. L'algorithme **Close-by-One** [Kuznetsov, 1993] utilise une technique similaire à NextClosure pour la génération des concepts et une structure particulière appelée *arbre CbO* pour garder en mémoire les concepts générés. Sa complexité est $O(|G|^2|M||L|)$. Finalement, l'algorithme **Titanic** [Stumme et al., 2001] utilise la notion de fréquence pour calculer les concepts sans faire d'intersection entre les ensembles d'attributs.

Algorithmes incrémentaux

Les algorithmes incrémentaux considèrent le contexte formel ligne par ligne (ou colonne par colonne) et construisent le treillis de concepts par ajouts successifs de ligne ou de colonne tout en conservant sa structure. À une étape k, les concepts formels correspondants au k premières lignes du contexte formel sont calculés. L'ajout de la $(k+1)^{\text{ème}}$ ligne entraîne la modification d'une partie des concepts calculés à l'étape k et l'ajout d'éventuels nouveaux concepts. Conçu ainsi, les algorithmes incrémentaux permettent de gérer les contextes dynamiques, où le nombre d'objets et/ou d'attributs peut évoluer, sans avoir à recalculer le treillis à partir de zéro suite à une modification du contexte. Les étapes de la construction incrémentale du treillis de concepts correspondant au contexte donné dans la table 3.2 sont schématisées dans la figure 3.2. Bien que la motivation du coté applicatif des algorithmes incrémentaux n'est apparue qu'au milieu des années 90 [Carpineto and Romano, 1993, Godin et al., 1995a, Godin et al., 1995b, Carpineto and Romano, 1996], le premier algorithme incrémental publié, l'algorithme de **Norris** [Norris, 1978], date de 1978 et reste parmi les plus performants [Kuznetsov and Obiedkov, 2002]. La complexité de l'algorithme de **Norris** est $O(|G|^2|M||L|)$

Plus tard, deux nouveaux algorithmes incrémentaux similaires à celui de Norris ont été proposés dans le cadre des approches d'analyse de données par treillis de concepts incluant la classification et la recherche d'information [Godin et al., 1995b, Carpineto and Romano, 1996] : l'algorithme **Galois** [Carpineto and Romano, 1993] et l'algorithme de **Godin** [Godin et al., 1995a, Godin et al., 1995c].

Dans cette même catégorie, on distingue l'algorithme de **Nourrine&Raynaud** [Nourine and Raynaud, 1999] connus pour être le seul algorithme de construction de treillis (toutes catégories confondues) à avoir la plus petite complexité théorique ($O((|G|+|M|)|G||L|)$) et l'algorithme le plus récent **AddIntent** [der Merwe et al., 2004] qui construit les concepts du treillis de manière récursive. La complexité de l'algorithme **AddIntent** est $O(|G|^2|M||L|)$.

Algorithmes d'assemblage

Les algorithmes d'assemblage constituent une évolution des algorithmes incrémentaux qui généralise le caractère incrémental à des ensembles d'objets/attributs. Ils permettent diviser un contexte formel en deux parties verticalement ou horizontalement puis de calculer le treillis de concepts correspondant à chaque partie et enfin d'assembler les treillis obtenus en un seul. Le seul algorithme connu de cette famille est l'algorithme **Divide&Conquer** [Valtchev and Missaoui, 2001, Valtchev et al., 2002]. Pour illustrer le principe de cette approche, nous considérons à nouveau le contexte formel $\mathbb{K}$ donné dans la table 3.2 qu'on divisera en deux contextes $\mathbb{K}_1$ et $\mathbb{K}_2$ comme

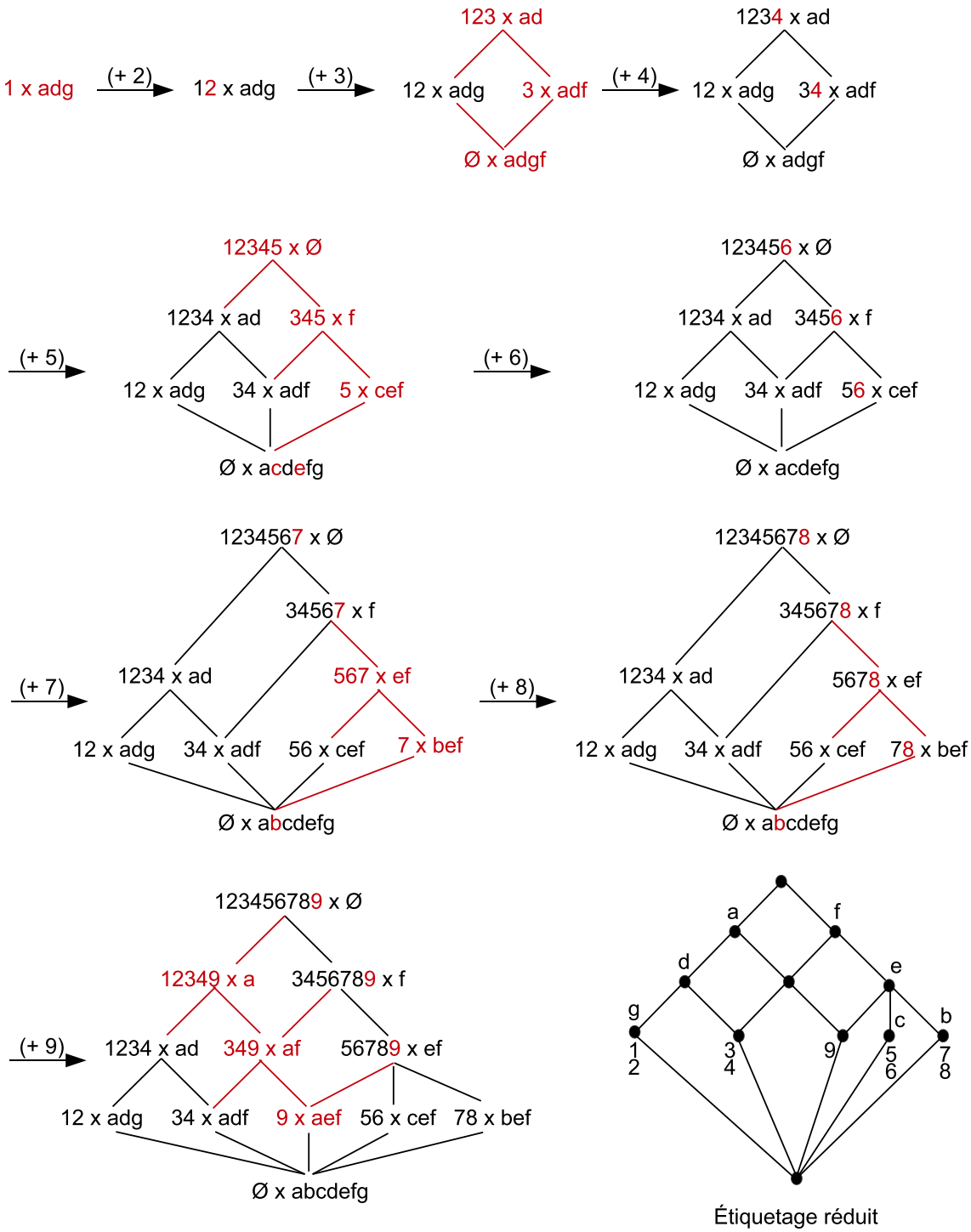


FIGURE 3.2 – Les étapes de la construction incrémentale du treillis de concepts correspondant au contexte formel donné dans la table 3.2. L'ajout d'une nouvelle ligne au treillis construit est symbolisée par "$\xrightarrow{(+\ n)}$" où n désigne le numéro de la ligne considérée. Les modifications engendrées par un tel ajout sont indiquées en rouge sur le treillis.

le montre la table 3.4 ($\mathbb{K}_1$ à gauche et $\mathbb{K}_2$ à droite).

TABLE 3.4 – Découpage du contexte formel en deux parties.

	a	b	c	d
1	×			×
2	×			×
3	×			×
4	×			×
5			×	
6			×	
7		×		
8		×		
9	×			

	e	f	g
1			×
2			×
3		×	
4		×	
5	×	×	
6	×	×	
7	×	×	
8	×	×	
9	×	×	

Les treillis $\underline{\mathfrak{B}}(\mathbb{K}_1)$ et $\underline{\mathfrak{B}}(\mathbb{K}_2)$ correspondant aux des deux contextes $\mathbb{K}_1$ et $\mathbb{K}_2$ sont donnés dans la figure 3.3.

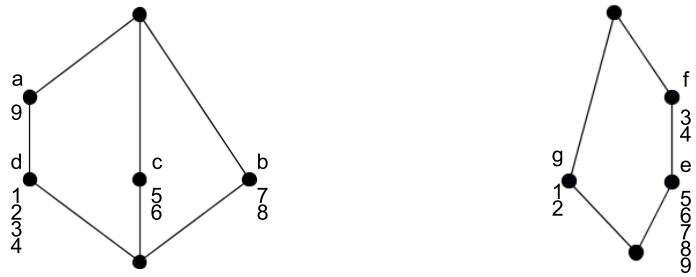


FIGURE 3.3 – Les treillis de concepts correspondant aux contextes formels donnés dans les tables 3.4 : $\underline{\mathfrak{B}}(\mathbb{K}_1)$ à gauche et $\underline{\mathfrak{B}}(\mathbb{K}_2)$ à droite.

Le lien entre les concepts des treillis $\underline{\mathfrak{B}}(\mathbb{K}_1)$ et $\underline{\mathfrak{B}}(\mathbb{K}_2)$ et ceux du treillis $\underline{\mathfrak{B}}(\mathbb{K})$ sont donnés dans la figure 3.4.

3.3.7 Implications dans un contexte formel

Définition 17 *Soient un contexte formel $\mathbb{K} = (G, M, I)$ et $B1, B2 \subseteq M$ deux ensembles d'attributs. On dit que $B1$ implique $B2$ si et seulement tout objet de G qui a les attributs de $B1$ a aussi les attributs de $B2$:*

$$B1 \rightarrow B2 \text{ ssi } B1' \subseteq B2'$$

Dans le contexte formel des planètes du système solaire donné dans la table 3.1, on a l'exemple d'implication suivante :

"Satellite : non" → "Taille : petite", "Distance au soleil : proche"

qui se lit : *toute planète n'ayant pas de satellite est de petite taille et proche du soleil.* Considérons $B1 =$ {"Satellite : non"} et $B2 =$ {"Taille : petite", "Distance au soleil : proche"}, nous avons

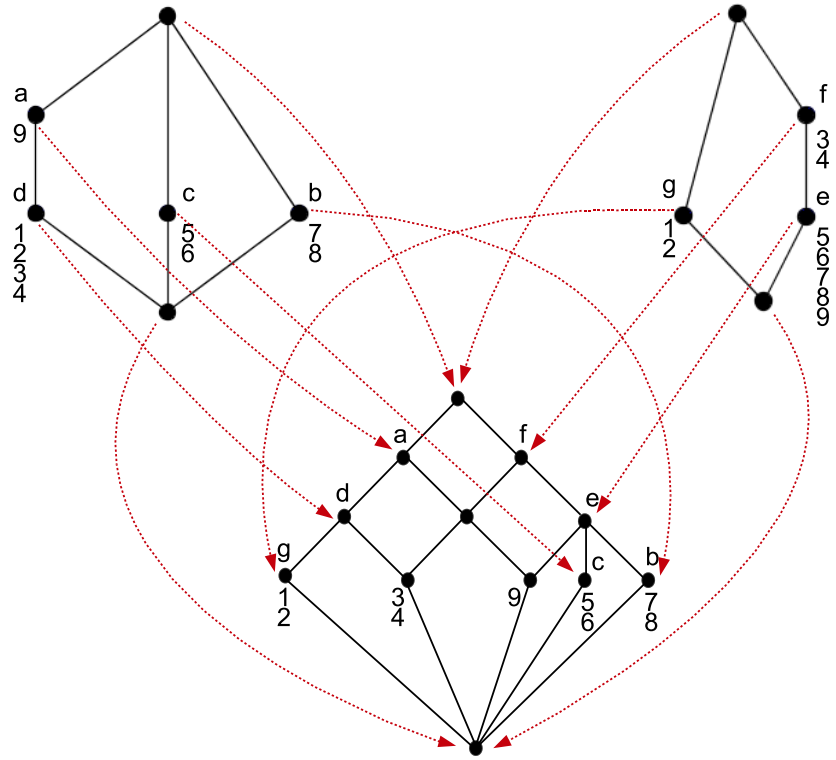


FIGURE 3.4 – Liens lien entre les concepts des treillis $\underline{\mathfrak{B}}(\mathbb{K}_1)$ et $\underline{\mathfrak{B}}(\mathbb{K}_2)$ et ceux du treillis $\underline{\mathfrak{B}}(\mathbb{K})$.

$B1' = \{Mercure, Venus\}$ et $B2' = \{Mercure, Venus, Terre, Mars, Pluton\}$ qui vérifient la définition 17.

Une implication de la forme $B1 \rightarrow B2$ peut être ramené à un ensemble d'implication de la forme $B1 \rightarrow b$ pour tout $b \in B2$. L'implication donné en exemple plus haut peut être ramenée aux deux implications suivantes :

"Satellite : non" $\rightarrow$ "Taille : petite"

et

"Satellite : non" $\rightarrow$ "Distance au soleil : proche".

L'ensemble d'implication dans un contexte formel $\mathbb{K} = (G, M, I)$ peut être déduit directement à partir du treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$ et on a : $P \rightarrow Q$ est une implication dans $\mathbb{K}$ si le concept (A, B) le plus général vérifiant $Q \subseteq B$ vérifie aussi $P \subseteq B$. Étant donné cette constatation, le diagramme de Hasse d'un treillis de concepts avec étiquetage réduit permet une lecture directe de l'ensemble minimal non redondant de toutes les implications du contexte qui ont un support non nuls (les attributs dans $P \cup Q$ sont possédés par au moins un objet dans G). Cet ensemble est dit minimal non redondant car aucune implication ne peut être déduite en combinant deux ou plusieurs autres règles et à partir de cet ensemble on peut déduire toutes les implications possibles dans le contexte. La figure 3.5 montre l'ensemble d'implications minimales à supports positifs déduites à partir des concepts du treillis correspondant au contexte des planètes du système solaire. Dans le cas où on considère aussi les implications à support nul, l'ensemble d'implications minimales non redondantes du contexte est appelé **base d'implications** et noté

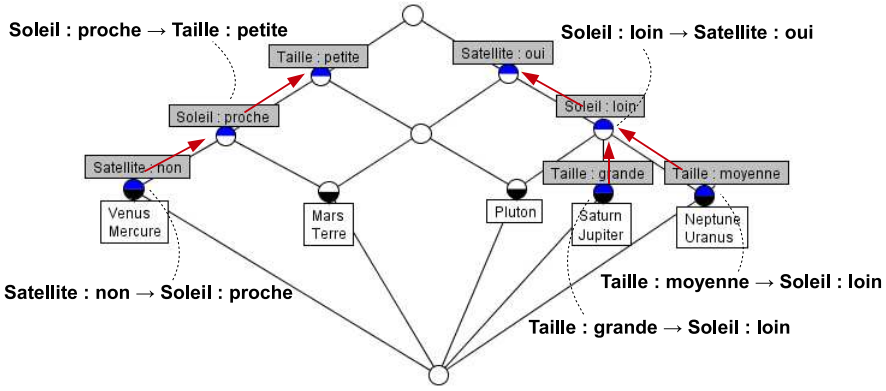


FIGURE 3.5 – Les implications minimales non redondantes à support non nul déduites directement à partir du treillis de concepts.

par $\mathcal{I}(\mathbb{K})$. La base d'implications du contexte des planètes du système solaire est constitué des implications suivantes :

"Satellite : non"	→	"Soleil : proche"
"Soleil : proche"	→	"Taille : petite"
"Taille : grande"	→	"Soleil : loin"
"Taille : moyenne"	→	"Soleil : loin"
"Soleil : loin"	→	"Satellite : oui"
"Taille : petite", "Taille : moyenne"	→	"Soleil : loin", "Satellite : non"
"Taille : petite", "Taille : grande"	→	"Taille : moyenne", "Satellite : non"
"Taille : moyenne", "Taille : grande"	→	"Satellite : non"
"Soleil : proche", "Soleil : loin"	→	"Taille : moyenne", "Taille : grande", "Satellite : non"
"Satellite : oui", "Satellite : non"	→	"Taille : moyenne", "Taille : grande"

3.4 ACF et découverte de ressources

L'ACF s'appuie sur un formalisme mathématique rigoureux, la théorie des treillis, et l'adapte afin de permettre d'analyser des données, sous la forme d'un contexte formel, et de les structurer en un treillis de concepts. La structure du treillis facilite la visualisation des relations sous-jacentes entre les données et l'interprétation de ses relations. Le fondement mathématique de l'ACF et les structures conceptuelles qu'elle permet de dériver [Godin et al., 1995b] ont été exploités dans plusieurs domaines d'analyse et d'exploitation de données tels que la classification, la recherche d'information [Carpineto and Romano, 2005], la selection de services Web [Azmeh et al., 2008], la construction d'ontologies [Bendaoud et al., 2008], l'extraction de connaissances [Lakhal and Stumme, 2005], l'apprentissage machine [Kuznetsov, 2005], l'ingénierie des logiciels [Tilley et al., 2005, Godin and Valtchev, 2005], la linguistique [Priss, 2005], etc. Les principales approches développées dans le cadre de ces domaines sont décrites en détails dans [Ganter et al., 2005a]. Dans la suite de cette section nous nous intéressons plus particulièrement aux applications de l'ACF à la découverte de ressources.

3.4.1 Principales motivations

L'utilisation des treillis de concepts pour la découverte de ressources est motivée par deux principales caractéristiques liées aux treillis de concepts : la structuration conceptuelle des données et l'ordre hiérarchique entre les concepts.

La structuration conceptuelle

Dans un treillis de concepts les données sont structurées sous forme de concepts. Un concept peut être vu comme une classe d'objets (l'extension du concept) caractérisée par un ensemble de propriétés (l'intension du concept). Dans l'optique de découverte de ressources, les concepts du treillis fournissent une première classification des objets dans des classes homogènes qui répondent chacune à un ensemble de critères bien déterminés. Connaissant les critères que doit satisfaire les ressources à découvrir, la découverte se ramène à identifier la classe qui vérifie ces critères.

Les liens généralisation/spécialisation

Dans un treillis de concepts les concepts sont ordonnés selon deux critères duaux liés à leurs extensions et à leurs intensions (voir définition 15). Les concepts les plus généraux sont situés en haut du treillis alors que les concepts les plus spécifiques sont situés en bas du treillis. Les liens entre les concepts peuvent être interprétés comme des généralisations ou des spécialisations entres les classes représentées par les concepts. En effet, un parcours ascendant des concepts d'un treillis se traduit à chaque étape par la diminution progressive du nombre d'attributs dans les intensions des concepts et l'augmentation progressive du nombre d'objets dans leurs extensions. Cela correspond au passage d'une classe plus spécifique, qui contient peu d'objets qui vérifient plusieurs critères, à une classe plus générale, qui contient plus d'objets qui ne vérifient qu'une partie des critères de la classe spécifique. De façon duale, un parcours descendant des concepts d'un treillis correspond au passage d'une classe générale à une classe plus spécifique.

3.4.2 ACF et Recherche d'Information

La recherche d'information (RI) a été l'une des premières applications phare de treillis de concepts à la découverte de ressources. Les premiers travaux [Carpineto and Romano, 1993, Godin et al., 1993, Godin et al., 1995b] ont étudié la possibilité d'utiliser les treillis de concepts comme support pour la recherche documentaire. Des collections de documents sont alors représentées sous la forme de contextes formels. Les objets du contexte sont des documents et les attributs sont les termes d'indexation de ces documents. Chaque concept du treillis correspondant est vu comme un couple formé par une requête, dont les mots clés sont les termes contenus dans l'intension du concept, et l'ensemble de documents pertinents pour cette requête sont les documents contenus dans l'extension du concept. Le critère de pertinence dans ce cas étant celui considéré dans le cas de la recherche booléenne (conjonctive) à savoir la vérification de tous les critères spécifiés dans la requête. Ceci justifie l'interprétation donnée aux concepts du treillis puisque les objets dans l'extension d'un concept partagent tous les attributs dans son intension.

Le calcul de la réponse à une requête donnée revient à identifier, dans le treillis, le concept dont l'intension est identique à la requête. Les liens de spécialisation/généralisation entre les concepts permettent d'effectuer une recherche progressive dans le treillis. Cette façon de procéder suppose que la requête existe déjà dans le treillis. Pour assurer cette condition, des algorithmes de construction incrémentale de treillis de concepts sont utilisés pour l'insertion des requêtes dans un treillis déjà construit. De cette manière, un premier mode de recherche par treillis a été

défini : la recherche par interrogation. La structure hiérarchique des treillis de concepts permet la définition d'un deuxième mode de recherche : la recherche par navigation. Ces deux modes sont détaillés dans les sections suivantes.

Interrogation

Ce mode de recherche est facilité par la mise en place d'algorithmes performants pour la construction incrémentale des treillis de concepts. La définition de requête consiste à spécifier directement les termes d'indexation qui décrivent le(s) document(s) à trouver. La requête est ensuite insérée dans le treillis. La recherche des documents pertinents revient à localiser le concept le plus général incorporant les termes spécifiés dans la requête.

Navigation

Ce mode de recherche exploite la structure hiérarchique des treillis. Il consiste à explorer librement les concepts en s'appuyant sur la visualisation des treillis par des diagrammes de Hasse. Cette forme d'interaction tire profit d'une caractéristique importante de la cognition humaine : "il est plus facile de reconnaître quelque chose d'intéressant que de le décrire". Le diagramme de Hasse est utilisé comme structure de base pour la recherche. Il offre une interface de navigation permettant de suivre les liens de spécialisation/généralisation entre concepts pour spécialiser ou élargir graduellement l'espace de recherche. Dans le cas général, le scenario de recherche peut être présenté comme suit : partant du concept le plus général du treillis qui représente la classe de tous les documents avec un ensemble de termes communs souvent vide, on effectue une spécialisation graduelle en suivant les liens descendants dans le treillis. Chaque pas dans le treillis est equivalent à l'ajout d'un nombre minimal de nouveaux termes qui spécifient la description des documents à trouver. Ceci entraîne la restriction de l'espace de recherche à un sous ensemble de documents. Cette opération est répétée jusqu'à l'identification du sous ensemble minimal de documents recherchés. Dans d'autres cas, partant d'une description très précise de documents, on peut relâcher progressivement cette description en suivant les lien ascendants du treillis. Chaque étape correspond à la suppression d'un ensemble minimal de termes ce qui entraîne l'augmentation du nombre de documents qui satisfont la description allégée.

De manière générale, on peut effectuer une navigation libre dans le treillis en suivant les liens descendants et/ou ascendants en fonction du besoin en précision lors du passage d'un concept à l'autre dans le treillis.

La combinaison des deux modes

L'utilisation des treillis de concepts permet de définir deux modes de recherche, par interrogation et par navigation, qui peuvent être combinés. Le treillis de concepts sert d'espace de recherche commun et assure la cohérence des résultats de recherche des deux modes. De plus, le passage d'un mode à l'autre peut se faire à tout moment pendant la recherche. La recherche se ramène ainsi à effectuer une combinaison libre de (i) spécification directe de requête, résultant en un saut dans le concept le plus générale incorporant les termes de la requête, et (ii) la navigation libre en suivant les liens entre les concepts du treillis.

Classement des résultats de la recherche par treillis

Dans les premières approches de recherche d'information par treillis de concepts citées plus hauts, les treillis sont utilisés dans la perspective de la recherche booléenne. Une requête consiste

en une conjonction de termes. Les documents pertinents sont ceux indexés par tous les termes de la requête. Plus tard, de nouveaux critères de pertinence ont été défini. En conséquence, des documents n'apparaissant pas dans le même concept que la requête deviennent à leurs tours pertinents. Cependant leur pertinence est inférieure à ceux dans le concept représentant la requête. De ce fait il est essentiel de pouvoir classer l'ensemble de documents en réponses de façon à refléter cette différence de pertinence. La structure hiérarchique des treillis a été exploitée pour déduire automatiquement un tel classement. Dans [Carpineto and Romano, 2000], Carpineto et Romano proposent une approche qui s'appuie sur la distance entre les concepts dans le treillis pour calculer la similarité entre les documents et la requête. L'idée est la suivante : les documents en extension du concept représentant la requête dans le treillis sont classés en première position. Les documents dans les extensions des concepts joignables en un seul pas à partir de la requête sont classés en deuxième position. Les documents classés dans les positions supérieurs sont obtenus de la même façon en incrémentant le nombre de pas dans le treillis à chaque fois.

Approches récentes de recherche d'information par treillis

Pendant les dernières années, les approches de recherche d'information par treillis ont donné lieux à des méta-moteurs de recherche d'information. Ces méta-moteurs se placent entre les utilisateurs et les moteurs de recherche tels que Google et Yahoo. Leur rôle principal est d'améliorer la présentation des résultats de recherche aux utilisateurs. Le fonctionnement général est le suivant. Tout d'abord, une requête utilisateur est passée à un moteur de recherche. Le résultat retourné par le moteur de recherche, sous forme de liens vers des documents et un ensemble de "synsets" (mots décrivant le document : souvent une ou deux lignes du document qui contiennent un ou plusieurs mots clés de la requête), est transformé sous la forme d'un contexte formel. Les objets sont les liens et les attributs sont formés par l'ensemble des éléments des synsets utilisés. Après construction du treillis correspondant à ce contexte, on propose à l'utilisateur le nouveau classement du résultat fourni par le treillis. Ce type d'approche est implémenté dans plusieurs systèmes opérationnels tels que CREDO [Carpineto and Romano, 2004b], CREDINO [Carpineto et al., 2006], FooCA [Koester, 2006], Insighter [Jaoua, 2007], CRECHAINDO [Nauer and Toussaint, 2007] et SearchSleuth [Ducrou and Eklund, 2007].

CREDO est le premier système proposé. Après classification du résultat d'une requête sous la forme d'un treillis de concept, il affiche la partie supérieur du treillis (Iceberg) sous la forme d'une hiérarchie de liens similaire à celle des hierarchies de fichiers dans les systèmes hiérarchiques. Chaque nœud de la hiérarchie est un mot qui indexe un certain nombre de documents dans le résultat. Ce nombre est aussi indiqué avec le mot clé. Le système Credino est une adaptation dy système CREDO pour les PDA.

Le Système FooCA est le seul système à visualiser le résultat retourné par le moteur de recherche, Google, sous la forme d'un contexte formel (une matrice) et le diagramme de Hasse du treillis correspondant à ce contexte. A travers cette visualisation, FooCA permet à l'utilisateur d'apporter ses modifications au contexte en supprimant des mots clés qui lui paraissent inutiles pour sa recherche. Les modifications apportées sont ensuite propagées au treillis pour la visualisation de la nouvelle classification.

Le système CreChainDo reprend le même principe et les même fonctionnalités que CREDO et leurs ajoute la possibilité d'interagir et de modifier le contexte en y incluant les retours de l'utilisateur. Il s'agit des fonctionnalités similaires à celle de FooCA mais avec soumission d'une nouvelle requête à Google et reconstruction du treillis après chaque modification apportée au contexte.

Le système Insighter diffère des autres par la représentation de chaque concept du treillis

par un seul mot, le plus représentatif de l'intension du concept. Le choix du mot représentatif s'appuie sur une heuristique qui consiste à construire un graphe biparti complet [23] (ou biclique) entre les mots utilisés dans les documents et à effectuer la pondération des arcs du graphe selon les co-occurrences des pairs de mots. Le mot le plus représentatif d'un ensemble de mots est celui dont les liens ont les poids maximaux.

Le système SearchSleuth fournit une catégorisation supplémentaire du résultat en exploitant les relations entre le concept requête et les autres concepts du treillis. Cette catégorisation consiste à répartir les concepts du treillis sur quatre ensembles. Le premier ensemble, affiché comme résultat principal, est formé par les liens aux documents qui forment l'extension du concept requête. Le deuxième et le troisième ensemble sont formés respectivement par les sous concepts et les super-concepts de la requête dans le treillis. Le quatrième ensemble est formé par la fratrie du concept requête dans le treillis, les concepts ayant un parent commun avec la requête. Contrairement au premier ensemble, les autres ensembles ne sont pas affichés mais sont accessibles via des liens de raffinement en fonction du besoin en spécialisation, généralisation ou la combinaison des deux.

Recherche dans des domaines spécifiques

En plus des systèmes de recherche d'information sur le web cités dans la section précédente, d'autres approches de découvertes de ressources par treillis de concepts ont aussi été définies pour la découverte de ressources dans des domaines particuliers ou encore pour la découverte de types particuliers de ressources. Parmi ces approches nous pouvons mentionner les systèmes de gestion de messagerie électronique HireMail et MailSleuth [II et al., 2003], les systèmes de recherche d'images Image-Sleuth [Ducrou and Eklund, 2008], Camelis [24] [Ferré, 2007] et de séquences vidéo [Mimouni and Slimani, 2006] et le système de recherche de bugs dans des fichiers de code source JAVA [Tilley et al., 2005, Cellier, 2008]. Le principe général de ces système est le même que ceux de la recherche d'information sur le web. Cependant, la spécificité de chaque système provient de sa définition de la pertinence qui peut dépendre de la particularité du domaine étudié.

3.5 ACF et données complexes

3.5.1 Contexte formel multivalué

Les approches qui s'appuient sur l'ACF sont définies pour l'analyse et l'exploitation des données du monde réel. Ces données ne se présentent pas forcément sous la forme de contextes binaires. Par exemple, dans le cas de la recherche documentaire, une façon plus précise de représenter la relation entre les documents et leurs termes d'indexation est de considérer les fréquences d'apparition des termes dans les documents au lieu de la simple information binaire indiquant la présence ou l'absence d'un terme dans un document. Les relations à valeurs non binaires entre les objets et les attributs caractérisent aussi les données contenues dans les bases de données relationnelles qui restent l'un des formats les plus utilisés pour la stockage de données. La représentation formelle de ces données ne peut pas se faire par de simples contextes formels. Elle sont plutôt représentées sous la forme de contextes formels multivalués définis comme suit.

23. Un graphe est dit biparti s'il existe une partition de son ensemble de sommets en deux sous-ensembles U et V telle que chaque arête ait une extrémité dans U et l'autre dans V. Un graphe biparti est dit biparti complet (ou biclique) si chaque nœud de U est relié à chaque nœud de V.

24. Camelis utilise une extension de l'ACF appelée l'Analyse Logique de Concepts détaillée à la fin de ce chapitre.

TABLE 3.5 – Mesures réelles relatives aux planètes du système solaire.

	Diamètre (km)	Distance au Soleil (10^6km)	Satellite
Mercure	4 878	58	0
Venus	12 400	108	0
Terre	12 756	150	1
Mars	6 800	228	2
Jupiter	142 800	778	16
Saturne	120 800	1 427	19
Uranus	47 600	2 870	5
Neptune	44 600	4 500	8
Pluton	2.320	9 950	1

Définition 18 (Contexte formel multivalué) *Un* ***contexte formel multivalué*** *est un quadruplet (G, M, W, I) où G est un ensemble d'objets, M est un ensemble d'attributs multivalués, W est l'ensemble de valeurs prises par les attributs et $I \subseteq G \times M \times W$ une relation ternaire entre G, M et W, telle que :*

$$(g, m, w) \in I \text{ et } (g, m, v) \in I \text{ implique } w = v.$$

Les notations $(g, m, w) \in I$ et $I(g, m) = w$ sont équivalentes et expriment que l'attribut m a la valeur w pour l'objet g.

Un contexte formel multivalué, ou simplement contexte multivalué, permet une représentation différente des données qu'un contexte formel (appelé aussi contexte monovalué). Dans le cas des données relatives au système solaire, il est possible de représenter des valeurs de mesures réelles des diamètres des planètes ou de leurs distances au soleil au lieu des simples attributs qualificatifs indiquant qu'une planète est grande ou petite et qu'elle est proche ou loin du soleil. Les valeurs réelles des mesures relatives aux planètes du système solaire représenté par le contexte formel donné dans la table 3.1 peut être représenté avec plus de précision sous la forme d'un contexte multivalué. Dans ce cas, les cases du tableau prennent les mesures précises des attributs pour chaque planète. Le contexte multivalué obtenu est donné dans la table 3.5.

L'application des résultat de l'ACF à un contexte multivalué nécessite la transformation de celui-ci en un contexte monovalué. Cette étape de transformation s'appelle **échelonnage conceptuel** [Ganter and Wille, 1999].

3.5.2 Échelonnage conceptuel

L'échelonnage conceptuel (*conceptual scaling*) consiste à transformer chaque attribut multivalué en un ensemble d'attributs binaires (monovalués) qui forment un contexte monovalué appelé **échelle conceptuelle** (*conceptual scale*) de l'attribut multivalué. Une échelle conceptuelle est un contexte formel dont les objets sont des valeurs et les attributs sont des attributs d'échelle. Ce contexte permet de structurer le domaine de valeurs de cet attribut sous la forme d'un treillis de concepts qui définit une hiérarchie entre les attributs d'échelle.

Reprenons l'exemple de contexte multivalué des planètes du système solaire donné dans la table 3.5. Une échelle conceptuelle possible pour l'attribut "*Distance au soleil*" est donné dans

la table 3.6 [Carpineto and Romano, 2004a]. Les attributs binaires de l'échelle dans cet exemple sont "≥ 10", "≥ 100", "≥ 500" et "≥ 1000". De la même manière on peut définir des échelles pour les attributs multivalués "Diamètre" et "Satellite(s)". Les échelles conceptuelles correspondantes aux deux attributs sont données dans les tables 3.7 et 3.8, respectivement.

TABLE 3.6 – Échelle conceptuelle de l'attribut "*Distance au soleil*" et le treillis correspondant.

	$ds \geq 10$	$ds \geq 100$	$ds \geq 500$	$ds \geq 1000$
58	x			
108	x	x		
150	x	x		
228	x	x		
778	x	x	x	
1 427	x	x	x	x
2 870	x	x	x	x
4 500	x	x	x	x
9 950	x	x	x	x

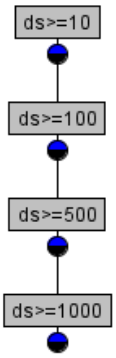


TABLE 3.7 – Échelle conceptuelle de l'attribut "*Diamètre*" et le treillis correspondant.

	$d \leq 10$	$10 < d \leq 20$	$20 < d \leq 50$	$d \geq 50$
4 878	x			
12 400		x		
12 756		x		
6 800	x			
142 800				x
120 800				x
47 600			x	
44 600			x	
2.320	x			

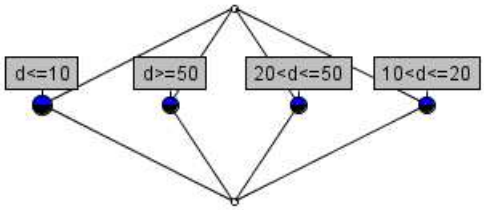
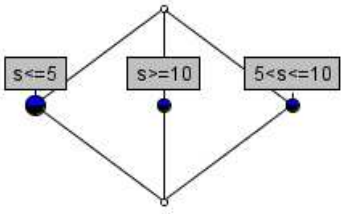


TABLE 3.8 – Échelle conceptuelle de l'attribut "*Satellite*" et le treillis correspondant.

	$s \leq 5$	$5 < s \leq 10$	$s \geq 10$
0	x		
0	x		
1	x		
2	x		
16			x
19			x
5	x		
8		x	
1	x		



La transformation n'est pas unique et ne suit pas une règle générale de transformation. Elle dépend d'une interprétation particulière de l'attribut multivalué et des valeurs qu'il prend dans le

contexte multivalué. Cependant, Ganter et Wille ont définit une quinzaine d'échelonnages conceptuels typiques pour des attributs multivalués de formats particuliers [Ganter and Wille, 1999]. Considérant ces définitions, l'échelle conceptuelle de l'attribut multivalué "*Distance au soleil*" est obtenue en effectuant un échelonnage **ordinal** alors que celles de "Diamètre" et "Satellite(s)" sont obtenues en effectuant un échelonnage **nominal** [Ganter and Wille, 1999].

Le contexte monovalué est obtenu à partir du contexte multivalué, en fonction des échelles conceptuelles de chaque attribut. L'ensemble des objets est conservé et l'ensemble des attributs est l'union disjointe des attributs des échelles conceptuelles. Une fois que le contexte monovalué est obtenu, on peut construire le treillis de concepts correspondant par l'une des méthodes de construction de treillis détaillé dans la section 3.3.5.

Le contexte monovalué obtenu à partir du contexte multivalué des planètes du système solaire est donné dans la table 3.9. Le treillis de concepts correspondant à ce contexte est donné dans la figure 3.6.

TABLE 3.9 – Contexte monovalué résultant de l'échelonnage conceptuel du contexte multivalué des planètes du système solaire.

	Diamètre (km)				Distance au Soleil (10^6km)				Satellite		
	$d \leq 10$	$10 < d \leq 20$	$20 < d \leq 50$	$d > 50$	$ds \geq 10$	$ds \geq 100$	$ds \geq 500$	$ds \geq 1000$	$s \leq 5$	$5 < s \leq 10$	$s > 10$
Mercure	x				x				x		
Venus		x			x	x			x		
Terre		x			x	x			x		
Mars	x				x	x			x		
Jupiter				x	x	x	x				x
Saturne				x	x	x	x	x			x
Uranus			x		x	x	x	x	x		
Neptune			x		x	x	x	x		x	
Pluton	x				x	x	x	x	x		

3.6 Les extensions de l'ACF

L'utilisation de l'échelonnage conceptuel permet d'élargir les domaines d'application de l'ACF à des données qui peuvent se présenter sous la forme de contextes multivalués. De cette manière plusieurs approches d'analyse de données ont pu profiter de la richesse du formalisme de l'ACF en présence de données complexes. Cependant, en l'absence de règles générales de transformation, le passage d'un contexte multivalué à un contexte monovalué est souvent confronté aux problèmes suivants. D'une part le choix des échelles est souvent difficile à automatiser et nécessite beaucoup d'expertise vis-à-vis des données. D'autre part le passage d'un attribut multivalué à des attributs binaires entraîne la perte des relations entre les données ainsi que l'augmentation de la taille du contexte monovalué obtenu. Ces deux problèmes sont souvent aggravés par la complexité des données et leurs grandes quantités dans le cadre des applications réelles.

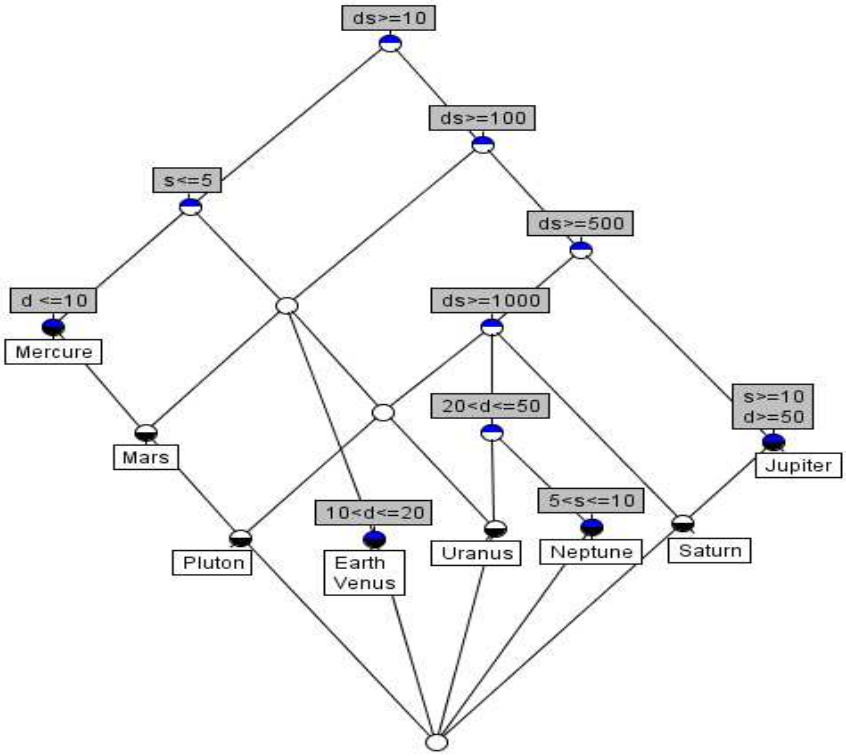


FIGURE 3.6 – Le treillis de concepts correspondant au contexte formel donné dans le tableau 3.9.

Pour pallier ces limites, plusieurs travaux de recherche ont été menés dans le but d'étendre les definitions de l'ACF pour couvrir les données complexes. Cependant, la diversité des données étudiées et les caractéristiques particulières de chaque domaine d'application étudié dans le cadre de ces travaux ont abouti à la définition de différentes extensions. Chacune de ces extensions est adaptée à un format particulier des données et une interprétation particulière des attributs multivalués et de leurs valeurs dans les contextes multivalués. Le point commun de ces approches est qu'elle redéfinissent la connexion de Galois entre les objets et les attributs en vue de l'adapter à la particularité du domaine étudié.

3.6.1 Analyse de Concepts Formels Flous

L'Analyse de Concepts Formels Flous (ACFF) consiste à étendre les résultats de l'ACF aux contextes flous. Un contexte flou est un contexte multivalué où les valeurs de la relation objet×attribut représentent un degré de vérité pour lequel l'objet est en relation avec l'attribut. De nombreux travaux ont été effectués –souvent indépendamment les un des autres– dans le cadre de l'ACFF. Une étude comparative des principales approches obtenues est présentée dans [Belohlávek and Vychodil, 2005]. Dans cette section nous allons nous restreindre à la discussion de deux approches : celle introduite par Ben Yahia et Jaoua [Yahia and Jaoua, 2001] et celle introduite par Belohlavek [Belohlavek, 2002]. Ces deux approches sont représentatives de deux familles de propositions dites **mono-facette** et **bi-facette** ou **générale**. Les autres approches sont des formalisations analogues ou contenues dans l'une des deux approches. En dehors de ces deux familles d'approches qui adaptent les définitions de l'ACF aux contextes flous, il existe d'autres propositions qu'on appellera "à seuil(s)" et qui effectuent une transformation

particulière du contexte flou en un contexte binaire puis appliquent les résultats de l'ACF au contexte obtenu.

Méthodes à seuil(s)

Dans [Ma et al., 2006], la méthode dite "*à seuil variable*" consiste à fixer un seuil δ au départ puis transformer le contexte flou en un contexte monovalué en enlevant toutes les valeurs inférieures à δ et en remplaçant par "$\times$" (pour indiquer la relation entre l'objet et l'attribut correspondant de la même manière que dans le cas de l'ACF classique) celles supérieures à δ. Dans la méthode dite "*à fenêtre*" [Zhou et al., 2007], deux bornes sont considérées à la fois et définies pour chaque attribut : une borne supérieure et une borne inférieure. Le contexte multivalué est transformé en contexte monovalué en enlevant les valeurs qui ne sont pas entre les deux bornes et en remplaçant celles qui le sont par "$\times$". Dans [Quan et al., 2004] l'idée est similaire à celle de la méthode "*à seuil variable*" dans la mesure où un seuil est choisi et le contexte multivalué est transformé en contexte monovalué. La différence réside dans le fait qu'une fois le treillis de concepts construit à partir du nouveau contexte monovalué, les valeurs supérieures aux seuils fixés sont considérées à nouveau pour être utilisées comme poids des attributs dans les intensions des concepts du treillis. D'autres méthodes de discrétisation similaires sont détaillées dans [Pensa et al., 2004].

Approches "mono-facette"

Dans [Yahia and Jaoua, 2001], un contexte flou est défini comme suit.

Définition 19 *Un* ***contexte flou*** *est un triplet* (X, Y, I) [25] *où* X *est un ensemble d'objets,* Y *est un ensemble d'attributs flous et* I *une relation binaire floue (*$I : X \times Y \rightarrow [0, 1]$*). Chaque couple* $(x, y)^\alpha \in I$ *signifie que l'objet* $x \in X$ *a l'attribut* $y \in Y$ *à un degré supérieur ou égal à* $\alpha \in [0, 1]$.

La table 3.10 donne un exemple de contexte flou (l'exemple est pris de [Yahia and Jaoua, 2001] avec renommage des objets et des attributs). Un exemple de sous ensemble flou de Y est $\{a^{0.5}, b^1, c^{0.7}, d^{0.5}\}$.

Pour la définition des opérateurs d'union, d'intersection, d'inclusion, etc. entre parties de Y les auteurs ont repris les opérateurs classiques entre les ensembles flous [Zadeh, 1965]. Étant donné ces opérateurs, l'ensemble des parties de Y, $\mathfrak{P}(Y)$, est un ensemble partiellement ordonné et la relation d'ordre partiel entre les éléments de $\mathfrak{P}(Y)$ est l'inclusion entre ensembles flous. Les sous ensembles de X sont considérés de la même façon que dans le cas des contextes monovalués.

Les opérateurs de dérivation entre $\mathfrak{P}(X)$ et $\mathfrak{P}(Y)$ sont définis comme suit [26] :

$$A^\uparrow = \{y^\alpha \in Y \mid \forall x \in A, \alpha = min(I(x, y))\}$$

$$B^\downarrow = \{x \in X \mid \forall y^\alpha \in B, I(x, y) \geq \alpha\}$$

Dans l'exemple de contexte flou donné dans la table 3.10 on a : $\{x1, x2\}^\uparrow = \{a^{0.5}, b^{0.7}, c^{0.7}, d^{0.5}\}$ et $\{a^{0.5}, b^{0.7}, c^{0.7}, d^{0.5}\}^\downarrow = \{x1, x2\}$.

Les opérateurs $^\uparrow$ et $^\downarrow$ forment une connexion de Galois entre $\mathfrak{P}(X)$ et $\mathfrak{P}(Y)$ [Yahia and Jaoua, 2001] et sont à la base de la définition de concepts formels flous et de treillis de concepts flous. Un

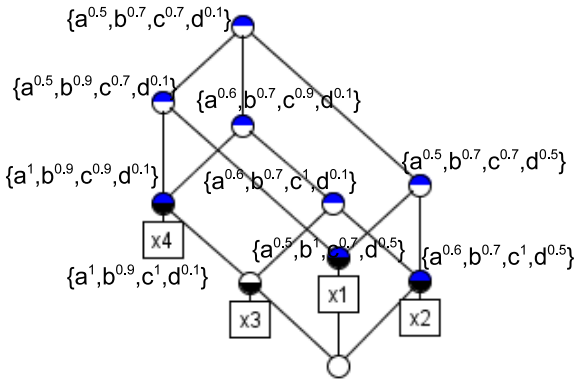
25. Cette notation est utilisé pour l'homogénéité avec l'approche suivante. Dans [Yahia and Jaoua, 2001] les auteurs utilisent plutôt (O, I, R) (O pour Objets, I pour Items et R pour Relation).

26. La notation des opérateurs de dérivation est celle utilisée dans [Belohlávek and Vychodil, 2005]. Dans [Yahia and Jaoua, 2001] on utilise f au lieu de $^\uparrow$ et h au lieu de $^\downarrow$.

concepts formel flou est défini de la même façon qu'un contexte formel dans le cas de l'ACF comme un couple (A, B) tel que $A \in \mathfrak{P}(X)$, $B \in \mathfrak{P}(Y)$, $A^{\uparrow} = B$ et $B^{\downarrow} = A$. Les concepts formels flous sont organisés en **treillis de concepts flous** en s'appuyant sur la relation super-concept/sous-concept définie de façon similaire à celle dans le cadre de l'ACF. Le treillis de concepts flous correspondant au contexte flou donné dans la table 3.10 est aussi donné dans la table 3.10 (à droite).

TABLE 3.10 – Un exemple de contexte flou et le treillis de concepts flous correspondant selon la méthode de Ben Yahia et Jaoua.

	a	b	c	d
x1	0.5	1	0.7	0.5
x2	0.6	0.7	1	0.5
x3	1	0.9	1	0.1
x4	1	0.9	0.9	0.1



Dans les approches mono-facettes, l'extension d'un concept formel flou est un ensemble d'objets alors que son intension est un ensemble flou d'attributs. Ceci exprime que les valeurs dans un contexte formel flou sont utilisés comme degrés de vérités qui s'appliquent uniquement aux attributs. Cette façon de considérer les contextes flous est à l'origine de l'appellation "mono-facette" de ces approches [Belohlávek and Vychodil, 2005].

Approches générales

Dans [Belohlávek, 1999, Belohlavek, 2002], la définition de contexte flou est plus générale que celle dans le cas mono-facette. La généralisation consiste à considérer que l'ensemble des objets est lui aussi un ensemble flou et que l'ensemble des valeurs de vérité (l'intervalle [0,1] dans l'approche précédente) est un ensemble L quelconque (souvent dans les exemples considérés $L \subseteq [0, 1]$). Pour assurer la généricité de l'approche, Belohlavek propose un cadre formel pour l'ensemble L qui soit une généralisation du cas binaire (lorsque $L = \{0, 1\}$, on retrouve les résultats de l'ACF). Et comme la généralisation vise à étendre les résultat de l'ACF aux contextes flous, Belohlavek propose de considérer l'equivalent des opérateurs de la logique classique dans la logique floue. Le point de départ consiste à considérer que l'ensemble $(L, \leq)$ est partiellement ordonné et a respectivement 0 et 1 comme plus petit et plus grand élément. $(L, \wedge, \vee, 0, 1)$ est donc un treillis complet ayant 1 comme top et 0 comme bottom. L'opérateur de conjonction, entre les éléments de L, qui généralise la conjonction classique est noté par $\otimes$. Cet opérateur vérifie, $1 \otimes 1 = 1$ et $1 \otimes 0 = 0 \otimes 1 = 0 \otimes 0 = 0$. Tout comme la conjonction classique, l'opérateur $\otimes$ est commutatif et associatif et a 1 comme élément neutre. Par définition, les propriétés précédentes signifient que $(L, \otimes, 1)$ est un monoide commutatif. La généralisation de la règle d'inference *Modus ponens* dans le cas de la logique floue aboutit à la définition de l'opérateur d'inclusion "$\rightarrow$" qui étend l'implication de la logique classique "$\Rightarrow$". Les deux opérateurs $\otimes$ et $\rightarrow$ sont dits opérateur de conjonction flou et opérateur d'implication flou, respectivement, et vérifient pour tout triplet

$a, b, c \in L$,

$$a \otimes b \leq c \quad \text{si et seulement si} \quad b \leq a \rightarrow c.$$

Étant donné toutes ces propriétés, la structure algébrique formée par l'ensemble des valeurs de vérité, L, et les opérateurs définis sur L est $\mathbf{L} = (L, \wedge, \vee, \otimes, \rightarrow, 0, 1)$. $\mathbf{L}$ est une algèbre appelée **treillis résidué** et l'ensemble L est appelé support de $\mathbf{L}$.

Étant donné une structure $\mathbf{L}$ de valeurs de vérité, un contexte flou, appelé aussi **L-contexte** par Belohlavek, est défini comme suit.

Définition 20 *Un **contexte flou** ou **L-contexte** est un triplet (X, Y, I) où X est un ensemble d'objets flous, Y est un ensemble d'attributs flous et I une relation binaire floue ($I : X \times Y \rightarrow L$) dite aussi **L-relation** entre X et Y. La valeur $I(x, y) \in L$ est interprétée comme le degré de vérité pour lequel l'objet $x \in X$ a l'attribut $y \in Y$.*

Les opérateurs de dérivation sont définis entre les ensembles flous d'objets, notés par L^X, et les ensembles flous d'attributs, notés par L^Y, comme suit pour tous sous ensembles flous $A \in L^X$ et $B \in L^Y$.

$$A^{\uparrow}(y) = \bigwedge_{x \in X} (A(x) \rightarrow I(x, y))$$

$$B^{\downarrow}(x) = \bigwedge_{y \in Y} (B(y) \rightarrow I(x, y))$$

$A^{\uparrow}(y)$ étant la valeur de vérité de "l'attribut y est partagé par tous les objets dans A" et $B^{\downarrow}(x)$ est la valeur de vérité de "l'objet x possède tous les attributs dans B".

Un **concept flou** ou **L-concept** est défini de la même façon que dans le cas de l'ACF comme un couple (A, B) tel que $A \in L^X$, $B \in L^Y$, $A^{\uparrow} = B$ et $B^{\downarrow} = A$. A est l'extension du **L**-concept et B est son intension. Les **L**-concepts d'un contexte flou sont organisés par l'intermédiaire d'une relation d'ordre partiel définie de la même manière que dans le cas de l'ACF en remplaçant les inclusions entre ensemble d'éléments discrets par les inclusions entre ensembles flous. Étant donné deux **L**-concepts, $(A1, B1)$ et $(A2, B2)$, $(A1, B1)$ est dit un sous-concept de $(A2, B2)$ (auquel cas $(A2, B2)$ est dit super-concept de $(A1, B1)$) et on note $(A1, B1) \leq (A2, B2)$ si et seulement si $A1 \subseteq A2$ (ou de manière équivalente $B2 \subseteq B1$). L'ensemble des **L**-concepts d'un **L**-contexte organisé par l'intermédiaire de la relation d'ordre partiel "$\leq$" forme un **treillis de concepts flous** ou **L-treillis** noté par $\mathfrak{B}(X, Y, I)$.

Les **L**-concepts et le **L**-treillis correspondant à un **L**-contexte dépendent du choix de la structure **L**.

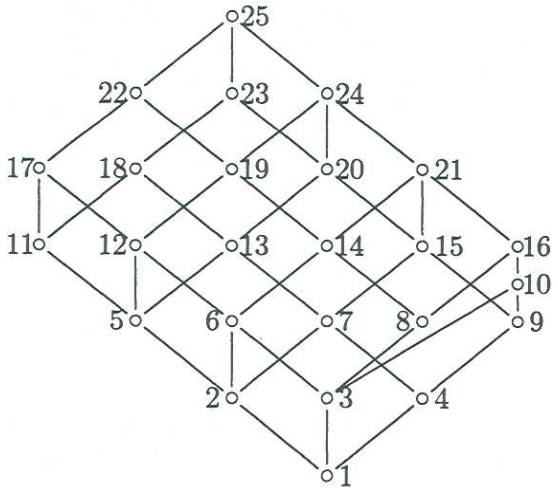
3.6.2 Analyse de Concepts Logiques

Les extensions logiques de l'ACF

Les extensions logiques de l'ACF peuvent être réparties en deux familles. La première consiste à utiliser la logique pour définir des échelonnages de contextes multivalués à fin de les transformer en contextes binaires et de pouvoir appliquer les résultats de l'ACF par la suite. Cette famille contient les travaux de Prediger [Prediger, 1997, Prediger and Stumme, 1999] qui a définit l'échelonnage logique (logical scaling) pour les contextes multivalués. Le principe de l'échelonnage logique est d'utiliser un langage formel pour générer des prédicats unaires à partir des attributs et des valeurs d'un contexte multivalué. Cette approche peut être vue comme un cas particulier d'échelonnage conceptuel (section 3.5.2).

TABLE 3.11 – L-contexte donné par les planètes et leurs propriétés.

	Taille		Distance au Soleil	
	petit (tp)	grand (tg)	loin (dl)	proche (dp)
Mercure	1	0	0	1
Vénus	1	0	0	1
Terre	1	0	0	1
Mars	1	0	1/2	1
Jupiter	0	1	1	1/2
Saturne	0	0	1	1/2
Uranus	1/2	1/2	1	0
Neptune	1/2	1/2	1	0
Pluton	1	0	1	0



La deuxième famille consiste à adapter l'ACF à des contextes logiques où la relation objet/attribut se présente sous la forme d'une formule dans une logique donnée. Une première approche étudiant ce type de contexte a été proposée dans [Chaudron and Maille, 1998]. Dans cette approche, les objets des contextes étudiés sont décrits par des expressions de la logique du premier ordre et plus particulièrement par des des conjonctions de littéraux positifs appelés cubes logiques. Plus tard, Ferré a proposé dans [Ferré, 2002], une approche plus générale, appelée *Analyse de Concepts Logiques*, qui généralise les résultats de l'ACF aux contextes logiques. Contrairement à l'approche de Chaudron et Maille, la formalisation de l'ACL proposée par Ferré est indépendante de la logique choisie pour la description des objets du contexte. Celle-ci devient un paramètre à spécifier selon le domaine d'application considéré.

L'Analyse de Concepts Logiques

L'**Analyse de Concepts Logiques** (ACL) [Ferré, 2002] consiste à étendre les résultats de l'ACF aux contextes logiques. Un contexte logique est un contexte multivalué dans lequel les attributs sont des descriptions qui prennent comme valeurs des formules logiques décrivant les objets du contexte. Contrairement à l'approche de Chaudron et Maille où une logique particulière est choisie pour la description des objets, l'ACL donne une définition abstraite des logiques qui peuvent être utilisées. De manière analogue à la définition de la structure des valeurs de vérité dans le cas de l'ACFF [Belohlavek, 2002] (section 3.6.1), la définition de la logique en ACL se fait de manière à étendre naturellement les opérateurs et relations dans le cas binaire au cas logique. Formellement, une logique en ACL est définie comme suit.

Définition 21 *Une logique $\mathcal{L}$ est définie par le 6-uplet $(L, \sqsubseteq, \sqcap, \sqcup, \top, \bot)$, où*
- *L est un langage de formules logiques (correspond à $\mathfrak{P}(M)$ en ACF)*
- *$\sqsubseteq$ est la relation de subsomption sur L (correspond à $\supseteq$ sur $\mathfrak{P}(M)$ en ACF)*
- *$\sqcap$ est l'opérateur de conjonction (correspond à $\cap$ en ACF)*
- *$\sqcup$ est l'opérateur de disjonction (correspond à $\cup$ en ACF)*
- *$\top$ est la tautologie dans L (correspond à $\varnothing$ en ACL)*
- *$\bot$ est la contradiction dans L (correspond à tout l'ensemble M en ACL)*

De cette manière, la logique devient un paramètre formel dans la théorie de l'ACL. Étant donné une logique $\mathcal{L}$, un contexte logique est défini comme suit.

Définition 22 *Un **contexte logique**, noté par K, est un triplet $(\mathcal{O}, \mathcal{L}, d)$ où $\mathcal{O}$ est un ensemble fini d'objets, $\mathcal{L}$ est une logique, et d est une fonction de $\mathcal{O}$ dans $\mathcal{L}$ qui associe à chaque objet une formule décrivant les propriétés de l'objet (ou une description logique).*

Un exemple de contexte logique est donné dans la table 3.12 (gauche) [Ferré, 2002].

Les opérateurs de dérivation entre $\mathfrak{P}(\mathcal{O})$ et $\mathcal{L}$ sont définis comme suit.

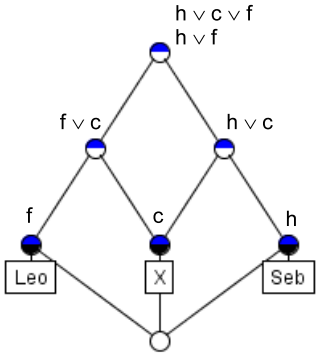
$$\sigma_K : \mathfrak{P}(\mathcal{O}) \rightarrow \mathcal{L}, \;\; \sigma_K(O) = \sqcup_{o \in O} d(o)$$

$$\tau_K : \mathcal{L} \rightarrow \mathfrak{P}(\mathcal{O}), \;\; \tau_K(f) = \{o \in \mathcal{O} \mid d(o) \sqsubseteq f\}$$

$\sigma_K(O)$ est l'expression par une formule logique des propriétés communes à tous les objets dans O et $\tau_K(f)$ est l'ensemble de tous les objets de $\mathcal{O}$ dont la description est subsumée par la formule f. Les deux opérateurs forment une connexion de Galois entre $\mathfrak{P}(\mathcal{O})$ et $\mathcal{L}$. Ils sont à l'origine de la définition de concepts logiques et du treillis de concepts logiques correspondant à un contexte logique donné. Un **concept logique** est une paire (O, f) telle que $\sigma_K(O) = f$ et $\tau_K(f) = O$. O et f sont respectivement l'extension et l'intension du concept. Les concepts logiques peuvent être ordonnés selon l'inclusion entre leurs extensions ou de manière équivalente selon la subsomption entre leurs intensions. L'ensemble des concepts logiques d'un contexte logique ordonnés de cette façon forme un **treillis de concepts logiques**. Le treillis de concepts logiques correspondant au contexte logique donné dans la table 3.12 (gauche) est donné dans la même figure à droite. Les nœuds du diagramme de Hasse de ce treillis sont étiquetés selon l'étiquetage réduit des concepts logiques [Ferré, 2002].

TABLE 3.12 – Un exemple de contexte logique (gauche) et le treillis de concepts logiques correspondant (droite). Les lettres h, f et c sont les abbreviations respectives de *homme*, *femme* et *chauve*.

objet	description
Seb	h
Leo	f
X	$c \wedge (h \vee f)$



3.6.3 Extensions Relationnelles de l'ACF

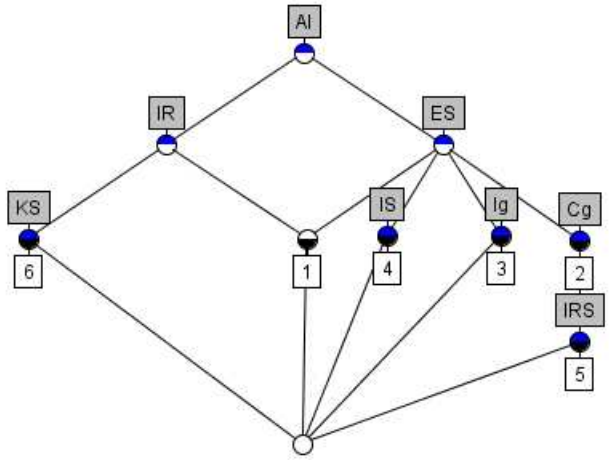
Relations entre attributs

La motivation principale de l'extension de l'ACF pour la prise en compte des relations entre attributs dans un contexte formel est de pouvoir intégrer des relations sémantiques explicites dans les structures conceptuelles de l'ACF. Ces relations sémantiques se présentent souvent sous la forme de taxonomies ou de thésaurus. Deux principales approches ont été proposées dans ce cadre. La première approche, définie essentiellement par Carpineto et Romano [Carpineto and Romano, 2004a], consiste à enrichir un contexte formel à partir d'une hiérarchie explicitant des relations entre certains attributs du contexte formel. L'enrichissement

du contexte à partir d'une taxonomie s'effectue de la manière suivante. Dans le cas ou deux attributs du contexte sont reliés par un chemin de généralisation/spécialisation dans la taxonomie, les objets qui possèdent l'attribut le plus spécifique sont reliés à l'attribut le plus général (un "×" est ajouté au contexte dans les cases intersections des lignes de ces objets avec la colonne de l'attribut le plus général). L'illustration du processus d'enrichissement du contexte à partir d'une taxonomie est donnée par l'exemple du contexte dans la table 3.13 (gauche) et la taxonomie représentée par la hiérarchie d'attributs donnée dans la figure 3.7. Le treillis original correspondant au contexte non enrichi est donné dans la table 3.13 (droite). Le contexte enrichi et le treillis correspondant sont donnés dans la figure 3.14. Une comparaison rapide de ces deux treillis montre que dans le treillis enrichi, la hiérarchie des attributs est conforme à la hiérarchie de la taxonomie, autrement dit, les attributs les plus généraux dans la taxonomie sont en haut du treillis alors que les attributs les plus spécifiques sont en bas du treillis.

TABLE 3.13 – Le contexte formel et du treillis correspondant donné dans [Carpineto and Romano, 2004a].

	Artificial Intelligence (AI)	Expert Systems (ES)	Information Retrieval (IR)	Cataloguing (Cg)	Indexing (Ig)	Information Science (IS)	Information Retrieval Systems (IRS)	Knowledge-based Systems (KS)
1	x	x	x					
2	x	x		x				
3	x	x			x			
4	x	x				x		
5	x	x		x			x	
6	x		x					x



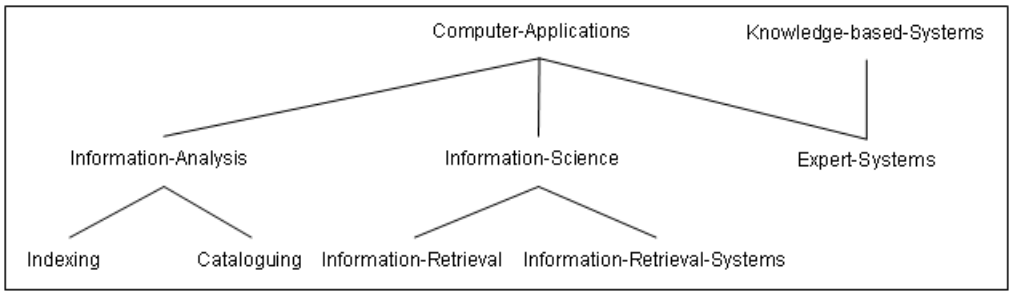
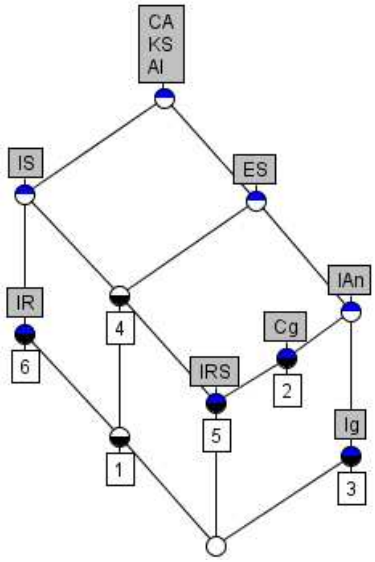


FIGURE 3.7 – Taxonomie d'enrichissement - Hiérarchie d'attributs

TABLE 3.14 – Contexte formel enrichi par la taxonomie de la figure 3.7 et le treillis correspondant

	Artificial Intelligence (AI)	Expert Systems (ES)	Information Retrieval (IR)	Cataloguing (Cg)	Indexing (Ig)	Information Science (IS)	Information Retrieval Systems (IRS)	Knowledge-based Systems (KS)	Computer-Applications (CA)	Information-Analysis (IAn)
1	x	x	x			x		x	x	
2	x	x		x				x	x	x
3	x	x			x			x	x	x
4	x	x				x		x	x	
5	x	x		x		x	x	x	x	x
6	x		x			x		x	x	



L'enrichissement du contexte peut aussi être fait entre attributs du contexte et termes d'une taxonomie (ne figurant pas dans le contexte initial). Dans ce cas de nouveaux attributs sont d'abord ajoutés au contexte ensuite la relation avec les objets est établie de la même façon que précédemment. Dans ce cas les attributs ajoutés jouent un rôle de restructuration du treillis conformément à la hiérarchie des attributs dans la taxonomie.

La deuxième approche, définie par U. Priss [Priss, 1997], consiste à combiner un treillis de concepts et un thésaurus, représenté aussi par un treillis de concepts, dans un même treillis. Priss a défini plusieurs façons de combiner les deux treillis dont celle proposée par Carpineto et Romano et détaillée précédemment. Cette approche a été étendue par la suite pour considérer plusieurs facettes de thésaurus, représentées sous la forme de treillis de concepts, et les combiner dans un même système de recherche documentaire (le système FaIR) [Priss, 2000b].

Relations entre objets

L'étude des relations entre objets dans un contexte formel en ACF a fait l'objet de plusieurs travaux [Priss, 1996, Prediger and Wille, 1999, Rouane-Hacene, 2006]. Dans ses travaux de thèse, U. Priss a proposé une approche, appelée **Analyse Relationnelle de Concepts** (ARC) [Priss, 1996], pour prendre en compte des relations sémantiques dans les bases de données lexicales. Ces bases de données sont alors représentées sous la forme de contextes formels, appelés **contextes formels linguistiques** représentant des mots (les objets) en fonction de leurs caractéristiques à partir de la terminologie linguistique (les attributs) auquel cas le contexte formel linguistique est dit dénotatif, ou des sens (les objets) et leurs caractéristiques (les attributs) auquel cas le contexte est dit connotatif.

L'étude de ces contextes s'effectue en deux étapes. La première étape consiste à construire le treillis de concepts, appelé **treillis linguistique**, à partir d'un contexte linguistique en utilisant

l'ACF classique. La deuxième étape consiste à établir des liens entre les concepts du treillis construit en fonction des relations sémantiques et lexicales, telles que la synonymie, entre les mots en extensions de ces concepts. L'établissement d'un lien entre une paire de concepts dans le treillis, dépend de la nature de la relation entre les mots des deux extensions. Ces relations sont gérées par des quantificateurs qui permettent d'exprimer que tous les mots sont concernés par une relation ou que seule une partie des mots est concernée. À la fin de la deuxième étape, des termes et/ou expressions lexicaux peuvent être dérivés des treillis linguistiques.

Plus tard, S. Prediger a étendu l'échelonnage conceptuel en définissant l'**échelonnage relationnel** pour les contextes multivalués [Prediger and Wille, 1999]. L'échelonnage relationnel est défini comme un échelonnage conceptuel qui met en jeu plusieurs attributs et objets à la fois. L'approche est définie dans le cadre général de la *logique contextuelle* qui combine l'ACF et les graphes conceptuels (GC) [Sowa, 1992]. L'échelonnage relationnel est défini dans le but de pouvoir dériver, à partir d'un contexte multivalué, des structures de jugement formel en plus des concepts formels. Ces structures de jugement se présentent sous la forme de graphes conceptuels.

Concrètement, l'approche consiste à définir, à partir d'un contexte multivalué, un ensemble d'échelles conceptuelles et relationnelles. Ces échelles forment une **famille de contextes relationnels**. Ensuite, un treillis de concepts est construit à partir de chaque échelle conceptuelle et un graphe de concepts est construit à partir de l'échelle relationnelle. À la fin, un treillis global appelé **treillis des graphes de concepts** est construit à partir de l'ensemble des treillis d'échelles et du graphe de concepts. Cette approche a été appliquée pour l'organisation et l'interprétation de l'activité aérienne civile entre les aéroports autrichiens.

Récemment, M.A. Rouane-Hacene [Rouane-Hacene, 2006] a redéfini, dans le cadre de ses travaux de thèse, les fondements de l'ARC ainsi que la façon dont la structure conceptuelle est extraite à partir d'une famille de contextes. Contrairement à U. Priss qui exploite les liens entre les objets après construction du treillis pour établir de nouveaux liens entre les concepts, M.A. Rouane-Hacene exploite de telles relations lors de l'étape de formation des concepts du treillis final.

L'approche consiste à partir d'un contexte multivalué dont certains attributs multivalués, dits relations, expriment des relations entre les objets du contexte. Ce contexte est transformé de la manière suivante. Les attributs multivalués sont transformés en ensembles d'attributs monovalués par l'intermédiaire de l'échelonnage conceptuel pour former un premier contexte monovalué. Les relations sont représentées séparément dans un deuxième contexte. On obtient donc une famille de contextes monovalués appelée **Famille de Contextes Relationnels**. Cette famille constitue le point de départ du processus de formation des structures conceptuelles correspondantes appelées famille de treillis relationnels. La table 3.15 donne un exemple de contextes monovalués, la relation entre les objets de ces contextes est donnée dans la table 3.16.

Les treillis correspondants aux contextes monovalués de la table 3.15 sont données dans la figure 3.8.

La première étape de ce processus consiste à construire le treillis de concepts correspondant au contexte monovalué représentant la relation entre les objets et les attributs. Les étapes suivantes sont des itérations des deux opérations suivantes : (1) enrichir le contexte à partir du treillis obtenu et (2) construire le nouveau treillis du contexte enrichi. Le processus s'arrête lorsqu'aucune nouvelle relation ne peut être déduite à partir du treillis obtenu à l'étape précédente. L'étape d'enrichissement du contexte est appelée **échelonnage relationnel**. Elle consiste à ajouter les concepts du treillis comme nouveaux attributs au contexte monovalué. L'ajout d'un concept au contexte est effectué lorsqu'un objet du contexte est en relation avec les objets en extension du concept considéré. On distingue deux types d'échelonnage relationnel : existentiel et universel. L'échelonnage relationnel existentiel consiste à considerer qu'un objet du contexte est en relation

TABLE 3.15 – Contextes Monovalués

	Jeune	Adulte	Salaire< 2000	Salaire>= 2000
Pierre	×		×	
Marie	×			×
Jean		×		×
Annie		×	×	

	Montagnes	Mer	Cher	Bas prix
Rocheuses Américaines	×		×	
Voges	×			×
Tunisie		×		×
Bahamas		×	×	

TABLE 3.16 – Relation : A_Visité

	Rocheuses Américaines	Voges	Tunisie	Bahamas
Pierre			×	
Marie	×			
Jean		×		
Annie				×

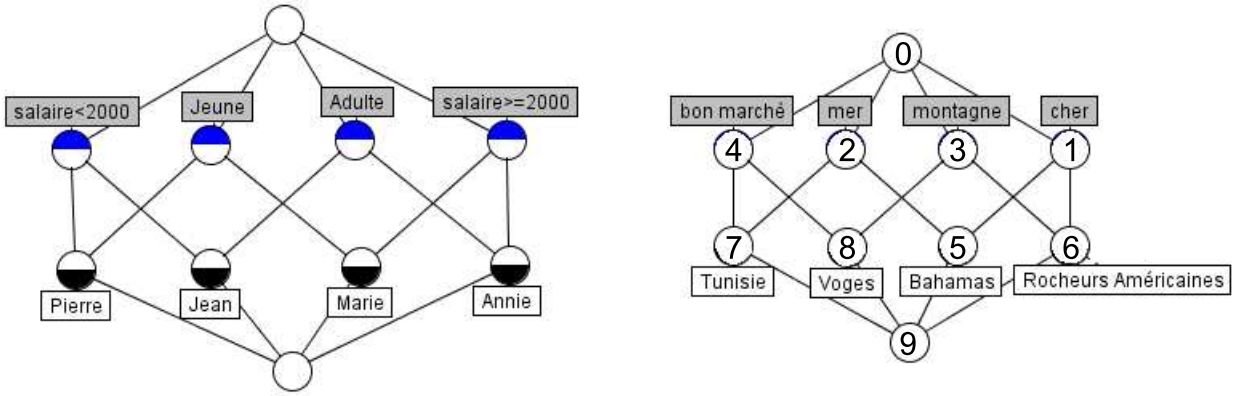


FIGURE 3.8 – Treillis monovalués : Touristes (à gauches), Destinations (à droite).

avec un concept du treillis lorsqu'il est en relation avec au moins un objets dans son extension. L'échelonnage relationnel universel consiste à considerer qu'un objet du contexte est en relation avec un concept du treillis lorsque tous les objets avec lesquels cet objet est en relation sont dans l'extension du concept.

Le treillis résultant après l'exécution des étapes de ce processus sur l'exemple de la famille de contextes donné dans la table 3.15 et la table 3.16 est donnée dans la figure 3.9. L'interprétation de certains concepts de ce treillis permet de déduire les relations entre les classes de tourists et les classes de destinations. Par exemple, le concept ({Marie, Pièrre}, {A_Visité : c2, Jeune}) permet de déduire que les jeunes touristes choisissent les destinations où il y a la mer (cf. concept numéro 2 dans le treillis des destinations donné dans la figure 3.8). De la même manière, le concept ({Annie, Jean}, {A_Visité : c3, Adulte}) permet de déduire que les touristes adultes choisissent les destinations montagneuses.

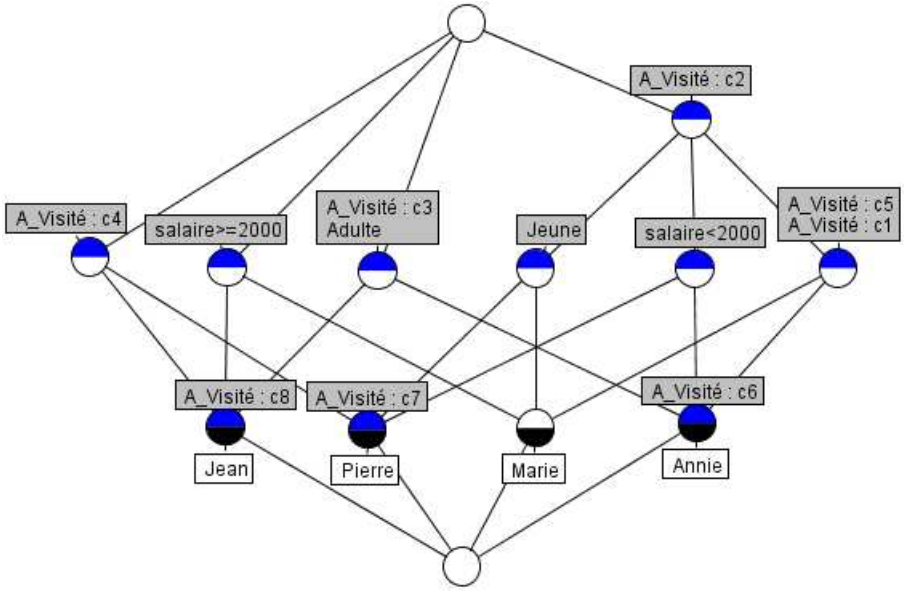


FIGURE 3.9 – Treillis résultant après enrichissement relationnel.

3.6.4 Extension aux objets symboliques

L'extension de l'ACF aux objects symboliques [Bock and Diday, 2000] a été proposée par G. Polaillon dans le cadre de ses travaux de thèse [Polaillon, 1998]. Trois types de données ont été considérés par cette extension : les données multivaluées qui se présentent sous la forme d'un contexte multivalué où la relation objet×attribut peut prendre un ensemble de valeurs au lieu d'une seule, les données intervalles qui se présentent sous la forme de contexte multivalué où la relation objet×attribut prend un intervalle de valeurs et les données histogrammes qui se présentent sous la forme de contexte multivalué où la relation objet×attribut prend pour valeurs un histogramme. Un exemple de données multivaluées est donné dans la table 3.17. Pour traiter ce type de contextes, G. Polaillon défini de nouvelles connexions de Galois qui permettent de dériver à partir de chaque contexte deux types de treillis : un **treillis d'union** et un **treillis d'intersection**. Les treillis d'union et d'intersection correspondant au contexte donné dans la table 3.17 sont donnés dans la figure 3.10 à gauche et à droite respectivement.

TABLE 3.17 – Exemple de tableau de données multivaluées

	$y1$	$y2$
$w1$	vert	grand
$w2$	bleu	petit
$w3$	rouge	petit

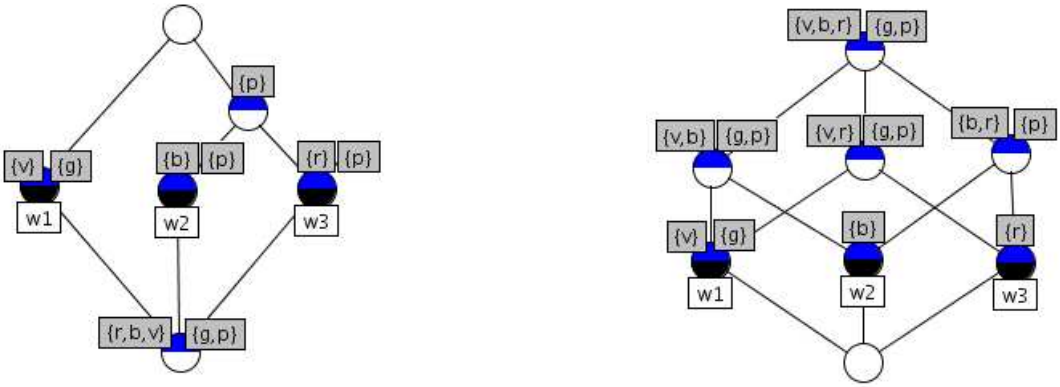


FIGURE 3.10 – Treillis de données multivaluées : Intersection (à gauche), Union (à droite).

3.6.5 Autres extensions

Il existe d'autre extensions qui ont été proposées. Ganter et Kuznetsov ont proposé une extension de l'ACF pour l'extraction de patrons dans des graphes [Ganter and Kuznetsov, 2001]. Cette approche est considérée comme un moyen de prendre en compte des relations sémantiques entre ensembles d'attributs (à la différence de l'approche de Carpineto et Romano et de celle de U. Priss qui considèrent les relations entre attributs) [Carpineto and Romano, 2004a]. Brito et al. ont défini une approche pour l'analyse des données probabilistes [Brito and Polaillon, 2005] similaire à l'approche ACFF mono-facette définie par Ben Yahia et Jaoua. Elloumi et Jaoua ont défini une approche pour la prise en compte des données imprécises dans des contextes multivalués [Jaoua and Elloumi, 2002]. Cette approche est similaire à celle proposée par G. Polaillon pour les données intervalles.

Chapitre 4

Découverte de ressources par treillis de concepts dans BioRegistry

Sommaire

4.1 Introduction 55
4.2 Représentation du contenu de BioRegistry en ACF 56
4.2.1 Classification flexible du contenu de BioRegistry 58
4.3 Exploitation du contenu de BioRegistry 58
4.3.1 Navigation 58
4.3.2 Interrogation 59
4.4 Formalisation : l'algorithme BR-Explorer 61
4.4.1 Définitions 61
4.4.2 L'algorithme BR-Explorer 63
4.4.3 Déroulement de BR-Explorer sur un exemple 63
4.4.4 Expressivité des requêtes dans BR-Explorer 66
4.4.5 Correction et complétude de BR-Explorer 67
4.4.6 Complexité de BR-Explorer 68
4.5 Raffinement de requête à partir de ressources sémantiques 68
4.5.1 Raffinement de requête par généralisation 70
4.5.2 Raffinement de requête par spécialisation 71
4.5.3 Raffinement mixte 72
4.5.4 Choix du type de raffinement 72
4.6 Dépendances entre attributs : hiérarchie d'attributs dans BR-Explorer 73
4.6.1 Le besoin d'exprimer les dépendances entre attributs 73
4.6.2 Formalisation 74
4.6.3 Application des hiérarchies d'attributs à un treillis de concepts 75
4.6.4 Hiérarchies d'attributs pour la découverte de ressources guidée par des connaissances 77
4.6.5 Connaissances globales ou connaissances locales 82

4.1 Introduction

Dans un scénario d'exploitation de BioRegistry par l'intermédiaire d'un formulaire (voir section 2.3.3), le biologiste doit entrer les valeurs de quelques champs de métadonnées et reçoit

en réponse une liste triée de sources de données de BioRegistry répondant à sa requête. Cette approche oblige l'utilisateur à construire une requête, ce qui peut se révéler inefficace sans une vision globale des sources de données contenues dans BioRegistry. Une fonctionnalité de navigation à travers des groupes de sources de données partageant un nombre variable de métadonnées communes serait en effet une aide précieuse pour l'utilisateur dépassé par le nombre et l'hétérogénéité de ces sources. Connaissant alors l'existence et les propriétés de ces sources de données, il lui sera plus facile de formuler une requête précise pour laquelle il attendra une réponse exhaustive. Pour faire face à ces exigences, nous avons résolu d'appliquer l'Analyse de Concepts Formels au contenu de BioRegistry afin de rendre possibles (i) une classification flexible des sources de données sur la base du partage de propriétés et (ii) une interrogation exhaustive de BioRegistry.

Le reste de ce chapitre est organisé comme suit. Les sections 4.2 et 4.3 présentent de manière détaillée l'approche de découverte de ressources par treillis de concepts dans BioRegistry. La section 4.4 contient la formalisation cette approche. Les sections 4.5 et 4.6 présentent différentes façons de prendre en compte des connaissances et définissent l'approche de découverte de ressources guidée par des connaissances dans BioRegistry.

4.2 Représentation du contenu de BioRegistry en ACF

La présentation du contenu de BioRegistry en terme d'ACF consiste à traduire le schéma d'organisation des sources de données et de leurs méta-données en terme d'une relation objets× attributs. Une première façon de procéder est de considérer les sources de données comme des objets et les catégories des méta-données comme attributs. Dans ce cas le contexte obtenu est un contexte multivalué. Les valeurs de la relation objets×attributs sont les méta-données utilisées dans chaque catégorie pour annoter les sources de données. La table 4.1 donne un exemple de contexte multivalué correspondant à une partie des données de BioRegistry. Dans cet exemple,

Table 4.1 – Un contexte multivalué correspondant à une partie des données de BioRegistry

	Organism	Content Quality	Ontology
BD_1	Amphibians, Fishes	Complete	NCBI
BD_2	Amphibians, Fishes	Complete, Updated	NCBI
BD_3	Amphibians, Mammals		NCBI
BD_4	Birds, Mammals	Updated	
BD_5	Amphibians, Mammals	Complete, Updated	
BD_6	Birds, Mammals	Complete	GO
BD_7	Birds, Mammals	Complete, Updated	GO, NCBI
BD_8	Birds, Mammals		NCBI

nous avons considéré trois catégories de méta-données : la catégorie *Organism* qui regroupe les organismes (ou espèces) concernés par les données de sources de données (les amphibiens, les mammifères, les oiseaux et les poissons), la catégorie *Content Quality* qui donne un idée sur la qualité des contenus des sources de données (à jour, complet) et la catégorie *Ontology* qui spécifie l(es)'ontologie(s) utilisée(s) comme référence(s) lors de l'annotation des sources de données (GO, NCBI). La relation objets×attributs matérialisée par ce contexte exprime le fait qu'une source de données est annotée par une ou plusieurs métadonnées appartenant à l'une des catégories mentionnées. Par exemple, la source de données BD_1 est annotée par les méta-données

Amphibians et Fishes de la catégorie *Organism*, *Complete* de la catégorie *Content Quality* et *NCBI* de la catégorie *Ontology*. Cette relation est interprétée comme suit : BD_1 contient des données complètes relatives aux organismes amphibiens et utilise des termes provenant de la taxonomy NCBI.

Pour pouvoir appliquer les résultats de l'ACF sur ce contexte, nous avons d'abord procédé à sa transformation en contexte monovalué en utilisant un échantillonnage conceptuel adéquat (section 3.5.2). Le contexte monovalué obtenu est donné dans la table 4.2. Le treillis de concepts

TABLE 4.2 – Contexte formel représentant une partie du corpus BioRegistry.

	Organism				Content Quality		Ontology	
	Amphibiens	Birds	Fishes	Mammals	Complete	Updated	GO	NCBI
BD_1	×		×		×			×
BD_2	×		×		×	×		×
BD_3	×			×				×
BD_4		×		×		×		
BD_5	×			×	×	×		
BD_6		×		×	×		×	
BD_7		×		×	×	×	×	×
BD_8		×		×				×

correspondant à ce contexte est donné à la figure 4.1.

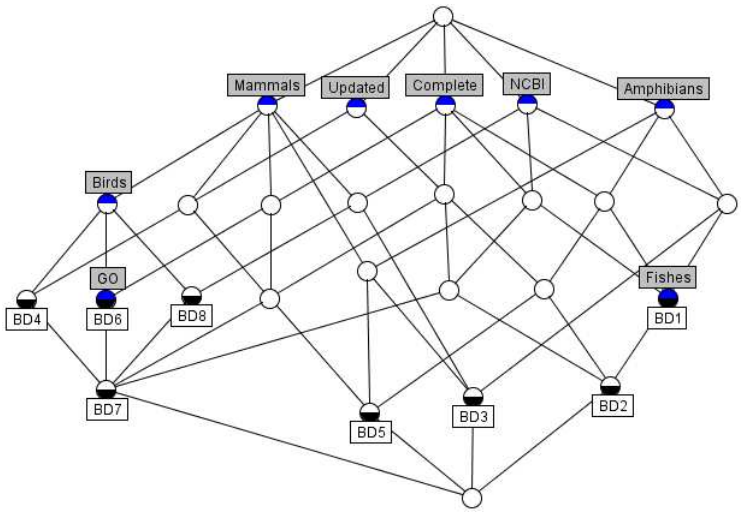


FIGURE 4.1 – Le treillis de concepts correspondant au contexte formel donné dans la table 4.2

4.2.1 Classification flexible du contenu de BioRegistry

La formalisation du contenu de BioRegistry sous la forme d'un treillis de concepts permet soit de naviguer dans le treillis soit d'interroger les données du treillis et extraire des informations pertinentes pour un objectif donné. Par exemple un utilisateur intéressé par les méta-données de type Subjects associées à l'ensemble des sources de données peut définir un contexte formel où les attributs ne sont constitués que des méta-données de la catégorie Subjects. L'ensemble d'objets correspondant comprend toutes les sources de données pour lesquelles au moins une méta-donnée de cette catégorie a été renseignée. Un treillis peut alors être construit, permettant de visualiser le partage des méta-données de la catégorie Subjects dans BioRegistry. Dans un autre cas, l'utilisateur peut être intéressé par la classification des sources de données relatives à l'organisme humain. Un nouveau contexte formel est construit automatiquement, dans lequel l'ensemble des objets comprend toutes les sources de données de BioRegistry dont la méta-donnée Organism a la valeur Human. L'ensemble des attributs est constitué de toutes les méta-données associées à cet ensemble de sources. Un nouveau treillis peut alors être construit pour répondre à ce nouveau besoin.

4.3 Exploitation du contenu de BioRegistry

4.3.1 Navigation

Une fois le treillis de concepts correspondant aux données de BioRegistry construit, il est possible d'effectuer une exploitation par navigation des concepts de ce treillis de la manière présentée en section 3.4.2. Pour ce faire, on doit disposer d'une visualisation du diagramme de Hasse afin de faciliter l'interaction avec le treillis et l'identification des concepts intéressants. À ce niveau, les modules supplémentaires de visualisation de graphes peuvent être d'une grande utilité.

La visualisation du voisinage du concept courant à une étape donnée facilite l'identification du concept à atteindre à l'étape qui suit. Cette possibilité montre aussi la répartition des concepts autour du concept courant et donne ainsi une idée sur la distribution des données. Elle correspond à la méthode de visualisation appelée *fish eye* (œil de poisson). Appliquée aux concepts du treillis, cette méthode consiste à placer le concept courant au centre et distribuer les concepts de son voisinage sur des anneaux concentriques autour de ce concept. Les concepts à un pas sont placés sur l'anneau le plus interne et ceux à deux pas sur le suivant et ainsi de suite.

La possibilité de faire des zooms sur une partie du treillis permet de réduire l'espace de recherche aux concepts intéressants. De plus, un affichage incrémental des concepts au fur et à mesure qu'on avance dans l'exploration du treillis permet une meilleure analyse des liens de spécialisation et/ou de généralisation qu'un concept peut avoir avec son voisinage. Ces deux opérations (zoom et affichage incrémental) facilite l'exploration du treillis de concepts à la recherche de sources de données pertinentes. Connaissant globalement l'objectif recherché, on peut entamer l'exploration à partir du concept le plus général du treillis. Ce concept correspond à la classe de toutes les sources de données considérées dans le contexte. À partir de ce concept on peut analyser les liens de spécialisation et choisir le concept à atteindre. Le déplacement au nouveau concept entraîne l'affichage des nouveaux concepts plus spécifiques et la disparition des concepts du niveau précédent. Ce traitement peut être répété jusqu'à l'identification du concept correspondant au mieux aux objectifs de l'exploration du treillis.

4.3.2 Interrogation

Le principe général de l'interrogation du treillis de concepts représentant les données de BioRegistry est similaire à celui des méthodes introduites en section 3.4.2 dans la mesure où une requête doit être positionnée dans le treillis puis le voisinage de la requête est examiné pour construire la réponse. Cependant, la différence réside dans la façon dont ce voisinage est examiné lors de la construction de la réponse à une requête donnée.

Stratégie de recherche dans BR-Explorer

Une fois que le treillis de concepts est construit à partir des données de BioRegistry, la stratégie de recherche de sources de données consiste à appliquer la suite des étapes suivantes :

1. Définition d'une requête : il s'agit de donner une description des sources de données qui pourraient répondre aux questions qu'on se pose à travers la requête. Autrement dit, il s'agit de donner les attributs qui reflètent les caractéristiques des sources de données à identifier. Une requête se présente donc sous la forme d'un ensemble de méta-données.
2. Insertion de la requête dans le treillis de concepts : cette étape est facilitée par l'existence d'algorithmes performants pour la construction incrémentale des treillis de concepts (Section 3.3.6). La construction incrémentale du treillis de concept se fait soit par ajout d'objet auquel cas on doit disposer de l'ensemble d'attributs avec lesquels l'objet est en relation ou par ajout d'attribut auquel cas on doit disposer de l'ensemble d'objets qui le possèdent. Dans notre cas, on dispose initialement du treillis de concepts construit à partir des données de BioRegistry et d'une requête qui consiste en un ensemble de méta-données (les attributs). On est donc dans le cas où on dispose d'un ensemble d'attributs décrivant un (ou plusieurs) objet(s) qui existent mais qu'on cherche à déterminer. Il est plus facile de se ramener au cas de la construction incrémentale par ajout d'objet. En effet il suffit de considérer un objet fictif possédant toutes les méta-données dans la requête. La requête est donc présentée sous la forme d'un couple (objet, attributs) qui peut être inséré dans le treillis de concepts.
3. Localisation de la requête dans le treillis de concepts obtenu : cette étape consiste à localiser, dans le treillis de concepts modifié (suite à l'insertion de la requête), le concept le plus général incorporant toutes les méta-données spécifiées dans la requête. La recherche d'un tel concept est facilitée par l'élagage de parties de treillis en considérant la propriété d'inclusion entre les intensions des concepts formels matérialisée dans la relation sous-concept/super-concept. En effet, étant donné un concept, si son intension contient les attributs de la requête, on le considère et on teste ses super-concepts sinon on ignore toute la partie du treillis au dessus du concept considéré. Le point de départ de la recherche est le concept le plus spécifique dans le treillis (bottom). Le concept représentant la requête dans le treillis est le concept qui vérifie les deux conditions suivantes : (1) il contient les attributs de la requête dans son intension et (2) il n'a pas de superconcept qui vérifie la condition (1).
4. Recherche des sources de données pertinentes dans le voisinage du concept requête dans le treillis. Cette recherche doit être effectuée conformément à une définition de pertinence pour ne considérer que les concepts susceptibles de contenir dans leurs extensions des sources de données pertinentes pour la requête considérée.
5. Présentation de la réponse sous la forme d'un ensemble de sources de données ordonnées selon leur pertinence pour la requête. Le classement des sources de données dans le résultat est facilité par un premier classement déduit directement à partir de la classification des concepts dans le treillis.

La stratégie de recherche de source de données pertinentes dont les étapes sont listées précédemment repose sur deux facteurs : le premier est le critère de pertinence qui permet de déterminer l'ensemble de concepts du treillis qui sera considéré lors de la recherche et le second est l'organisation des concepts dans le treillis qui détermine la manière dont les concepts pertinents seront identifiés.

Critère de pertinence

À la différence des systèmes d'indexation automatique utilisés par les moteurs de recherche d'information sur le Web, l'indexation des sources de données biologiques dans BioRegistry consiste à choisir soigneusement les métadonnées d'annotation pour chaque source (voir section 2.3.2). De cette manière, l'ensemble de métadonnées associé à chaque source reflète de manière précise et sûre les caractéristiques (qualités) et le contenue de la source. De ce fait, la définition d'une requête en vue de l'identification des sources de données pertinentes consiste à choisir parmi l'ensemble de métadonnées utilisées dans BioRegistry celles qui décrivent les sources à identifier. Étant donné qu'une requête décrit la source de données idéale recherchée, la source la plus pertinente pour une requête donnée est celle qui est annotée par toutes les métadonnées utilisées dans cette requête. Une source de données annotée par une partie de la requête seulement est certes moins pertinente mais peut être considérée comme réponse secondaire à la requête. En conclusion, la pertinence d'une source de données vis-à-vis d'une requête données est fonction du nombre de métadonnées dans la requête qui annotent la source de données. Cette définition de la pertinence est à la base du processus de recherche détaillé dans la suite et illustré par un exemple. Elle est différente de la notion de voisinage utilisée dans [Carpineto and Romano, 2000], qui peut aboutir à l'obtention de documents ne partageant aucun terme avec la requête, ce qui ne correspond pas à nos besoins.

Parcours du treillis : localisation des sources de données pertinentes

Dans les approches de recherche d'information par treillis détaillées en section 3.4.2, l'interrogation du treillis consiste à placer un concept requête dans le treillis et à récupérer ensuite les sources de données dans son extension (Godin [Godin et al., 1995b]) et celles dans les extensions des concepts formant les anneaux concentriques centrées autour du concept requête (Carpineto et Romano [Carpineto and Romano, 2000]). Par rapport au critère de pertinence introduit dans la section précédente : le partage de métadonnées avec la requête, les sources de données pertinentes pour une requête sont celles qui apparaissent dans l'extension du concept requête (ces sources partagent toutes les métadonnées dans la requête) et celles dans les extensions des super-concepts de la requête (par la propriété d'inclusion entre intensions de concepts dans le treillis, ces sources partagent seulement une partie des métadonnées de la requête). La recherche des sources de données pertinentes pour une requête donnée se ramène donc à localiser d'abord le concept requête dans le treillis et de parcourir ensuite ses super-concepts jusqu'au concept le plus général du treillis. Ce parcours de treillis fournit en plus de l'ensemble de sources de données pertinentes, un premier classement des sources selon le nombre de métadonnées qu'elles partagent avec la requête.

Cette manière de définir la pertinence et la stratégie de recherche dans le treillis (considérer le concept requête et ses super-concepts) donnent une réponse plus complète que celle donnée par l'approche de Godin et moins bruitée que celle donnée par l'approche de Carpineto et Romano.

La formalisation de la stratégie de recherche nous a amené à définir l'algorithme BR-Explorer (BioRegistry Explorer) pour la découverte de ressources pertinentes dans BioRegistry. La forma-

lisation détaillée de BR-Explorer est donnée dans la section suivante.

4.4 Formalisation : l'algorithme BR-Explorer

4.4.1 Définitions

Définition 23 (Requête) *Une requête Q est un couple (X,Y) où Y est un ensemble d'attributs décrivant les objets à chercher et X est l'ensemble formé par un objet virtuel x ($X = \{x\}$) supposé satisfaire les attributs de la requête : $X' = Y$.*

Dans la définition 23, la requête se présente sous la forme d'un couple comme motivé en section 4.3.2. Cette forme facilite l'insertion de la requête dans le treillis de concepts en utilisant un algorithme de construction incrémentale de treillis de concepts. Une telle insertion peut être considérée comme l'ajout d'une nouvelle entrée (un nouvel objet et ses attributs) dans le contexte formel considéré.

Définition 24 ($\oplus$) *Pour un contexte formel $\mathbb{K} = (G, M, I)$ et une requête $Q = (X, Y)$, nous définissons l'opérateur d'addition $\oplus$ comme suit :*

$$\begin{aligned} \mathbb{K}_Q &= \mathbb{K} \oplus Q \\ &= (G, M, I) \oplus (X, Y) \\ &= (G \cup X, M \cup Y, I_Q) \\ &= (G_Q, M_Q, I_Q) \end{aligned}$$

Dans la définition de précédente, l'utilisation de $M_Q = M \cup Y$ est introduite pour couvrir le cas où il est possible de définir des requêtes avec des métadonnées qui ne sont pas forcément dans M. Bien que ces métadonnées ne décrivent aucune source dans le contexte, leur utilisation peut être intéressante lors du recours au raffinement de requête développé dans les sections suivantes. La relation I_Q désigne la relation I à laquelle s'ajoute la relation entre X et Y.

L'insertion de la requête $Q = (X, Y)$ dans le treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$ produit un nouveau treillis $\underline{\mathfrak{B}}(\mathbb{K}_Q)$. On appellera *pivot* le concept représentant la requête dans $\underline{\mathfrak{B}}(\mathbb{K}_Q)$ et on le définit comme suit.

Définition 25 (Pivot) *Considérons $\mathbb{K} = (G, M, I)$ un contexte formel et $Q = (X, Y)$ une requête. Le concept pivot dans le treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K}_Q)$ est le concept formel $P = (Y', Y)$.*

Considérons l'ensemble de concepts formels $\mathfrak{B}(\mathbb{K})$ du contexte formel $\mathbb{K} = (G, M, I)$ et considérons la requête $Q = (X, Y)$. Selon la relation entre l'ensemble X' et les intensions des concepts formels dans $\mathfrak{B}(\mathbb{K})$, nous distinguons les cas suivants pour le concept pivot $P = (Y', Y)$ dans le treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K}_Q)$:

- S'il n'existe pas de concept $C = (A, B) \in \mathfrak{B}(\mathbb{K})$ tel que $Y \subseteq B$ alors X et Y sont fermés dans G_Q et M_Q respectivement et $Y' = X$. Cela signifie que l'ajout de la nouvelle entrée dans le contexte formel $\mathbb{K} \oplus Q$ produit au moins un nouveau concept formel $(X, Y) = (Y', Y)$. De plus, chaque concept formel $C_1 = (A_1, B_1)$ vérifiant $B_1 \subseteq Y$ est transformé en $C_2 = (A_1 \cup X, B_1)$ dans $\underline{\mathfrak{B}}(\mathbb{K}_Q)$.
- S'il existe un concept $C = (A, B) \in \mathfrak{B}(\mathbb{K})$ tel que $Y \subseteq B$ alors nous distinguons deux cas :
  - Si $Y \subset B$ alors $Y' = A \cup X$ et le concept pivot est $P = (A \cup X, Y)$. Cela signifie que l'objet virtuel x (la nouvelle entrée dans $\mathbb{K}_Q$) est fusionné avec d'autres objets possédant les mêmes attributs. Comme dans le cas précédent, l'opération $\mathbb{K} \oplus Q$ entraîne

la création d'au moins un nouveau concept formel dans $\underline{\mathfrak{B}}(\mathbb{K}_Q)$ à savoir le concept pivot $P = (A \cup X, Y)$. De plus, chaque concept formel $C_1 = (A_1, B_1)$ vérifiant $B_1 \subseteq Y$ est transformé en $C_2 = (A_1 \cup X, B_1)$ dans $\underline{\mathfrak{B}}(\mathbb{K}_Q)$.

– Si $Y = B$ alors $Y' = A \cup X$ et le concept pivot est $P = (A \cup X, Y) = (A \cup X, B)$. Toutefois, contrairement aux deux cas précédents, l'opération $\mathbb{K} \oplus Q$ ne produit aucun nouveau concept formel dans $\underline{\mathfrak{B}}(\mathbb{K}_Q)$. Elle transforme simplement les concepts formels de la forme $C_1 = (A_1, B_1)$ tels que $B_1 \subseteq Y$ en $C_2 = (A_1 \cup X, B_1)$.

Définition 26 (couverture supérieure) *(1) Considérons un contexte formel $\mathbb{K} = (G, M, I)$, l'ensemble des ses concepts formels $\mathfrak{B}(\mathbb{K})$ et le treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$. La couverture supérieure (abrégé en c_sup) d'un concept formel $D \in \mathfrak{B}(\mathbb{K})$ est constituée des super-concepts directs de D dans $\underline{\mathfrak{B}}(\mathbb{K})$:*

$$c_sup(D) = \{C \in \mathfrak{B}(\mathbb{K}) \mid D \leq C \text{ et } \nexists\, Z \in \mathfrak{B}(\mathbb{K}) \mid D \leq Z \leq C\}$$

(2) Étant donné un ensemble $\{C_j\}_{j \in J}$ (J est un ensemble dont le cardinal est un entier positif inférieur au nombre de concepts formels dans $\mathfrak{B}(\mathbb{K})$) de concepts formels dans $\mathfrak{B}(\mathbb{K})$, la couverture supérieure de l'ensemble $\{C_j\}_{j \in J}$ est définie par l'union des couvertures supérieures des concepts formels :

$$c_sup(\{C_j\}_{j \in J}) = \bigcup_{j \in J} c_sup(C_j)$$

Le critère de pertinence introduit de manière intuitive en section 4.3.2 est définit formellement comme suit.

Définition 27 (pertinence) *(1) Considérons une entrée (un couple formé par un objet et l'ensemble de ses attributs) ($\{g\}$,$\{g\}'$) dans un contexte formel $\mathbb{K} = (G, M, I)$ et une requête $Q = (X, Y)$. L'objet g est pertinent pour la requête Q si et seulement si $g \neq \varnothing$, $\{g\}' \neq \varnothing$ et $\{g\}' \cap Y \neq \varnothing$, i.e. il existe au moins un attribut dans Y qui est possédé par l'objet g.*

(2) Le degré de pertinence de l'objet g par rapport à la requête Q est donné par le nombre d'attributs communs entre $\{g\}'$ et Y i.e. $|\{g\}' \cap Y|$.

Proposition 1 *Considérons un contexte formel $\mathbb{K} = (G, M, I)$ et une requête $Q = (X, Y)$. Tous les objets dans G pertinents pour Q sont dans l'extension du concept pivot $P = (Y', Y)$ et dans les extensions de ses super-concepts dans $\underline{\mathfrak{B}}(\mathbb{K}_Q)$.*

Preuve 1 *Considérons d'abord les objets dans Y', l'extension du concept pivot. Selon la définition de $P = (Y', Y)$ (définition 25) et la définition de la pertinence (définition 27), tous les objets dans Y' sont pertinents pour $Q = (X, Y)$ puisqu'ils partagent tous les attributs de la requête (l'ensemble Y).*

Considérons maintenant le cas des super-concepts du concept pivot P. Soit $C = (A, B)$ un super-concept de P dans $\underline{\mathfrak{B}}(\mathbb{K}_Q)$, i.e. $P = (Y', Y) \leq C = (A, B)$. Par définition de la relation de d'ordre "$\leq$" entre les concepts dans le treillis de concepts, on a $B \subseteq Y$. Ce qui signifie que tout objet dans A possède au moins un attribut dans Y, d'où la pertinence pour Q.

Le seul concept qui constitue un cas particulier est top ($\top$), le concept le plus général du treillis, lorsque son intension est vide. Dans ce cas la définition de pertinence n'est pas vérifiée et les objets dans l'extension de $\top$ ne font pas partie de l'ensemble des objets pertinents pour Q.

Lorsque son intension est vide, le concept le plus général dans le treillis de concepts n'est pas considéré lors de l'identification des objets pertinents pour une requête. En effet dans un tel cas, il peut exister des objets dans l'extension de *top* qui ne possèdent aucun attribut de Y et ne vérifient donc pas le critère de pertinence (définition 27).

4.4.2 L'algorithme BR-Explorer

Dans cette section nous détaillons l'algorithme de recherche d'information par treillis BR-Explorer. Pour ce faire, nous considérons un contexte formel $\mathbb{K} = (G, M, I)$, le treillis de concepts qui lui correspond $\underline{\mathfrak{B}}(\mathbb{K})$ et une requête $Q = (X, Y)$.

Intuitivement BR-Explorer procède comme suit. D'abord la requête Q est insérée dans le treillis $\underline{\mathfrak{B}}(\mathbb{K})$ (Algorithme 1 ligne 1). Cette insertion produit un nouveau treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K}_Q)$ contenant le concept pivot $P = (Y', Y)$ (Algorithme 1 ligne 2). Comme mentionné précédemment, les objets pertinents pour la requête sont dans l'extension du concept pivot et de ses super-concepts. Donc pour identifier ces objets, il suffit de commencer la recherche dans le treillis $\underline{\mathfrak{B}}(\mathbb{K}_Q)$ à partir du concept pivot P et ensuite effectuer un parcours en largeur (vers le haut) du treillis jusqu'au concept le plus général, $\top$.

La localisation de P dans $\underline{\mathfrak{B}}(\mathbb{K}_Q)$ est détaillée dans la procédure *Localiser_Pivot* (Algorithme 2). L'idée de base de la procédure de recherche du concept pivot est l'élagage. La recherche de P commence à partir du concept *bottom* ($\perp$) et à chaque fois que l'intension B d'un concept $C = (A, B)$ ne vérifie pas $Y \subseteq B$, tous ses super-concepts sont ignorés puisque eux aussi ne vérifieront pas cette condition (car leurs intensions sont des parties de B).

Une fois que le concept pivot est localisé, nous procédons à la construction du résultat, noté $\mathcal{R}_{objets}$ (Algorithme 1 ligne 8 et 18). On note par $\mathcal{SUBS}_i$ l'ensemble de concepts considérés à l'itération i de BR-Explorer. La première étape consiste à considérer l'extension de P, le seul concept dans $\mathcal{SUBS}_0$, et à ajouter à $\mathcal{R}_{objets}$ les éventuels objets autres que x dans cette extension avec un rang égal à 1 (lignes 3 à 10 de BR-Explorer). À l'étape suivante nous considérons $SUBS_1 = c_sup(P)$. L'ensemble des objets dans les extensions des concepts de $\mathcal{SUBS}_1$ qui ne figurent pas encore dans le résultat (objets émergeants) sont ajoutés à $\mathcal{R}_{objets}$ avec leur rang correspondant (soit 1 si aucun objet n'a été ajouté au résultat à l'étape précédente, soit 2 sinon) (Algorithme 1 lignes 11 à 21). Le reste des étapes est constitué des itérations sur $\mathcal{SUBS}_i$ en effectuant le même traitement que sur $\mathcal{SUBS}_1$ jusqu'à atteindre un ensemble $\mathcal{SUBS}_i$ qui est vide, auquel cas BR-Explorer s'arrête et retourne le résultat $\mathcal{R}_{objets}$ obtenu. À chaque étape i, si le concept $\top$ appartient à $SUBS_i$ et si son intension est vide alors les objets figurants dans son extension sont ignorés.

4.4.3 Déroulement de BR-Explorer sur un exemple

Considérons le contexte formel $\mathbb{K} = (G, M, I)$ donné dans la table 4.2, le treillis de concepts correspondant, $\underline{\mathfrak{B}}(\mathbb{K})$, donné à la figure 4.1 et la requête $Q = (X, Y)$ avec $Y =$*{Mammals, Complete, Updated}* dont le but de *chercher les sources de données de BioRegistry contenant des données* ***complètes*** *et* ***à jour*** *sur les* ***mammifères***.

L'ajout de Q au contexte formel $\mathbb{K} = (G, M, I)$ à l'aide de l'opérateur "$\oplus$" produit le contexte formel $\mathbb{K}_{\mathbb{Q}} = (G_Q, M_Q, I_Q) = \mathbb{K} \oplus Q$ donné dans la table 4.3.

Le treillis de concepts résultant de l'insertion de Q dans le treillis $\underline{\mathfrak{B}}(\mathbb{K})$, noté par $\underline{\mathfrak{B}}(\mathbb{K}_Q)$, est donné à la figure 4.2.

Sur cette figure nous avons schématisé les étapes du déroulement de l'algorithme BR-Explorer pour la requête $Q = (\{x\}, \{Complete, Mammals, Updated\})$. Les traitements effectués dans chacune de ces étapes sont respectivement les suivantes :

1. Localisation du concept pivot P dans le treillis $\underline{\mathfrak{B}}(\mathbb{K}_Q)$ à l'aide de la procédure *Localiser_Pivot*. Ici $P = (\{BD_5, BD_7, x\}, \{Complete, Mammals, Updated\})$ et $\mathcal{SUBS}_0 = \{P\}$. $Y' = \{BD_5, BD_7, x\} \neq X$ (BR-Explorer ligne 6) donc les sources de données BD_5 et BD_7

Algorithm 1 BR-Explorer

Require: $\mathbb{K} = (G, M, I)$, $\underline{\mathfrak{B}}(\mathbb{K})$ et $Q = (X, Y)$
Ensure: R_{objets}
1: Insérer Q dans $\underline{\mathfrak{B}}(\mathbb{K})$
2: $P = (Y', Y) := Localiser_Pivot(\underline{\mathfrak{B}}(\mathbb{K}_Q), Q)$
3: $n := 1$ \\ n est le niveau dans le treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K}_Q)$ à partir de P
4: $\mathcal{SUBS}_{n-1} := \{P\}$
5: $rang := 1$
6: **if** $Y' \neq X$ **then**
7: $\quad \mathcal{R}_{rang} := Y' \backslash X$
8: $\quad \mathcal{R}_{objets} := (rang, \mathcal{R}_{rang})$
9: $\quad rang := rang + 1$
10: **end if**
11: **while** $\mathcal{SUBS}_{n-1} \neq \emptyset$ **do**
12: $\quad \mathcal{SUBS}_n := c_sup(\mathcal{SUBS}_{n-1})$
13: $\quad \mathcal{R}_{rang} := \emptyset$
14: $\quad$ **for all** $C = (A, B) \in \mathcal{SUBS}_n$ tel que $B \neq \emptyset$ **do**
15: $\quad\quad \mathcal{R}_{rang} := \mathcal{R}_{rang} \cup A$
16: $\quad$ **end for**
17: $\quad ObjetsEmergeants := \mathcal{R}_{rang} \backslash (X \cup \mathcal{R}_1, \mathcal{R}_2, ..., \mathcal{R}_{rang-1})$
18: $\quad \mathcal{R}_{objets} := \mathcal{R}_{objets} \cup (rang, ObjetsEmergeants)$
19: $\quad n := n + 1$
20: $\quad rang := rang + 1$
21: **end while**

Algorithm 2 Localiser_Pivot

Require: $\underline{\mathfrak{B}}(\mathbb{K}_Q)$ et $Q = (X, Y)$
Ensure: $P = (Y', Y)$
1: trouvé := faux
2: $\mathcal{SUBS} := \perp$ \\ $\perp$ est le concept bottom dans $\underline{\mathfrak{B}}(\mathbb{K}_Q)$
3: **while** trouvé = faux **do**
4: $\quad$ **for** tout $C = (A, B) \in \mathcal{SUBS}$ **do**
5: $\quad\quad$ **if** $Y = B$ **then**
6: $\quad\quad\quad P := C$
7: $\quad\quad\quad$ trouvé := vrai
8: $\quad\quad\quad$ break
9: $\quad\quad$ **else if** $Y \subset B$ **then**
10: $\quad\quad\quad \mathcal{SUBS} := c_sup(\mathcal{SUBS})$
11: $\quad\quad\quad$ break
12: $\quad\quad$ **end if**
13: $\quad$ **end for**
14: **end while**

TABLE 4.3 – Ajout de la requête $Q = (X, \{Mammals, Complete, Updated\})$ au contexte $\mathbb{K} = (G, M, I)$.

	Organism				Content Quality		Ontology	
	Amphibiens	Birds	Fishes	Mammals	Complete	Updated	GO	NCBI
BD_1	×		×		×			×
BD_2	×		×		×	×		×
BD_3	×			×				×
BD_4		×		×		×		
BD_5	×			×	×	×		
BD_6		×		×	×		×	
BD_7		×		×	×	×	×	×
BD_8		×		×				×
x				×	×	×		

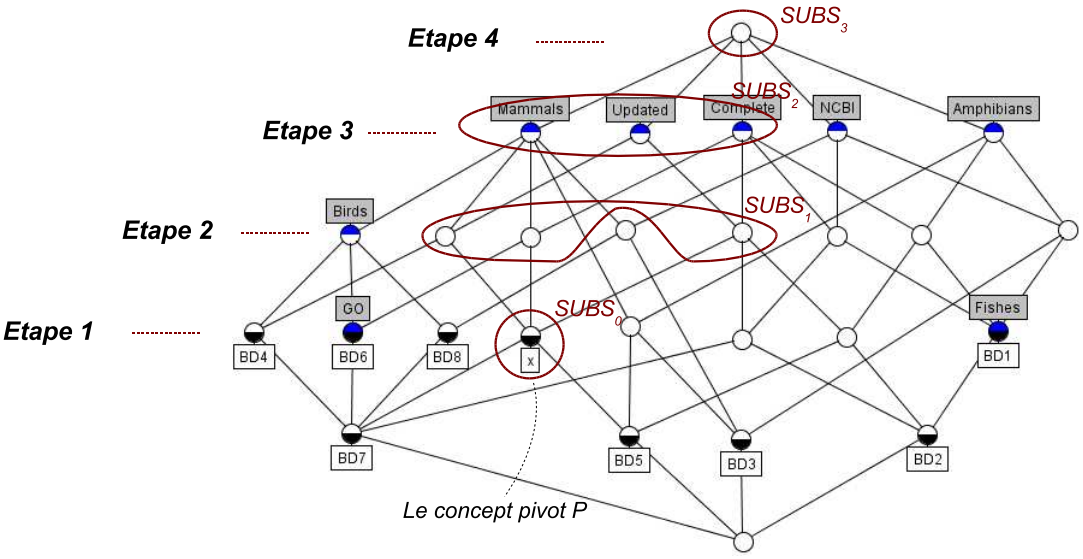


FIGURE 4.2 – Le treillis de concepts résultant de l'insertion de la requête.

sont ajoutées au résultat $\mathcal{R}_{objets}$ (jusque là vide) avec un rang égal à 1, le rang est ensuite incrémenté de 1 (BR-Explorer ligne 9).

2. Comme $\mathcal{SUBS}_0 \neq \varnothing$ (BR-Explorer ligne 11), nous calculons $\mathcal{SUBS}_1$:
   $\mathcal{SUBS}_1 = c_sup(\mathcal{SUBS}_0)$
   $= \{$
   $(\{BD_4,BD_5,BD_7,x\},\{Mammals,\ Updated\}),$
   $(\{BD_5,BD_6,BD_7,x\},\{Complete,\ Mammals\}),$
   $(\{BD_2,BD_5,BD_7,x\},\{Complete,\ Updated\})$
   $\}.$
   Les objets qui émergent à cette étape sont BD_2, BD_4 et BD_6, qui sont ajoutés à $\mathcal{R}_{objets}$ avec un rang égal à 2 (BR-Explorer ligne 18) et le rang est incrémenté de 1.
3. Comme $\mathcal{SUBS}_1 \neq \varnothing$, $\mathcal{SUBS}_2$ est calculé :
   $\mathcal{SUBS}_2 = \{$
   $(\{BD_3,BD_4,BD_5,BD_6,BD_7,BD_8,x\},\{Mammals\}),$
   $(\{BD_2,BD_4,BD_5,BD_7,x\},\{Updated\}),$
   $(\{BD_1,BD_2,BD_5,BD_6,BD_7,x\},\{Complete\})$
   $\}.$
   Les objets qui émergent sont BD_1, BD_3 et BD_8 et sont ajoutés à $\mathcal{R}_{objets}$ au rang 3.
4. À cette étape, $\mathcal{SUBS}_3$ est formé du seul concept $\top$, qui a une intension vide : il n'y a donc pas d'objets retournés.
5. Cette étape correspond à l'arrêt de l'algorithme puisque $\mathcal{SUBS}_4 = \varnothing$. Le résultat $\mathcal{R}_{objets}$ est retourné.

Finalement le résultat obtenu en réponse à la requête $Q = (X,\{Mammals,\ Complete,\ Updated\})$ est :

1- $BD5$ pour le partage des attributs *Complete, Mammals* et *Updated*
   $BD7$ pour le partage des attributs *Complete, Mammals* et *Updated*
2- $BD4$ pour le partage des attributs *Mammals* et *Updated*
   $BD6$ pour le partage des attributs *Complete* et *Mammals*
   $BD2$ pour le partage des attributs *Complete* et *Updated*
3- $BD1$ pour le partage de l'attribut *Complete*
   $BD3$ pour le partage de l'attribut *Mammals*
   $BD8$ pour le partage de l'attribut *Mammals*

4.4.4 Expressivité des requêtes dans BR-Explorer

Étant donné une requête $Q = (X, Y)$, l'algorithme BR-Explorer cherche d'abord les objets qui partagent tous les attributs dans Y (les objets dans l'extension du concept pivot P) et ensuite ceux qui partagent seulement des parties non vides de Y dans l'ordre décroissant sur le nombre d'attributs partagés. À chaque étape les objets retrouvés partagent un attribut ou une conjonction d'attributs de la requête. De cette manière, le traitement d'une requête dans BR-Explorer est équivalent au traitement d'une suite de requêtes conjonctives. La première requête est formée par la conjonction de tous les attributs dans Y. Les requêtes suivantes sont formées par la conjonction des sous ensembles de Y jusqu'aux requêtes formées par un seul attribut. Le résultat final est formé par l'union des résultats de chaque requête en classant en premier les réponses aux requêtes ayant le plus d'attributs partagés. Dans le cas ou un objet appartient à plusieurs résultats intermédiaires, le rang le plus petit est considéré dans le résultat final (i.e. la première fois où l'objet apparaît).

L'illustration de cette décomposition de la requête en un ensemble de requêtes conjonctives est donnée dans le cas de la requête $Q = (X, \{Mammals, Complete, Updated\})$. Les requêtes conjonctives et les résultats qui leurs correspondent sont détaillés comme suit.

$Mammals \sqcap Complete \sqcap Updated$ – $BD5$ et $BD7$
$Mammals \sqcap Complete$ – $BD4$, $BD5$ et $BD7$
$Mammals \sqcap Updated$ – $BD5$, $BD6$ et $BD7$
$Complete \sqcap Updated$ – $BD2$, $BD5$ et $BD7$
$Mammals$ – $BD3$, $BD4$, $BD5$, $BD6$, $BD7$ et $BD8$
$Complete$ – $BD1$, $BD2$, $BD5$, $BD6$ et $BD7$
$Updated$ – $BD2$, $BD4$, $BD5$ et $BD7$

4.4.5 Correction et complétude de BR-Explorer

Correction

Définition 28 *Étant donné un contexte formel $\mathbb{K} = (G, M, I)$ et une requête $Q = (X, Y)$, l'algorithme BR-Explorer est correct par rapport au critère de pertinence si tout objet retrouvé par l'algorithme est pertinent pour la requête Q.*

Proposition 2 *L'algorithme BR-Explorer est correct par rapport au critère de pertinence donné à la définition 27.*

Preuve 2 *Considérons une requête $Q = (X, Y)$ et un objet g retrouvé par l'algorithme BR-Explorer ($g \in \mathcal{R}_{objets}$).*

Selon la proposition 1, si g est retrouvé par BR-Explorer alors g ($\neq \varnothing$) appartient soit à l'extension du concept pivot P soit à l'extension de l'un des super-concepts de P dans le treillis de concepts. Dans les deux cas, nous avons $\{g\}' \cap Y \neq \varnothing$, ce qui prouve la pertinence de g pour la requête Q (définition 27).

Complétude

Définition 29 *L'algorithme BR-Explorer est complet par rapport au critère de pertinence si tout objet de l'ensemble G pertinent pour Q est retrouvé par l'algorithme.*

Proposition 3 *L'algorithme BR-Explorer est complet par rapport au critère de pertinence donné à la définition 27.*

Preuve 3 *Considérons une requête $Q = (X, Y)$ et un objet $g \in G$ pertinent pour Q. Selon la définition de la pertinence (définition 27), nous avons $\{g\}' \cap Y \neq \varnothing$. Nous pouvons distinguer les deux cas suivants :*

- *Si $\{g\}' \subset Y$ alors il existe $C = (A, B) \in \underline{\mathfrak{B}}(\mathbb{K}_Q)$ tel que $B = \{g\}'$ et $g \in A$ ($C = (\{g\}'', \{g\}') = (A, \{g\}')$). Cela signifie que C est un super-concept du concept pivot P dans le treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K}_Q)$, et que g est dans l'extension de ce super-concept de P. Selon la proposition 1, g est retrouvé par l'algorithme BR-Explorer.*
- *Si $Y \subseteq \{g\}'$ alors $\{g\}'' \subseteq Y'$. Et puisque $g \in \{g\}''$ alors $g \in Y'$. Cela signifie que l'objet g appartient à l'extension du concept pivot $P = (Y', Y)$. Selon la proposition 1, g est retrouvé par l'algorithme BR-Explorer.*

Les propositions 2 et 3 nous permettent d'énoncer le théorème suivant.

Théorème 3 *Étant donné un contexte formel $\mathbb{K} = (G, M, I)$ et une requête $Q = (X, Y)$. L'algorithme de recherche d'information par treillis BR-Explorer (algorithme 1) est correct et complet par rapport au critère de pertinence donné à la définition 27.*

4.4.6 Complexité de BR-Explorer

Le calcul de la complexité de BR-Explorer peut être divisé en trois parties : (1) le calcul de la complexité de l'insertion d'une requête dans le treillis de concept, (2) le calcul de la complexité de la recherche du concept pivot dans le treillis résultant et (3) le calcul de la complexité de la construction du résultat.

1. La complexité de l'insertion d'une requête dans un treillis de concept dépend de l'algorithme de construction incrémentale de treillis de concepts choisi pour effectuer cette opération. Les complexités des algorithmes de construction incrémentale sont données en section 3.3.6.
2. Simple recherche $O((|M| - |Y|)(|M| - |Y| - 1)/2)$
3. Simple parcours des super-concepts du pivot $O(2^{|Y|})$.

La complexité de l'exploitation du treillis est négligeable par rapport à la construction du treillis. La complexité totale est de l'ordre le la complexité de construction du treillis.

4.5 Raffinement de requête à partir de ressources sémantiques

Le raffinement de requête consiste à reformuler une requête en y ajoutant des attributs à partir d'une ou de plusieurs ressources sémantiques (ontologies de domaine, thésaurus, taxonomies, etc.) [Messai et al., 2005, Messai et al., 2006a]. Les attributs ajoutés doivent être en relation sémantiques avec ceux de la requête. Le but du raffinement de requête est de fournir une réponse approximative à une requête lorsqu'une réponse exacte ne peut être trouvée.

Pour illustrer l'idée du raffinement de requête, nous considérons un nouveau jeu de données extrait de BioRegistry qui met en evidence les relations sémantiques entre attributs. Ce jeu de données est formalisé dans le contexte donné dans la table 4.4. Le treillis de concepts correspondant est donné à la figure 4.3. Les relations sémantiques entre les attributs de ce contexte sont extraites d'une taxonomie dont une partie est schématisée à la figure 4.4. Cette partie est extraite de Tree of life[27] et Taxo[28] en ne retenant que les organismes modèles les plus étudiés en génomique (les feuilles de l'arbre) et les nœuds structurants (le reste des nœuds). Cette taxonomie relie les méta-données relatives au type d'organisme couvert par les sources de données dans BioRegistry.

Dans la formalisation du raffinement de requête à partir d'uen taxonomie, nous considérons que la taxonomie est représentés sous la forme d'un arbre $\mathcal{T} = (V, E)$. Les sommets dans V sont des méta-données et les arcs dans E matérialisent les liens sémantiques entre les méta-données. A chaque lien correspond deux interprétations selon l'orientation qu'on considère. Considérons un arc entre deux sommets v_i, v_j de V tels que v_j est un descendant de v_i dans $\mathcal{T}$. Cet arc matérialise deux relations sémantiques entre v_i et v_j : d'une part v_j *est un* v_i et la relation est dite de généralisation et d'autre part, v_i *contient* v_j et la relation est dite de spécialisation. Par exemple, dans TaxoBR (figure 4.4), *Animals* est un *Eukaryotes* et *Eukaryotes* contient *Animals*.

Le raffinement de requête consiste à ajouter à la requête utilisateur de nouveaux attributs à partir de la taxonomie (ou toute autre ressource sémantique) disponible. Il s'agit donc, pour

27. http ://tolweb.org/tree/phylogeny.html
28. http ://www.ncbi.nlm.nih.gov/entrez/query.fcgi ?db=Taxonomy

TABLE 4.4 – Un contexte formel représentant une partie de BioRegistry.

	Subject			Organism					Content Quality
	Any Sequence (AS)	Nucleic Sequence (NS)	Proteic Sequence (PS)	Any Organism (AO)	Animals (An)	Human (Hu)	Mouses (Mo)	Vertebrates (Ve)	Manual Revision (MR)
Swissprot			×	×					×
RefSeq	×	×	×	×					×
TIGR-HGI		×				×			
GPCRDB			×	×					×
HUGE	×	×	×			×			
ENSEMBL		×			×				
Mouse Genome DB (M G BD)			×				×		
Vega Genome Browser (V G B)			×					×	

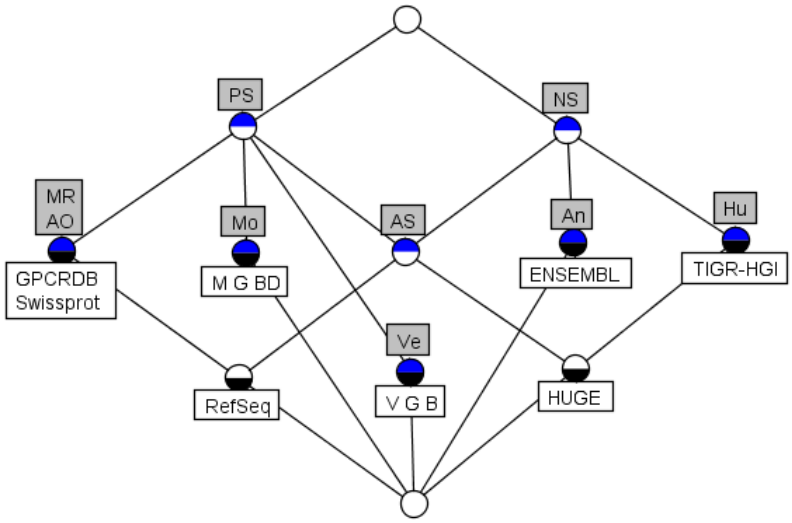


FIGURE 4.3 – Le treillis de concepts correspondant au contexte formel donné dans la table 4.4.

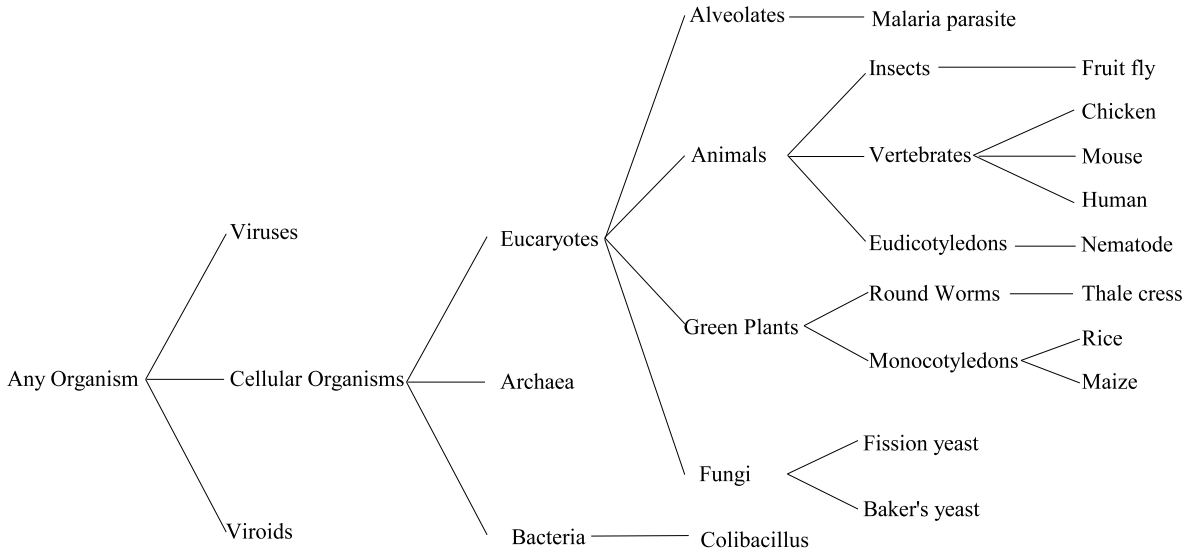


FIGURE 4.4 – *TaxoBR* : une partie de la taxonomie des organismes.

une requête donnée, de considérer les attributs figurant dans la taxonomie. Pour chacun de ces attributs, on effectue un parcours de la taxonomie pour en extraire des termes qui seront ajoutés en tant qu'attributs à la requête initiale. On distingue deux types de parcours d'une taxonomie, un parcours par généralisation et un parcours par spécialisation. Ces deux modes reflètent respectivement l'enrichissement par généralisation et l'enrichissement par spécialisation que nous allons présenter dans ce qui suit.

4.5.1 Raffinement de requête par généralisation

Le raffinement par généralisation relatif à un attribut $y \in Y$ d'une requête $Q = (X, Y)$ consiste à ajouter à Y des attributs plus généraux que y représentés par les ancêtres de y dans la taxonomie $\mathcal{T}$ considérée. Dans l'exemple d'une requête $Q = (X, \{Chicken\})$, le raffinement par généralisation de Q par rapport à l'attribut *Chicken* à partir de la taxonomie *TaxoBR* consiste à considérer les méta-données *Vertebrates, Animals, Eucaryotes, Cellular Organisms*, et *Any Organism*. Cependant, certaines de ces méta-données ne figurent pas en tant qu'attributs dans le contexte formel $\mathbb{K} = (G, M, I)$ donné dans la tableau 4.4 (*Eucaryotes* et *Cellular Organisms* ne sont pas dans M). Cela signifie que ces méta-données ne caractérisent aucune source de données du contexte $\mathbb{K}$, ce qui rend inutile leur ajout à la requête. Seules les méta-données présentes en tant qu'attributs dans $\mathbb{K}$ sont considérées lors du processus de raffinement par généralisation.

De manière générale, le raffinement d'une requête $Q = (X, Y)$ par généralisation à partir d'une taxonomie $\mathcal{T} = (V, E)$ et par rapport à un attribut $y \in Y$ vérifiant $y \in V$ consiste à transformer la requête $Q = (X, Y)$ en la requête enrichie $Q_E = (X, Y_E)$ telle que :

$$Y_E = (Y \cup \{v \in p\}) \cap M$$

où p est le chemin de y à la racine de $\mathcal{T}$.

Dans l'exemple de requête $Q = (X, \{Chicken\})$, l'enrichissement par généralisation par rapport à *Chicken* à partir de *TaxoBR* produit la requête enrichie $Q_E = (X, \{$*Vertebrates, Animals, Any Organism*$\})$.

Une fois que la requête enrichie est obtenue, l'algorithme BR-Explorer peut être appliqué pour retrouver les objets pertinents pour la nouvelle requête. Dans l'exemple de requête enrichie par généralisation donné précédemment, l'application de BR-Explorer produit la réponse suivante à la requête enrichie :

1 -	ENSEMBL	partage l'attribut *Animals* avec Q_E
	Vega Genome Browser	partage l'attribut *Vertebrates* avec Q_E
	Swissprot	partage l'attribut *Any Organism* avec Q_E
	RefSeq	partage l'attribut *Any Organism* avec Q_E
	GPCRDB	partage l'attribut *Any Organism* avec Q_E

Chaque source de données du résultat précédent comprend une partie qui correspond à ce qui est demandé dans la requête (par exemple *Vega Genome Browser*, qui concerne les organismes vertébrés, concerne *Chicken*, l'attribut de la requête, mais aussi *Mouse* et *Human*). Plus la distance, dans l'ontologie, entre la méta-donnée de la requête initiale et celle ajoutée lors du raffinement est petite, plus la source résultante est pertinente (*Vega Genome Browser* retrouvée grâce à *Vertebrates* est préférée à *ENSEMBL* retrouvée grâce à *Animals*). Cela justifie l'intérêt de contrôler les méta-données ajoutées lors du raffinement par généralisation de requêtes. Il est possible d'éviter l'introduction de sources de faible pertinence dans le résultat en ne considérant que les ancêtres les plus proches de la méta-donnée considérée.

Interprétation

L'enrichissement par généralisation permet d'obtenir une réponse enrichie par des sources plus générales que celles demandées par l'utilisateur vis-à-vis de la propriété appartenant à l'ontologie et figurant dans la requête initiale. Cet enrichissement s'appuie sur le fait qu'une source qui étudie un ensemble d'organismes donnés peut contenir des informations sur l'ensemble plus spécialisé qui interesse l'utilisateur. Par exemple lorsqu'un utilisateur demande une source relative à l'organisme "poulet", une source relative aux organismes vertébrés (plus général dans l'ontologie) peut l'intéresser puisqu'elle est susceptible de contenir des informations sur les poulets.

4.5.2 Raffinement de requête par spécialisation

De façon duale au raffinement par généralisation, le raffinement par spécialisation par rapport à un attribut $y \in Y$ d'une requête $Q = (X, Y)$ consiste à ajouter à Y les attributs plus spécifiques que y représentés par les descendants de y dans la taxonomie $\mathcal{T}$ considérée. Les descendants de y dans $\mathcal{T}$ forment un sous arbre de $\mathcal{T}$ de racine y qu'on notera par $\mathcal{T}_y = (V_y, E_y)$. De la même manière que dans le cas de raffinement par généralisation, seuls les sommets de V_y qui apparaissent en tant qu'attributs dans le contexte $\mathbb{K}$ sont considérés. Ainsi le raffinement par spécialisation de $Q = (X, Y)$ par rapport à un attribut $y \in Y$ produit la requête enrichie $Q_E = (X, Y_E)$ telle que

$$Y_E = (Y \cup V_y) \cap M.$$

Dans l'exemple de contexte donné dans la table 4.4, considérons l'exemple de requête $Q = (X, \{Eukaryotes\})$. L'enrichissement par spécialisation de Q par rapport à *Eukaryotes* revient à considérer la partie de *TaxoBR* (figure 4.4) constituée du sous arbre de racine *Eukaryotes*. Parmi les termes appartenant à cette partie de la taxonomie, seuls ceux apparaissant comme attributs dans $\mathbb{K}$ sont ajoutés à la requête. Ces termes sont *Animals, Vertebrate, Human* et *Mouse*. L'application de BR-Explorer sur la requête enrichie $Q_E = (X, \{Animals, Vertebrates, Human, Mouses\})$ retourne la réponse suivante :

1 -	ENSEMBL	partage l'attribut *Animals* avec Q_E
	Vega Genome Browser	partage l'attribut *Vertebrates* avec Q_E
	HUGE	partage l'attribut *Human* avec Q_E
	TIGR-HGI	partage l'attribut *Human* avec Q_E
	Mouse Genome DB	partage l'attribut *Mouses* avec Q_E

Dans cette réponse, chaque source de données apparaissant dans le résultat donne une réponse partielle à la requête. La composition de ces sources de données permet d'obtenir une réponse complète si chaque descendant de la méta-donnée de la requête indexe une source de données. De façon similaire au raffinement par généralisation, la distance entre la méta-donnée considérée dans la requête initiale et ses descendants dans la taxonomie reflète le degré de pertinence des sources de données présentes dans le résultat (par exemple *ENSEMBL* retrouvée grâce à *Animals* est préférée à *HUGE* et *Mouse Genome DB* retrouvées respectivement grâce à *Human* et *Mouses*). En effet, les sources de données relatives aux méta-données très distantes de la méta-donnée de la requête donnent des informations très spécialisées qui ne sont pas toujours pertinentes pour l'utilisateur. Le degré de spécialisation peut être contrôlé en ne considérant que les descendants dans la taxonomie les plus proches de la méta-donnée de la requête et constituant la meilleure couverture de la requête.

Interprétation

Du point de vue biologique, les descendants d'un attribut de raffinement y dans $\mathcal{T}$ sont des spécialisations de y. Leur ajout à la requête a pour but d'enrichir le résultat par des sources de données répondant à une partie de la requête initiale. Par exemple lorsqu'un utilisateur demande une source relative aux eucaryotes, une source relative aux animaux (un sous ensemble des eucaryotes) peut l'intéresser dans le sens ou elle contient des informations sur une partie du groupe d'organismes qu'il demande.

4.5.3 Raffinement mixte

Il est possible de combiner les deux types de raffinement de requête en ajoutant à la requête à la fois les descendants et les ancêtres de la méta-donnée considérée dans la taxonomie. Dans les deux types de raffinement, le nombre de méta-données ajoutées à la requête peut être contrôlé, en considérant uniquement les ancêtres les plus proches de la méta-donnée considérée dans la taxonomie (dans le cas du raffinement par généralisation) et/ou ses descendants les plus proches (dans le cas du raffinement par spécialisation). Une fois que le raffinement de requête est effectué, l'algorithme BR-Explorer est appliqué en considérant la requête enrichie obtenue.

4.5.4 Choix du type de raffinement

Dans le cas où la méta-donnée considérée est une feuille ou la racine de la taxonomie, le problème du choix ne se pose pas puisque dans les deux cas un seul type de raffinement est possible (raffinement par généralisation pour une feuille et par spécialisation pour la racine). Cependant, lorsque la méta-donnée considérée n'est ni une feuille ni la racine de la taxonomie, les deux types de raffinement sont possibles et le choix peut être fait selon les préférences de l'utilisateur. En effet, si l'utilisateur accepte des sources de données dont une partie seulement correspond à sa requête, alors le raffinement par généralisation peut être choisi. Si par contre il accepte des sources de données correspondant à une partie seulement de sa requête, alors le raffinement par spécialisation peut être choisi. La composition de sources de données correspondant partiellement à une requête peut alors permettre d'obtenir une réponse complète. Dans les deux cas, il est utile

de procéder à un ordonnancement à posteriori des sources de données ajoutées au résultat suite au raffinement de la requête. Cet ordonnancement doit être basé sur la similarité entre les méta-données annotant ces sources, d'une part et les méta-données constituant la requête, d'autre part [Ganesan et al., 2003].

4.6 Dépendances entre attributs : hiérarchie d'attributs dans BR-Explorer

4.6.1 Le besoin d'exprimer les dépendances entre attributs

Dans la découverte de ressources par treillis de concepts, les requêtes sont définies par un ensemble d'attributs caractérisant les objets (ressources) cherchés. Cette représentation ne permet pas l'expression de relations sémantiques qui peuvent exister entre les attributs d'une requête. Cependant, la définition de telles relations est indispensable pour que les requêtes soient correctement interprétées et par conséquence que les résultats retournés correspondent mieux aux objectifs de la recherche. Ce besoin a été exprimé de manière explicite par Carpineto et Romano dans [Carpineto and Romano, 2005]. L'exemple avancé pour illustrer ce besoin est la requête qui consiste à "*retrouver les restaurants italiens près du musée du Louvre et ayant une terrasse*". Les objets recherchés par cette requête sont des restaurants et les attributs qui les décrivent sont "*italiens*", "*près du musée du Louvre*" et "*ayant une terrasse*". Un restaurant vérifiant tous les attributs de la requête serait une réponse idéale. Cependant, si un tel restaurant n'existe pas, il est intéressant de savoir quel(s) attribut(s) doi(ven)t être considéré(s) en priorité. Par exemple la localisation géographique en premier (près du musée du Louvre), ensuite le type de cuisine (Italien) et enfin l'existence d'une terrasse (ayant une terrasse) [Carpineto and Romano, 2005].

Dans le cas des données de BioRegistry, la complexité des méta-données a accentué le besoin d'étendre le language de requêtes afin de permettre la définition de préférences sur les attributs d'une requête. En effet, une première exploration du contenu de BioRegistry pourrait s'intéresser aux sources de données contenant des informations sur *l'espèce humaine*. L'attribut relatif à ce nom d'espèce, *Human*, sera préféré aux autres attributs. Si plusieurs sources de données contiennent des informations sur l'espèce humaine, celles dont les informations ont été mises à jour récemment seront plus intéressantes à explorer. Si une base de données contient des informations à jour mais relatives à une espèce autre que l'espèce humaine, elle ne sera pas utile à explorer. On considère dans ce cas que l'attribut "*Updated*" donne une information supplémentaire. Dans une deuxième exploration, on s'intéresse aux bases de données contenant les informations les plus récentes afin de réaliser une étude statistique. Dans ce cas l'attribut "*Updated*" devient préféré. De façon similaire à l'exemple précédent, une information sur le nom de l'espèce traitée dans chaque source de données peut être significative dans le cas où il s'agit d'une source de données dont le contenu est à jour. Ces deux exemples illustrent la variation de l'importance à accorder aux attributs dans un contexte formel en fonction de l'exploitation des données qu'il contient. Dans les deux cas on peut distinguer un attribut principal qui doit être présent pour qu'une base de données soit considérée intéressante, et un (ou plusieurs) attribut(s) secondaires qui donnent des informations complémentaires.

Dans la suite de cette section nous proposons une méthode qui répond aux besoins exprimés précédemment. La méthode généralise la notion de Formules de dépendance entre attributs proposé par Belohlavek et al. dans [Belohlávek et al., 2004, Belohlávek and Sklenar, 2005] et l'étend à la découverte de ressources par treillis de concepts [Messai et al., 2008b].

4.6.2 Formalisation

Dans cette formalisation nous commençons par rappeler la définition des formules de dépendance entre attributs (ADFs) introduite dans [Belohlávek et al., 2004, Belohlávek and Sklenar, 2005].

Définition 30 (Formule de dépendance entre attributs) *Soient m, m_1, ... , m_n des attributs dans M. Une formule φ de dépendance entre attributs est de la forme :*

$$m \sqsubseteq m_1 \sqcup m_2 \sqcup ... \sqcup m_n$$

m_1, ... , m_n sont appelés attributs principaux et m est appelé attribut secondaire.

Une formule de dépendance entre attributs s'interprète de la façon suivante : m est un attribut secondaire dont la présence est prise en compte uniquement lorsqu'il est accompagné par au moins un des attributs principaux m_1, ... , m_n. Autrement dit, si m apparaît dans l'intension d'un concept formel alors m_1 ou m_2 ou ... ou m_n apparaissent aussi dans cette même intension. La principale application qui a motivé la définition des ADFs est la formation de catégories et la répartition des données représentées par un treillis de concepts au seins de ces catégories. Intuitivement, si on veut effectuer une classification des objets, on considère les attributs principaux comme discriminants pour définir les classes d'objets tandis que les attributs secondaires seront considérés comme des attributs de raffinement de la classification. La présence d'un attribut secondaire caractérisant un ensemble d'objets n'aura un sens que si elle est accompagnée par la présence d'un attribut principal.

Dans la suite nous étendons la définition d'ADFs pour la définition de hiérarchies d'attributs.

Définition 31 (Dépendance entre attributs) *(1) Un attribut m_2 est dependant d'un attribut m_1 lorsque la presence de m_2 n'est pas significative sans la présence de m_1. On note cette dépendance par $m_1 \succ m_2$.*
(2) Plus généralement, une dépendance entre attributs est de la forme $e_1 \succ e_2$ où e_1 et e_2 sont des attributs atomiques, des conjonctions d'attributs ou des disjonctions d'attributs.

Dans $e_1 \succ e_2$, les attributs dans e_2 sont moins importants que ceux dans e_1 et la présence de e_2 est significative uniquement lorsqu'elle est associée avec e_1. La figure 4.5 donne une représentation graphique des exemples simples de dépendances entre attributs.

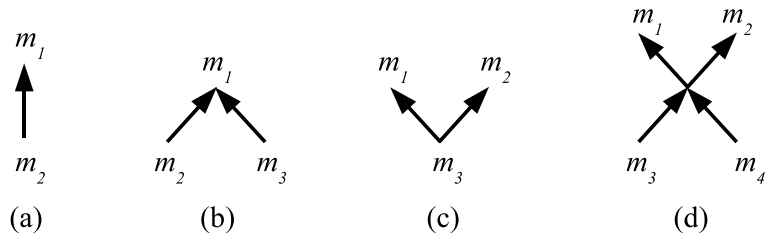


FIGURE 4.5 – Représentation graphique des exemples de dépendances entre attributs : (a)- $m_1 \succ m_2$, (b)- $m_1 \succ (m_2 \sqcup m_3)$ équivalente à $(m_1 \succ m_2) \sqcap (m_1 \succ m_3)$, (c)- $(m_1 \sqcap m_2) \succ m_3$ (équivalente à $(m_1 \succ m_3) \sqcap (m_2 \succ m_3)$), (d)- $(m_1 \sqcap m_2) \succ (m_3 \sqcup m_4)$ (équivalente à $(m_1 \succ m_3) \sqcap (m_2 \succ m_3) \sqcap (m_1 \succ m_4) \sqcap (m_2 \succ m_4)$).

L'opérateur de dépendance entre attributs "$\succ$" définit un ordre partiel sur un ensemble d'attributs et permet de former une hiérarchie d'attributs définie comme suit.

Définition 32 (Hiérarchie d'attributs) *Une hiérarchie d'attributs (HA), notée par $\mathcal{HA}$, est un ensemble partiellement ordonné $\mathcal{HA} = (A, \succ)$ où A est un ensemble d'attributs et "$\succ$" est un ordre partiel sur A.*

Les hiérarchies d'attributs correspondant aux deux exemples cités précédemment, en section 4.6.1, sont représentées à la figure 4.6. Ces hiérarchies se décrivent comme suit :

$$\mathcal{HA} : \text{“}Near\ the\ Louvre\ museum\text{”} \succ \text{“}Italian\text{”} \succ \text{“}with\ a\ dehors\text{”}$$

pour l'exemple de requête cherchant des restaurants [Carpineto and Romano, 2005] et

$$\mathcal{HA} : \text{“}Mammals\text{”} \succ (\text{“}Complete\text{”} \sqcup \text{“}Updated\text{”})$$

pour l'exemple de requête cherchant les sources de données dans BioRegistry.

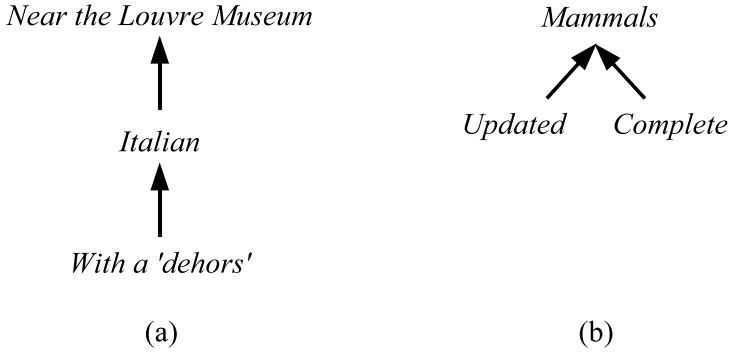


FIGURE 4.6 – La représentation graphique de deux exemples de hiérarchies d'attributs. (a) La hiérarchie d'attributs obtenue par la définition des dépendances entre attributs de la requête cherchant les restaurants *italiens près du musée du Louvre* et *ayant une terrasse* [Carpineto and Romano, 2005] : $\mathcal{HA}$:*Near the Louvre museum* $\succ$ *Italian* $\succ$ *with a "dehors"*. (b) La hiérarchie d'attributs obtenue par la définition des dépendances entre attributs de la requête cherchant les sources de données de BioRegistry contenant des données relatives aux *mammifères* et ayant un contenu *complet* et *à jours* : $\mathcal{HA}$: *Mammals* $\succ$ *(Complete* $\sqcup$ *Updated).*

4.6.3 Application des hiérarchies d'attributs à un treillis de concepts

La prise en compte d'une HA dans un treillis de concepts consiste à restreindre la structure du treillis aux concepts qui vérifient les dépendances entre attributs exprimées dans la HA considérée. Un concept vérifie une dépendance entre attributs lorsque son intension ne contient pas un attribut secondaire sans l'attribut principal duquel il est dépendant. Un tel concept est dit cohérent vis-à-vis de la dépendance considérée. Formellement, on définit la cohérence d'un concept formel comme suit.

Définition 33 (Cohérence de concepts formels) *(1) Un concept (A, B) est dit cohérent vis-à-vis d'une dépendance entre attributs $e_1 \succ e_2$ si et seulement si lorsque e_2 est vérifiée dans B alors e_1 est aussi vérifiée dans B. Dans ce cas, on dit que le concept (A, B) satisfait la dépendance $e_1 \succ e_2$. Selon la forme des expressions e_1 et e_2 dans "$e_1 \succ e_2$", on distingue les cas suivants pour la cohérence de (A, B).*

1. $m_1 \succ m_2$ *:* (A, B) *est cohérent si* $m_2 \in B$ *alors* $m_1 \in B$.
2. $(m_1 \sqcup ... \sqcup m_n) \succ m_p$ *:* (A, B) *est cohérent si* $m_p \in B$ *alors* $\exists i \in \{1..n\}$ *tel que* $m_i \in B$.
3. $(m_1 \sqcap ... \sqcap m_n) \succ m_p$ *:* (A, B) *est cohérent si* $m_p \in B$ *alors* $\forall i \in \{1..n\}$, $m_i \in B$.
4. $m_p \succ (m_1 \sqcup ... \sqcup m_n)$ *:* (A, B) *est cohérent si* $\exists i \in \{1..n\}$ *tel que* $m_i \in B$ *alors* $m_p \in B$.
5. $m_p \succ (m_1 \sqcap ... \sqcap m_n)$ *:* (A, B) *est coherent si* $\forall i \in \{1..n\}$, *si* $m_i \in B$ *alors* $m_p \in B$.
6. *Les cas précédents peuvent être combinés pour la définition de la cohérence dans le cas général* $e_1 \succ e_2$ *où* e_1 *et* e_2 *sont des expressions arbitraires.*

(2) Un concept (A, B) *est dit cohérent vis-à-vis d'une hiérarchie d'attributs* $\mathcal{HA}$ *si et seulement si* (A, B) *est cohérent vis-à-vis de toute dépendance entre attributs dans* $\mathcal{HA}$.
(3) L'application d'une hiérarchie d'attributs $\mathcal{HA}$ *à un treillis de concepts* $\underline{\mathfrak{B}}(\mathbb{K})$ *consiste à supprimer les concepts non cohérents vis-à-vis de* $\mathcal{HA}$ *du treillis.*

Reprenons l'exemple de contexte formel représentant les données de BioRegistry donné dans la table 4.2 et le treillis de concepts correspondant de figure 4.1. Supposons que l'on s'intéresse à une exploration des sources de données de BioRegistry selon les organismes qu'elles traitent. Cela revient à considérer les attributs *Amphibians*, *Birds*, *Fishes* et *Mammals* comme attributs principaux. Les attributs relatifs à la qualité du contenu et aux ontologies sont des attributs secondaires. En terme de dépendances entre attributs cela donne la hiérarchie d'attributs suivante :

$$\mathcal{HA}_1 : (Mammals \sqcup Birds \sqcup Amphibiens \sqcup Fishes) \succ (Complete \sqcup Updated \sqcup GO \sqcup NCBI).$$

En appliquant $\mathcal{HA}_1$ au treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$ donné à la figure 4.1 les concepts incohérents vis-à-vis de cette hiérarchie d'attributs sont écartés. Un exemple de concept incohérent est $(\{BD_1, BD_2, BD_5, BD_6, BD_7\}, \{Complete\})$. Son incohérence est due au fait que son intension contient l'attribut secondaire *Complete* et ne contient aucun des deux attributs principaux *Amphibians*, *Birds*, *Fishes* et *Mammals*. L'interprétation de cette incohérence est la suivante : l'attribut *Complete* signifie que les contenus des sources de données $BD1$, $BD2$, $BD5$, $BD6$ et $BD7$ (extension du concept formel) sont complets, cependant il n'y a aucune information sur l'objet de ces contenus. Ceci fait que le concept n'est pas assez informatif quand aux espèces décrites dans les bases de données $BD1$, $BD2$, $BD5$, $BD6$ et $BD7$.

Un exemple de concept cohérent vis-à-vis de $\mathcal{HA}_1$ dans $\underline{\mathfrak{B}}(\mathbb{K})$ est $(\{BD5, BD6, BD7\}, \{Mammals, Complete\})$. Dans l'intension de ce concept, l'attribut secondaire *Complete* est accompagné de l'attribut principal *Mammifères*.

Dans un treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$, l'ensemble de concepts cohérents vis-à-vis d'une hiérarchie d'attributs $\mathcal{HA}$ est noté par $\mathfrak{B}_{\mathcal{HA}}(\mathbb{K})$ et la hiérarchie de concepts cohérents est notée par $\underline{\mathfrak{B}}_{\mathcal{HA}}(\mathbb{K})$. Belohlavek et al. ont prouvé dans [Belohlávek and Sklenar, 2005] que l'ensemble de concepts cohérents vis-à-vis d'un ensemble d'ADFs forme un treillis complet qui est un $\bigvee$*-sous-treillis* du treillis de concepts initial. Ce résultat reste vrai après la généralisation des ADFs aux hiérarchies d'attributs. De ce fait, $\underline{\mathfrak{B}}_{\mathcal{HA}}(\mathbb{K})$ est un treillis complet qui est un $\bigvee$*-sous-treillis* de $\underline{\mathfrak{B}}(\mathbb{K})$.

Considérons le treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$ donné à la figure 4.1 et la hiérarchie d'attributs $\mathcal{HA}_1$ donnée précédemment. L'application de $\mathcal{HA}_1$ sur $\underline{\mathfrak{B}}(\mathbb{K})$ produit $\underline{\mathfrak{B}}_{\mathcal{HA}_1}(\mathbb{K})$ représenté à la figure 4.7.

Dans $\mathfrak{B}_{\mathcal{HA}_1}(\mathbb{K})$, on remarque que les attributs principaux apparaissent en haut du treillis et que les attributs secondaires apparaissent toujours en dessous des attributs principaux. Cela s'explique par le fait que chaque attribut principal caractérise une classe de concepts (tous les

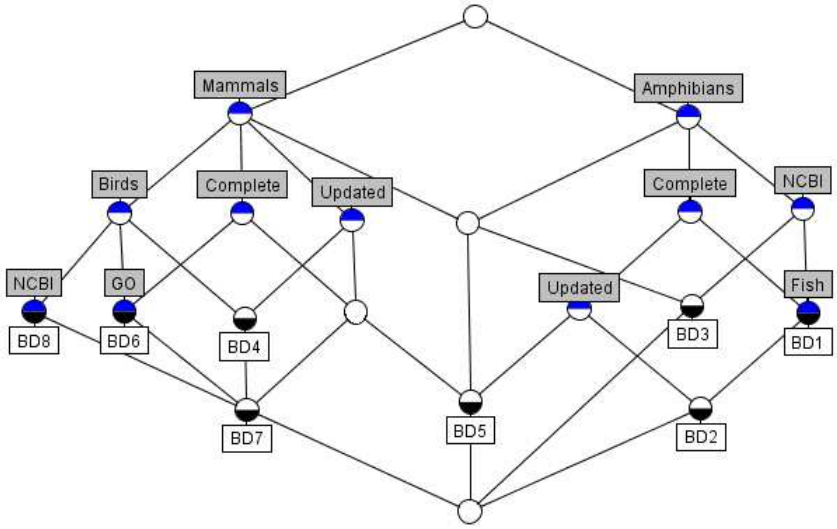


FIGURE 4.7 – La hiérarchie de concepts $\mathfrak{B}_{\mathcal{HA}_1}(\mathbb{K})$ résultant de l'application de $\mathcal{HA}_1$ au treillis $\underline{\mathfrak{B}}(\mathbb{K})$ donné dans Figure 4.1.

concepts ayant une intension qui contient cet attribut) alors que les attributs secondaires constituent des spécialisations dans chacune des classes de concepts définies par les attributs principaux. Cet aspect justifie la motivation de l'utilisation des dépendances entre attributs dans les treillis de concepts pour la définition de catégories par Belohlavek et al. [Belohlávek and Sklenar, 2005]. Dans la suite nous développons l'utilisation des hiérarchies d'attributs dans les treillis de concepts pour la découverte de ressources guidée par des connaissances.

4.6.4 Hiérarchies d'attributs pour la découverte de ressources guidée par des connaissances

Connaissances locales et connaissances globales

Une dépendance entre attributs matérialise une relation sémantiques entre ces attributs. Cette relation exprime des connaissances de domaine non représentées dans un contexte formel. Ces connaissances peuvent provenir soit de ressources externes telles que des ontologies de domaine soit de l'expertise. La représentation de telles connaissances sous la forme de hiérarchies d'attributs permet de les exprimer explicitement dans un treillis de concepts. Dans le cadre de notre travail, les hiérarchies d'attributs sont utilisées avec les treillis de concepts pour définir une méthode de découverte de ressources guidée par des connaissances de domaine. Cette méthode couvre à la fois la navigation et l'interrogation du treillis de concepts. Les connaissances de domaines exprimées sous la forme de hiérarchies d'attributs sont de deux types : locales ou globales. Les connaissances locales sont des connaissances considérées lors d'une session de découverte (navigation ou interrogation). Elles portent sur les attributs de la requête en cas d'interrogation et orientent la navigation en cas de navigation. Les connaissances locales n'affectent pas directement le treillis de concepts. Les connaissances globales sont des connaissances qui expriment un point de vue selon lequel la découverte de ressources est effectuée. Elle portent sur plusieurs sessions et affectent le treillis de concepts en le restreignant à l'ensemble des concepts cohérents vis-à-vis des hiérarchies d'attributs qui expriment ces connaissances. Selon que les hiérarchies d'attributs re-

présentent des connaissances locales ou globales, la prise en compte de connaissances de domaine abouti à une forme différente de découverte de ressources. Ces différentes formes sont définies, discutées et illustrées par des exemples dans la suite de cette section.

Navigation guidée par des connaissances

Nous appelons navigation guidée par des connaissances dans un treillis de concepts, la navigation avec prise en compte des hiérarchies d'attributs comme connaissances locales. Dans ce cas, la navigation s'effectue normalement et à chaque étape, avant de se déplacer vers un concept destination, on doit s'assurer que ce concept vérifie les dépendances exprimées dans la hiérarchie d'attributs considérée. Dans le cas où un concept destination ne vérifie pas l'une des dépendances considérée, il est ignoré et les liens sortants de ce concept ne sont pas examinés. De cette manière, le passage d'un concept à l'autre lors de la navigation dans le treillis est influencé par les connaissances de domaine. Considérons l'exemple de treillis de concepts donné à la figure 4.1 et la hiérarchie d'attributs donnée à la figure 4.6 (b), représentée par $\mathcal{HA} : Mammals \succ (Complete \sqcup Updated)$. En supposant que la navigation commence à partir du concept le plus général du treillis, les concepts atteignables tout en respectant les dépendances considérées sont montrés sur le treillis donné à la figure 4.8.

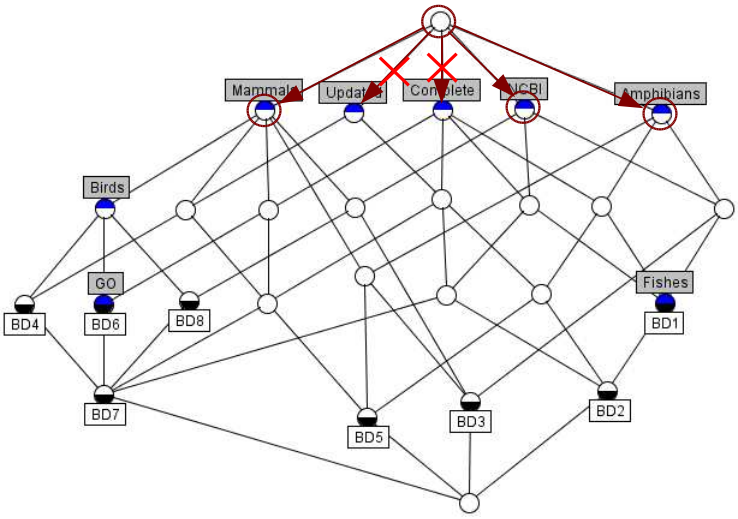


FIGURE 4.8 – Première étape de navigation dans le treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$ guidée par des connaissances exprimées par $\mathcal{HA} : Mammals \succ (Complete \sqcup Updated)$.

Navigation selon un point de vue

Nous appelons navigation dans un treillis de concepts selon un point de vue la navigation avec prise en compte d'une hiérarchie d'attributs (comme connaissances globales). Dans ce cas, le treillis de concepts est réduit conformément à ces connaissances avant de procéder à la navigation. Les connaissances globales formalisent le point de vue considéré. La navigation dans un treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$ selon un point de vue se déroule selon les étapes suivantes :

1. Traduire le point de vue considéré en terme de dépendances entre attributs et le formaliser par une hiérarchie d'attributs $\mathcal{HA}$.

2. Appliquer $\mathcal{HA}$ au treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$ en écartant les concepts non cohérents vis-à-vis de $\mathcal{HA}$. Ces concepts sont incompatibles avec le point de vue considéré. Le treillis des concepts cohérents vis-à-vis de $\mathcal{HA}$, $\underline{\mathfrak{B}}_{\mathcal{HA}}(\mathbb{K})$, constitue une projection du treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$ conformément au point de vue exprimé par $\mathcal{HA}$.
3. Naviguer librement dans $\underline{\mathfrak{B}}_{\mathcal{HA}}(\mathbb{K})$. Au cours de la navigation, aucune vérification de cohérence n'a besoin d'être faite.

Pour illustration, considérons le treillis de concepts donné à la figure 4.1 et considérons le champ *Subject* des sources de données comme critère principal d'exploration. Ceci correspond à un point de vue formalisé par la hiérarchie d'attributs

$$\mathcal{HA}_1 : (Mammals \sqcup Birds \sqcup Amphibiens \sqcup Fishes) \succ (Complete \sqcup Updated \sqcup GO \sqcup NCBI).$$

Les attributs représentant les organismes sont considérés comme attributs principaux et les autres attributs du contexte (table 4.2) sont considérés comme attributs secondaires. Le treillis $\underline{\mathfrak{B}}_{\mathcal{HA}_1}(\mathbb{K})$ obtenu est donné à la figure 4.7. Ce treillis met en évidence la navigation selon le point de vue des attributs relatifs aux champs *Subject* des sources de données. Ces attributs apparaissent au dessus des autres attributs. Chaque concept du treillis $\underline{\mathfrak{B}}_{\mathcal{HA}_1}(\mathbb{K})$ (autre que $\top$ et $\bot$) contient au moins un attribut relatif à *Subject* dans son intension. Ces concepts correspondent à des classes de sources de données relatives aux mêmes thèmes. Les autres attributs dans l'intension d'un concept sont des raffinements de ces classes.

Une fois le treillis $\underline{\mathfrak{B}}_{\mathcal{HA}}(\mathbb{K})$ obtenu, il n'y a pas de limite sur le nombre de navigations à effectuer pour explorer les concepts de ce treillis. Cet aspect constitue l'avantage de cette méthode qui réduit l'espace d'exploration. Par rapport à la navigation dans $\underline{\mathfrak{B}}(\mathbb{K})$, cette méthode permet d'éviter l'exploration de concepts sans intérêt (les concepts non cohérents) et par rapport à la navigation en considérant les connaissances locales, cette méthode évite les tests répétés de cohérence des concepts à considérer à l'étape suivante lors de la navigation.

Interrogation guidée par des connaissances : requêtes avec préférences

De la même manière que la navigation, l'interrogation d'un treillis de concepts avec prise en compte de hiérarchies d'attributs comme connaissances locales conduit à une interrogation guidée par des connaissances. Dans ce cas, les dépendances entre attributs ne sont pas appliquées au treillis de concepts. Elles sont spécifiées pour chaque requête afin de préciser la manière dont les attributs de la requête doivent être interprétés et considérés lors de l'interrogation du treillis de concepts. De cette manière, les hiérarchies d'attributs spécifiées pour une requête peuvent être considérées comme un moyen d'exprimer des préférences sur les attributs de la requête. Chaque dépendance dans ces hiérarchies exprime une préférence entre les attributs dans laquelle les attributs principaux de la dépendance sont préférés aux attributs secondaires. L'interrogation du treillis de concepts par des requêtes avec préférences peut être effectuée de deux manières qu'on appellera méthode naïve et méthode optimisée.

Méthode naïve : Elle consiste à appliquer les étapes suivantes :

1. Formulation d'une requête.
2. Formulation des préférences sur les attributs de la requête.
3. Application de l'algorithme BR-Explorer.
4. Vérification de la cohérence des objets dans la réponse obtenue dans 3 vis-à-vis des préférences exprimées dans 2.

Pour illustrer cette méthode, reprenons l'exemple donné en section 4.4.3 où il s'agit de retrouver les sources de données de BioRegistry contenant des données relatives aux *mammifères* et ayant un contenu *complet* et *à jour*. La formalisation de cette requête donne un concept requête $Q = (X, Y)$ avec $Y = \{Mammals,\ Complete,\ Updated\}$. L'objectif principal de cette requête étant de retrouver les sources de données relatives aux mammifères. De ce fait l'attribut *Mammals* doit être considéré comme attribut principal dans une dépendance entre attributs. Parmi les sources de données relatives aux mammifères, celles qui ont un contenu complet et à jour seront plus intéressantes à considérer. Cela signifie que les attributs *Complete* et *Updated* sont des raffinements dans la description des sources de données à retrouver. Ces attributs sont donc considérés comme attributs secondaires de l'attribut principal *Mammals*. La formalisation de ces préférences donne la hiérarchie d'attributs $\mathcal{HA} : Mammals \succ (Complete \sqcup Updated)$. Après application de BR-Explorer avec la requête $Q = (X, \{Mammals,\ Complete,\ Updated\})$ (les détails sont donnés en section 4.4.3), les préférences exprimées dans $\mathcal{HA}$ sont appliquées au résultat obtenu pour écarter les objets qui ne satisfont pas ces préférences. Ces objets sont BD_1 et BD_2. La réponse finale est la suivante :

1- BD_5 : *Mammals, Complete, Updated*
   BD_7 : *Mammals, Complete, Updated*
2- BD_4 : *Mammals, Updated*
   BD_6 : *Mammals, Complete*
   ~~BD_2~~ : ~~*Complete, Updated*~~ non cohérent vis-à-vis des préférences
3- BD_3 : *Mammals*
   BD_8 : *Mammals*
   ~~BD_1~~ : ~~*Complete*~~ non cohérent vis-à-vis des préférences

Méthode optimisée : Elle consiste à modifier la procédure de recherche de BR-Explorer pour considérer les préférences au fur et à mesure de la recherche des objets pertinents dans le treillis. À chaque étape de la recherche on teste si le concept en cours satisfait les formules de dépendance entre attributs auquel cas on considère les objets dans son extension et on continue la recherche dans ses super-concepts. Dans le cas où le concept en cours ne satisfait pas les préférences, il est inutile de considérer ses super-concepts dans le treillis car ces concepts sont soit incohérents vis-à-vis des préférences soit non pertinents pour la requête. En effet, d'après la définition de la cohérence de concepts, l'intension d'un concept incohérent ne contient que des attributs secondaires sans les attributs principaux dont il sont dépendants. Et comme les subsumants d'un concept C ont des intensions incluses dans l'intension de C, ces intensions ne peuvent pas contenir les attributs principaux absents dans C qui peuvent rétablir la cohérence. Si ces intensions contiennent toujours l'attribut secondaire à l'origine de l'incohérence, alors les concepts sont incohérents. Si ces concepts ne contiennent pas ces attributs alors ils ne partagent rien avec la requête et dans ce cas ils ne sont pas pertinents.

Finalement, la méthode optimisée consiste à appliquer les étapes suivantes :

1. Formulation d'une requête.
2. Formulation des préférences sur les attributs de la requête.
3. Application de l'algorithme BR-Explorer modifié avec test de la cohérence des concepts considérés et élagage de la partie du treillis au dessus de chaque concept non cohérent.

L'illustration de cette méthode sur le même exemple de requête et en considérant les mêmes préférences est donnée à la figure 4.9. Les concepts qui ne satisfont pas les préférences sont ignorés lors de la recherche. Ceci est marqué sur la figure par les liens barrés qui mènent vers ses concepts. Le résultat obtenu à la fin de la recherche est donné ci-après. Il ne contient que des objets qui satisfont les préférences exprimées avec la requête.

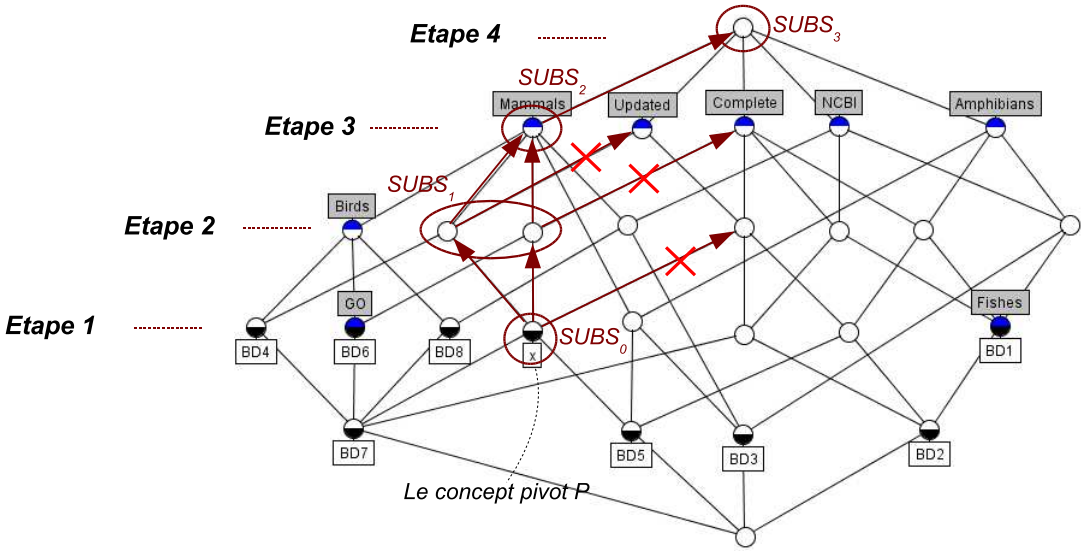


FIGURE 4.9 – Déroulement de l'algorithme BR-Explorer avec prise en compte de préférences entre attributs de la requête $Q = (X, \{Mammals, Complete, Updated\})$. Les préférences sont données par $\mathcal{HA}$: $Mammals \succ (Complete \sqcup Updated)$.

1- BD_5 : *Mammals, Complete, Updated*
   BD_7 : *Mammals, Complete, Updated*
2- BD_4 : *Mammals, Updated*
   BD_6 : *Mammals, Complete*
3- BD_3 : *Mammals*
   BD_8 : *Mammals*

L'optimisation apportée dans cette méthode par rapport à la méthode naïve est la réduction de l'espace de recherche lors de la construction de la réponse à une requête donnée. Cette réduction est induite par l'élagage des concepts incohérents et de leurs super-concepts dans le treillis.

Dans les deux méthodes, naïve et optimisée, la prise en compte de préférences entre attributs permet d'augmenter l'expressivité dans les requêtes. Ces préférences, considérées comme une partie de la requête, donnent la façon dont ces attributs doivent être interprétés et précise que certains attributs sont préférés à d'autres. La formalisation en terme de dépendances entre attributs permet d'exprimer des modalités sur les attributs d'une requête. En effet, la relation "$\succ$" peut être considérée comme un opérateur de degré intermédiaire entre la conjonction et la disjonction. D'une part une dépendance "$m_1 \succ m_2$" est plus forte que la disjonction "$m_1 \sqcup m_2$" car la présence de m_2 seul n'est pas admise. Et d'autre part "$m_1 \succ m_2$" est plus faible que la conjonction "$m_1 \sqcap m_2$" car la présence de m_1 sans m_2 est admise. La table 4.5 donne la table de vérité des trois opérateurs en fonction de la présence des attributs m_1 et m_2.

Interrogation selon un point de vue

De la même manière que la navigation, l'interrogation d'un treillis de concepts selon un point de vue consiste à effectuer l'interrogation en prenant en compte des hiérarchies d'attributs comme connaissances globales. La méthode consiste d'abord à exprimer le point de vue considéré à travers un ensemble de dépendances entre attributs qui sont ensuite regroupées dans une hiérarchie

TABLE 4.5 – Table de vérité des expressions "$m_1 \sqcap m_2$", "$m_1 \sqcup m_2$" et "$m_1 \succ m_2$" en fonction de la présence de m_1 et m_2. L'expression est vraie lorsque la case correspondante contient "×".

	$\{m_1\}$	$\{m_2\}$	$\{m_1, m_2\}$
$m_1 \sqcap m_2$			×
$m_1 \sqcup m_2$	×	×	×
$m_1 \succ m_2$	×		×

d'attributs. Ensuite, le treillis de concepts est réduit à l'ensemble de concepts cohérents vis-à-vis du point de vue exprimé à travers les hiérarchies d'attributs. Enfin, on effectue l'interrogation du treillis réduit en utilisant BR-Explorer. La dernière étape peut être effectuée autant de fois qu'il y a de requêtes à traiter.

Pour illustrer cette méthode, considérons le treillis de concepts représentant les données de BioRegistry donné à la figure 4.1 et considérons le point de vue donné en section 4.6.4 où les attributs relatifs au champ *Subject* constituent le critère principal. La formalisation de ce point de vue sous la forme d'une hiérarchie d'attributs donne la formule suivante :

$$\mathcal{HA}_1 : (Mammals \sqcup Birds \sqcup Amphibiens \sqcup Fishes) \succ (Complete \sqcup Updated \sqcup GO \sqcup NCBI).$$

Cette hiérarchie d'attributs est appliquée au treillis de concepts $\underline{\mathfrak{B}}(\mathbb{K})$ (figure 4.1) pour produire le treillis de concepts, $\underline{\mathfrak{B}}_{\mathcal{HA}_1}(\mathbb{K})$ (figure 4.7), cohérent vis-à-vis du point de vue considéré.

Pour illustrer l'application de BR-Explorer sur le treillis de concepts $\underline{\mathfrak{B}}_{\mathcal{HA}_1}(\mathbb{K})$, considérons à nouveau la requête $Q = (X, \{Mammals, Complete, Updated\})$. Le treillis obtenu suite à l'insertion de cette requête dans $\underline{\mathfrak{B}}_{\mathcal{HA}_1}(\mathbb{K})$ et les étapes de l'exécution de BR-Explorer pour la recherche de sources de données pertinentes sont données à la figure 4.10. Le résultat obtenu est le même que celui obtenu dans le cas de l'interrogation guidée par des connaissances car l'exemple de point de vue considéré ici couvre l'exemple de préférences exprimées sur la requête dans le cas de l'interrogation guidée par des connaissances.

Les requêtes considérées lors de l'interrogation selon un point de vue doivent elles mêmes être cohérentes avec le point de vue considéré. Ceci est le cas de la requête traitée dans l'exemple ci-dessus.

L'avantage principal de cette méthode réside dans la restriction de l'espace de recherche à la partie du treillis de concepts cohérente vis-à-vis du point de vue considéré. Cette méthode est particulièrement intéressante lorsque le treillis initial est de grande taille et que plusieurs interrogations successives relatives à un même point de vue doivent être effectuées.

Dans tous les cas, la prise en compte de connaissances de domaine lors de l'interrogation d'un treillis de concepts permet d'augmenter la précision dans le résultat obtenu.

4.6.5 Connaissances globales ou connaissances locales

La prise en compte de connaissances du domaine lors de l'exploration d'un treillis de concepts a abouti à la définition de plusieurs méthodes de découverte de ressources guidée par des connaissances. Chacune des méthodes proposées dans les sections précédentes est adaptée à un type d'exploration. Ainsi le choix d'une méthode plutôt qu'une autre est effectué selon la manière dont les données dans le treillis de concepts doivent être explorées, selon les types de connaissances dont on dispose et selon l'objectif de l'exploration.

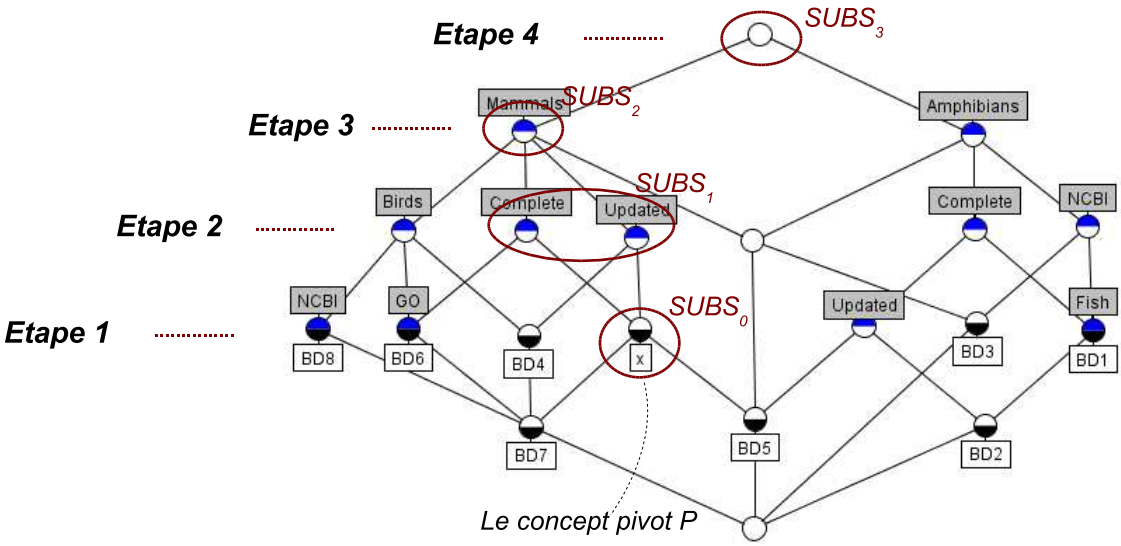


FIGURE 4.10 – Déroulement de l'algorithme BR-Explorer sur le treillis resultant de l'insertion de Q= ($\{x\}$,{*Mammifères, Complet, À jour*}) dans le treillis $\underline{\mathfrak{B}}_{\mathcal{HA}_1}(G, M, I)$.

Chapitre 5

Étude des contextes multivalués

Sommaire

5.1 Introduction 85
5.2 Besoin d'extension aux données complexes 86
5.2.1 Données réelles et contextes multivalués 86
5.2.2 Limites des approches existantes 87
5.2.3 Principe général et étapes de l'approche proposée 88
5.3 Étude des contextes numériques 88
5.3.1 Contextes numériques 88
5.3.2 Similarité entre les valeurs dans un contexte multivalué 89
5.3.3 Représentation équivalente des contextes multivalués 90
5.3.4 Opérateurs de dérivation 90
5.3.5 Ordres partiels 92
5.3.6 Connexion de Galois par similarité 93
5.3.7 Concepts formels multivalués et treillis de concepts multivalués 93
5.3.8 Variation de la précision dans les treillis de concepts multivalués 94
5.3.9 Évolution du treillis de concepts multivalués 96
5.3.10 Treillis extrêmes et liens avec l'échelonnage conceptuel 98
5.4 Études des contextes symboliques 100
5.4.1 Contextes symboliques 100
5.4.2 Similarité entre les valeurs dans un contexte symbolique 100
5.4.3 Ordres Partiels, opérateurs de dérivation et connexion de Galois par similarité 102
5.4.4 Concepts formels multivalués et treillis de concepts multivalués 104
5.4.5 Variation de la granularité des concepts multivalués et évolution du treillis de concepts multivalués 105
5.5 Généralisation aux contextes plurivalués et aux contextes hétérogènes 108
5.5.1 Contexte plurivalué numérique 109
5.5.2 Contexte plurivalué symbolique 110
5.5.3 Contexte hétérogène 112

5.1 Introduction

Lors du processus d'annotation des bases de données dans BioRegistry, les métadonnées d'annotation sont soigneusement choisies à partir de différentes ressources sémantiques : ontologies

du domaine (GO), taxonomies (NCBI), thésaurus (MeSH)) et autres (catégoristion NAR) (section 2.3). De ce fait, plusieurs métadonnées sont souvent liées par des relations sémantiques matérialisées dans les ressources sémantiques. La représentation du contenu de BioRegistry sous la forme d'un contexte formel consiste à considérer chaque métadonnée comme un attribut du contexte qui sera partagé ou non par un certain nombre de sources de données (les objets du contexte). Or dans un contexte formel, les attributs sont indépendants les uns des autres et le seul lien qui puisse exister entre deux attributs est la co-occurrence (lorsque ces attributs sont possédés par un même ensemble d'objets). Ainsi, la représentation du contenu de BioRegistry sous la forme d'un contexte formel ne permet pas la prise en compte directe des relations sémantiques entre les métadonnées d'annotation des sources de données. De cette manière, l'une des caractéristiques de BioRegistry, l'annotation sémantique des sources de données, n'est pas pris en compte dans la classification en treillis. Dans le processus de découverte de ressources par treillis détaillé dans le chapitre précédant, nous avons proposé de considérer les relations sémantiques après la phase de classification en treillis de concepts avec l'enrichissement sémantique de requête et les dépendances entre attributs. Dans ce chapitre nous proposons une approche originale qui étend l'ACF aux données complexes en exploitant les relations sémantiques. Cette approche permet d'atteindre un double objectif : d'une part éviter la transformation des données en considérant des contextes multivalués mieux adaptés pour garder les données complexes sous leur format d'origine et d'autre part exploiter la sémantique de ces données pour les classer en treillis de concepts. De cette manière il devient possible de combiner les caractéristiques sémantiques de BioRegistry et celles de la classification en treillis de concepts.

La suite du chapitre est organisée comme suit. La section 5.2 montre brièvement la nécessité de définir de nouvelles approches pour exploiter les données complexes et l'assise formelle de l'ACF. Les sections 5.3, 5.4 et 5.5 contiennent la formalisation de notre approche d'analyse de données complexes.

5.2 Besoin d'extension aux données complexes

5.2.1 Données réelles et contextes multivalués

Comme illustré au chapitre 4 et en section 3.4, l'ACF offre un cadre formel pour le développement d'applications efficaces d'analyse de données. De plus les structures conceptuelles de l'ACF, telles que les treillis de concepts, peuvent être exploitées à la fois comme outils de classification, support de découverte de ressources, de fouille de données, etc. Ces caractéristiques sont à l'origine du succès des approches d'analyse, de représentation et d'exploitation de données qui s'appuient sur l'ACF. Cependant, avant de profiter de la classification en treillis de concepts, il est indispensable de ramener les données manipulées, souvent complexes et hétérogènes, à un contexte formel. Dans les applications réelles les attributs décrivant les objets prennent différentes valeurs pour les différents objets et parfois même plusieurs valeurs pour un même objet. Dans le cas de BioRegistry, une représentation directe du modèle des métadonnées aboutit à un ensemble d'objets (les sources de données biologiques), un ensemble d'attributs décrivant ces objets (les sections et sous sections dans le modèle de métadonnées de BioRegistry e.g. *subjects*, *organisms*, etc.) et des valeurs prises par chaque attribut pour chaque objet (e.g. *proteins*, *Human*) (voir table 5.6 et table 5.9). Le contenu de BioRegistry se présente ainsi sous la forme d'un contexte multivalué (définition 18) ou de manière plus générale sous la forme d'un contexte plurivalué où un attribut peut avoir plusieurs valeurs pour un même objet (définition 51).

Les contextes multivalués assurent une présentation directe des données complexes dans le cadre des applications réelles de l'ACF. Par exemple, dans le cas de la recherche documentaire,

une représentation plus précise d'un corpus de recherche est donnée par un contexte multivalué où les objets sont des documents, les attributs sont des termes d'indexation et les valeurs dans le contexte sont les poids des termes d'indexation pour chaque document. Dans le cas de l'extraction de connaissances à partir de bases de données, le recours à des contextes multivalués allège l'étape de préparation de données. En effet, une relation dans une base de données relationnelle peut être vue comme un contexte multivalué où les instances sont les objets, les attributs de la relation sont les attributs (du contexte) et les valeurs des attributs dans chaque tuple sont les valeurs des attributs du contexte.

5.2.2 Limites des approches existantes

Echelonnage conceptuel

Face à des données sous la forme d'un contexte multivalué, le recours à un échelonnage conceptuel est systématique dans les approches qui s'appuient sur l'ACF. Il est simplement question du choix de l'échelonnage le mieux adapté aux données considérées selon une interprétation préalable. Cependant, la transformation en un contexte formel entraîne souvent des pertes au niveau de la précision des données (par exemple dans le cas du contexte des planètes du système solaire Table 5.1 et Table 3.1) ainsi qu'au niveau des relations entre ces données (par exemple dans le cas de BioRegistry Table 4.1 et Table 4.2). Une fois qu'un échelonnage conceptuel est effectué, le contexte formel obtenu est ensuite utilisé comme entrée du processus d'analyse de données. De cette manière, le traitement s'effectue sur une abstraction des données (le contexte formel obtenu) au lieu des données réelles (le contexte multivalué). En conséquence, les résultats obtenus dépendent fortement de l'échelonnage et peuvent être sensibles aux données réelles d'origine. Tout cela s'ajoute à la difficulté de l'automatisation des échelonnages conceptuels.

Prise en compte des connaissances du domaine

Une manière de pallier le manque de relations sémantiques explicites entre les attributs dans un contexte formel consiste à considérer des connaissances du domaine afin d'établir ces relations sémantiques. La première façon est celle proposée dans [Carpineto and Romano, 2004a] et décrite en section 3.6.3, consistant à modifier le contexte de manière à ce que les relations entre les attributs soient visibles sur le treillis de concepts. Cette approche est intéressante dans la mesure où elle prépare le treillis de concepts à une exploration conforme aux relations hiérarchiques entre les attributs dans la ressource sémantique considérée. Cependant, la modification du contexte l'écarte encore plus des données initiales et aboutit à des données redondantes et bruitées (en ajoutant des relations objet×attributs initialement absentes du contexte). La deuxième façon consiste à considérer les connaissances du domaine lors de la phase d'exploitation des données (interrogation, navigation, etc.) comme l'enrichissement de requêtes et les hiérarchies d'attributs détaillés pour BR-Explorer. Comme illustré dans le chapitre précédent, cette manière de considérer les relations sémantiques au moment de l'exploitation d'un treillis de concepts permet d'améliorer les performances du processus d'exploitation mais reste tout de même dépendante d'un échelonnage conceptuel à l'origine des données manipulées.

Les extensions de l'ACF

Les extensions de l'ACF décrites à la section 3.6 proposent des approches évitant l'étape de transformation de données en contextes formels. Chacune des approches proposées est adaptée à un format particulier des données contenues dans un contexte multivalué (e.g. données floues

dans le cas de l'ACFF, données logiques dans le cas de l'ACL, etc.). Cet aspect leur permet de bénéficier des théories et propriétés relatives à chacun de ces formats pour définir une extension de l'ACF. En effet, l'ACL repose sur les propriétés des objets logiques alors que l'ACFF repose sur la théorie des ensembles flous et les treillis résidués. Bien que ces approches apportent des solutions intéressantes pour l'analyse de données complexes, elles n'offrent pas une solution générale traitant les données hétérogènes telles que les ressources du web ou plus particulièrement les ressources biologiques de BioRegistry.

5.2.3 Principe général et étapes de l'approche proposée

Dans la suite de ce chapitre nous proposons une approche originale pour l'analyse de données complexes en exploitant les relations sémantiques entre ces données. Les données se présentent sous la forme d'un contexte multivalué arbitraire (sans restrictions sur le type des valeurs dans le contexte). L'idée principale consiste à considérer les relations sémantiques entre les données comme critère de classification en treillis de concepts. Il s'agit de généraliser l'ACF comme suit. Par analogie avec l'ACF où les objets du contexte formel qui ont un même (ensemble d') attribut(s) sont regroupés, nous regroupons les objets du contexte multivalué qui ont des valeurs similaires pour un même (ensemble d') attribut(s). La similarité entre les valeurs des attributs est un paramètre générique qui peut être instancié selon la particularité de chaque attribut sans contraindre l'approche. En s'appuyant sur ce principe, nous définissons la connexion de Galois par similarité qui exploite la sémantique des données pour dériver les concepts et le treillis de concepts à partir du contexte multivalué.

Pour simplifier la formalisation de cette approche, nous considérons d'abord des contextes multivalués où les données sont homogènes. Nous considérons dans un premier temps les données numériques pour la simplicité du calcul de la similarité entre elles et dans un deuxième temps les données symboliques avec des illustrations sur des exemples de BioRegistry. Ensuite nous étendons la formalisation aux données hétérogènes et aux contextes plurivalués.

5.3 Étude des contextes numériques

5.3.1 Contextes numériques

Un contexte numérique est un contexte multivalué dont les valeurs sont des nombres réels. La particularité des contextes numériques est qu'il est possible d'établir un ordre total entre les valeurs distinctes dans le contexte. Cet aspect est à l'origine de l'extension de l'ACF que nous proposons ici. La comparaison de deux valeurs d'attributs permet de juger de leur similarité. En procédant par analogie avec l'ACF où il s'agit de grouper les objets qui ont des attributs en commun, nous pouvons effectuer des groupements d'objets ayant des valeurs similaires pour certains attributs. De cette manière, nous allégeons la contrainte relative au partage d'attributs entre objets.

La table 5.1 donne un exemple de contexte numérique représentant les mesures réelles des caractéristiques des planètes du système solaire [29].

29. Mesures prise à partir des sites suivants :
http ://ssd.jpl.nasa.gov/ ?planet_phys_par
http ://www.planetobs.com/index.php

TABLE 5.1 – Mesures réelles des caractéristiques des planètes du système solaire

	Diamètre (km)	Distance au Soleil (10^6km)	Masse (10^{23}kg)	Satellite
Mercure	4 879	58	3.30	
Vénus	12 104	108	48.69	
Terre	12 756	150	59.74	1
Mars	6 794	228	6.42	2
Jupiter	142 984	778	18 988	16
Saturn	120 536	1 427	5 685	19
Uranus	51 118	2 870	866.25	5
Neptune	49 528	4 500	1027.8	8
Pluton	2 302	5 950	0.13	1

5.3.2 Similarité entre les valeurs dans un contexte multivalué

La définition intuitive de la similarité est la suivante : deux valeurs d'un même attributs sont similaires s'il n'y a pas une grande différence entre elle. La différence maximale permise entre deux valeurs similaires peut varier d'un attribut à l'autre et d'un contexte à l'autre. Par exemple, dans le cas du contexte multivalué donné en table 5.1, deux planètes ont des mesures de diamètre similaires si leur différence est de l'ordre de quelques centaines de kilomètres. Étant donné cette hypothèse, on peut dire que les planètes *Terre* et *Vénus* ont des diamètres similaires (12 104 et 12 756 respectivement). Si on élargit la différence en considérant que deux diamètres sont similaires si leur différence est de l'ordre de quelques milliers de kilomètres, on peut aussi dire que les planètes *Mars*, *Mercure* et *Pluton* ont des diamètres similaires (6 794, 4 879 et 2 302 respectivement).

La formalisation de cette définition consiste à choisir la différence maximale permise entre deux valeurs similaires. Dans l'exemple des valeurs de diamètre des planètes, la différence maximale dans le premier cas (quelques centaines) pourrait être 1 000 kilomètres alors que dans le deuxième cas (quelques milliers) cette différence pourrait être 10 000 kilomètres.

Définition 34 (Seuil de variation, valeurs similaires) *Soit un contexte formel multivalué* (G, M, W, I).

1. *On appelle seuil de variation des valeurs d'un attribut* $m \in M$ *dans un contexte multivalué* (G, M, W, I) *la différence maximale entre deux valeurs similaires de cet attribut. On note par* θ_m *ce seuil.*
2. *On dit que deux valeurs* w_i *et* w_j *d'un même attribut* $m \in M$ *sont similaires si et seulement si leur différence est inférieure au seuil de variation* θ_m *i.e.* $|w_i - w_j| \leq \theta_m$.

Le choix d'un seuil de variation θ_m pour un attribut m permet de juger si deux objets ont des valeurs similaires pour un attribut. Par exemple le seuil de variation des valeurs de l'attribut *Diamètre*, $\theta_D = 1\ 000$, permet de déclarer que les planètes *Terre* et *Vénus* ont des diamètres de valeurs similaires. Si on veut former des classes de planètes selon leurs dimensions (diamètres), ces deux planètes peuvent faire partie d'une même classe. En terme de partage d'attribut, on peut ainsi dire que les planètes *Terre* et *Vénus* partagent l'attribut diamètre pour une certaine plage de valeurs. Ceci nous amène à la définition du partage d'attributs entre objets dans un contexte multivalué.

Définition 35 (Partage d'attributs, intervalle de similarité) *Soit un contexte formel multivalué* (G, M, W, I).

1. *Deux objets* g_i *et* g_j ***partagent*** *un attribut* $m \in M$ *si et seulement si les valeurs* $m(g_i) = w_i$ *et* $m(g_j) = w_j$ *sont similaires i.e.* $|w_i - w_j| \leq \theta_m$. g_i *et* g_j *partagent l'attribut* m *pour la plage de valeurs* $[w_i, w_j]$ *noté par* $m_{[w_i,w_j]}$ *(en supposant que* $w_i \leq w_j$*). L'intervalle* $[w_i, w_j]$ *est appelé* ***intervalle de similarité*** *de* m *pour les objets* g_i *et* g_j.
2. *Un ensemble d'objets* $A \subseteq G$ *partage un attribut* $m \in M$ *si et seulement si toute paire d'objets de* A *partage* m. *L'intervalle de similarité de* m *pour* A *est* $[min_{g\in A}(m(g)), max_{g\in A}(m(g))]$. *Les objets de* A *partagent l'attribut* $m_{[min_{g\in A}(m(g)),max_{g\in A}(m(g))]}$.

Pour l'attribut *Diamètre* dans le contexte multivalué des planètes du système solaire (table 5.1) et pour le seuil de variation $\theta_D = 1\,000$, les planètes *Terre* et *Vénus* partagent $Diamètre_{[12104,\ 12756]}$. Pour $\theta_D = 10\ 000$, les planètes *Mars*, *Mercure* et *Pluton* partagent l'attribut $Diamètre_{[2302,\ 6794]}$

De la même manière, il est possible de choisir des seuils de variation pour tous les attributs du contexte multivalué. Le choix d'un seuil de variation pour un attribut dépend de l'interprétation de l'attribut et de son domaine de valeurs. Par exemple, lors du choix d'un seuil de variation pour l'attribut *Distance au Soleil*, on doit prendre en compte le fait que les valeurs prises par cet attribut sont exprimées en millions de millions de kilomètres. Pour cet attribut, le choix d'un seuil maximal de variation de quelques kilomètres ($\theta_{DS} = 10km$) n'aboutit à aucun partage de cet attribut par les objets du contexte. Par contre, le choix d'un seuil maximal de variation de quelques dizaines de millions de kilomètres ($\theta_{DS} = 100\ 10^6 km$) est plus pertinent et permet de regrouper les planètes *Mercure* et *Vénus* (partagent $Distance\ au\ Soleil_{[58,\ 108]}$), *Vénus* et *Terre* (partagent $Distance\ au\ Soleil_{[108,\ 150]}$) et *Terre* et *Jupiter* (partagent $Distance\ au\ Soleil_{[150,\ 228]}$).

5.3.3 Représentation équivalente des contextes multivalués

Pour faciliter la compréhension, nous choisissons un même seuil de variation, noté θ, pour tous les attributs du contexte multivalué (G, M, W, I). Pour que cela soit possible dans le cas des contextes dont les domaines des valeurs des attributs sont très variés, nous considérons une représentation équivalente du contexte qui consiste à normaliser en ramenant ces valeurs dans l'intervalle [0,1]. La transformation consiste à diviser chaque valeur d'un attribut par la valeur maximale prise par cet attribut dans le contexte. Il est à noter que la transformation se fait sans perte d'information et qu'il est possible de retrouver les valeurs initiales par de simples opérations de multiplication.

La transformation du contexte multivalué représentant les planètes du système solaire donné en table 5.1 produit le contexte multivalué donné en table 5.2.

Les mesures relatives aux planètes *Jupiter* et *Saturne* (Diamètre et Masse) sont nettement supérieures à celles des autres planètes. De ce fait la répartition des valeurs du contexte sur l'intervalle [0,1] n'est pas homogène et plusieurs valeurs sont très proche de 0. Cet aspect illustre mal l'approche que nous proposons. Un meilleur exemple consiste à considérer uniquement la partie de ce contexte formée par les quatre premières planètes (Mercure, Vénus, Terre, Mars). Le contexte multivalué correspondant à cette partie après normalisation des valeurs dans l'intervalle [0,1] est donné en table 5.3. Cet exemple sera considéré dans la suite.

5.3.4 Opérateurs de dérivation

Définition 36 ("$\Im_\theta$") *Soit un seuil de variation* $\theta \in [0, 1]$. *Les intervalles de variation des valeurs des attributs communs à un ensemble d'objets donné sont de la forme* $[\alpha, \beta]$ *tels que* $\beta - \alpha \leq \theta$. *L'ensemble de tous les intervalles possibles de cette forme est noté par* $\Im_\theta$.

TABLE 5.2 – Tableau des planètes du système solaire

	Diamètre	Distance au Soleil	Masse	Satellite
Mercure	0.0341	0.0097	1.739 e-4	
Vénus	0.0846	0.018	0.00256	
Terre	0.0892	0.025	0.00314	0.0526
Mars	0.0475	0.038	3.380 e-4	0.105
Jupiter	1	0.130	1	0.842
Saturne	0.843	0.239	0.299	1
Uranus	0.357	0.482	0.0456	0.263
Neptune	0.346	0.756	0.0541	0.421
Pluton	0.016	1	6.920 e-6	0.0526

TABLE 5.3 – Tableau des quatres premières planètes du système solaire.

	Diamètre	Distance au Soleil	Masse	Satellite
Mercure	0.38	0.25	0.05	
Vénus	0.94	0.47	0.81	
Terre	1	0.65	1	0.5
Mars	0.53	1	0.1	1

Dans la suite on notera par $\mathfrak{P}(G)$ (respectivement $\mathfrak{P}(M)$) l'ensemble des parties de G (respectivement M) et par $\mathfrak{P}(M \times \mathfrak{I}_\theta)$ tous les ensembles possibles d'attributs pondérés par leurs intervalles de variation étant donné un seuil de variation θ. Les éléments de $M \times \mathfrak{I}_\theta$ sont des paires de la forme $(m, [\alpha, \beta])$ où $m \in M$ et $[\alpha, \beta] \in \mathfrak{I}_\theta$ (i.e. $\beta - \alpha \leq \theta$). Dans la suite, on notera la paire $(m, [\alpha, \beta]) \in M \times \mathfrak{I}_\theta$ par $m_{[\alpha,\beta]}$.

En s'appuyant sur la définition de partage d'attributs entre les objets dans un contexte multivalué (définition 35), on définit les opérateurs de dérivation entre $\mathfrak{P}(G)$ et $\mathfrak{P}(M \times \mathfrak{I}_\theta)$ comme suit.

Définition 37 *Soit un contexte formel multivalué (G, M, W, I). Pour tout ensemble d'objets $A \in \mathfrak{P}(G)$,*

$$A^{\uparrow_\theta} = \{m_{[\alpha,\beta]} \in M \times \mathfrak{I}_\theta \text{ tel que } \forall g_i, g_j \in A,\ |m(g_i) - m(g_j)| \leq \theta,\ \alpha = min_{g \in A}(m(g)) \text{ et } \beta = max_{g \in A}(m(g))\}$$

est l'ensemble d'attributs dans $M \times \mathfrak{I}_\theta$ communs (au sens de la définition 35) à tous les objets de A.

De façon duale, Pour tout ensemble d'attributs $B \in \mathfrak{P}(M \times \mathfrak{I}_\theta)$,

$$B^{\downarrow_\theta} = \{g \in G \text{ tel que } \forall m_{[\alpha,\beta]} \in B, m(g) \in [\alpha, \beta]\}$$

est l'ensemble d'objets dans G qui partagent tous les attributs dans B.

Dans l'exemple de contexte multivalué donné en table 5.3 et pour un seuil $\theta = 0.2$, on a :

$$\{\textit{Vénus, Terre}\}^{\uparrow_\theta} = \{D_{[0.94,1]},\ DS_{[0.47,0.65]},\ M_{[0.81,1]}\}$$

et

$$\{D_{[0.94,1]},\ DS_{[0.47,0.65]},\ M_{[0.81,1]}\}^{\downarrow_\theta} = \{Vénus,\ Terre\}.$$

Dans le cas où l'intervalle de similarité $[\alpha, \beta]$ d'un attribut m est réduit à une seule valeur : $\alpha = \beta$, on le note par m_α au lieu de $m_{[\alpha,\alpha]}$. Ceci est le cas des attributs dans l'image par $^{\uparrow_\theta}$ de chacun des objets du contexte considéré séparément. Par exemple, dans le contexte numérique donné en table 5.3 :

$$\{Mercure\}^{\uparrow_\theta} = \{D_{0.38},\ DS_{0.25},\ M_{0.05}\}.$$

5.3.5 Ordres partiels

Il est possible de définir plusieurs relations d'ordre partiel sur $\mathfrak{P}(G)$ et sur $\mathfrak{P}(M \times \mathfrak{I}_\theta)$. Une relation d'ordre partiel possible sur $\mathfrak{P}(G)$ est l'inclusion "$\subseteq$" entre ensembles comme dans le cas de l'ACF. Par exemple $\{Mercure,\ Vénus\} \subseteq \{Mercure,\ Vénus,\ Terre\}$. $(\mathfrak{P}(G), \subseteq)$ est ainsi un ensemble partiellement ordonné.

Une relation d'ordre partiel sur $\mathfrak{P}(M \times \mathfrak{I}_\theta)$ qui s'appuie sur "$\subseteq$" et prend en compte les intervalles de variation des attributs étant donné un seuil de variation θ est notée par "$\subseteq_\theta$" et définie comme suit.

Définition 38 ("$\subseteq_\theta$") *Soient B_1 et B_2 deux ensembles d'attributs dans $\mathfrak{P}(M \times \mathfrak{I}_\theta)$. $B_1 \subseteq_\theta B_2$ si et seulement si tout attribut contenu dans B_1 est aussi contenu dans B_2 avec un intervalle de variation plus réduit. Formellement, $B_1 \subseteq_\theta B_2$ si et seulement si $\forall\ m_{[\alpha_1,\beta_1]} \in B_1$, $\exists\ m_{[\alpha_2,\beta_2]} \in B_2$ tel que $[\alpha_2, \beta_2] \subseteq [\alpha_1, \beta_1]$.*

Dans l'exemple de contexte multivalué donné en table 5.3 et pour un seuil de variation $\theta = 0.7$ on a :

$$\{D_{[0.38,1]},\ DS_{[0.25,0.65]}\} \subseteq_\theta \{D_{[0.94,1]},\ DS_{[0.47,0.65]},\ M_{[0.81,1]}\}$$

Proposition 4 *La relation "$\subseteq_\theta$" est une relation d'ordre partiel sur l'ensemble $\mathfrak{P}(M \times \mathfrak{I}_\theta)$.*

Preuve 4 *Pour que la relation "$\subseteq_\theta$" soit une relation d'ordre partiel il faut qu'elle soit reflexive, anti-symétrique et transitive.*

- *"$\subseteq_\theta$" est reflexive car $\forall\ B \in \mathfrak{P}(M \times \mathfrak{I}_\theta)$, $B \subseteq_\theta B$*
- *"$\subseteq_\theta$" est anti-symétrique car pour B_1, $B_2 \in \mathfrak{P}(M \times \mathfrak{I}_\theta)$ tels que $B_1 \subseteq_\theta B_2$ et $B_1 \neq B_2$, deux cas sont possibles : (1) soit il existe $m \in B_2$ tel que $m \notin B_1$ auquel cas $B_2 \not\subseteq_\theta B_1$ (2) soit tous les attributs de B_2 sont aussi dans B_1 et il existe $m_{[\alpha_1,\beta_1]} \in B_1$ et $m_{[\alpha_2,\beta_2]} \in B_2$ tel que $[\alpha_2, \beta_2] \subset [\alpha_1, \beta_1]$. Dans ce cas $m_{[\alpha_2,\beta_2]} \notin B_1$ ($[\alpha_1, \beta_1] \not\subseteq [\alpha_2, \beta_2]$) et par suite $B_2 \not\subseteq_\theta B_1$.*
- *"$\subseteq_\theta$" est transitive car pour B_1, B_2 et $B_3 \in \mathfrak{P}(M \times \mathfrak{I}_\theta)$ tels que $B_1 \subseteq_\theta B_2$ et $B_2 \subseteq_\theta B_3$, on a $\forall m_{[\alpha_1,\beta_1]} \in B_1$, $\exists m_{[\alpha_2,\beta_2]} \in B_2$ tel que $[\alpha_2, \beta_2] \subseteq [\alpha_1, \beta_1]$ (d'après $B_1 \subseteq_\theta B_2$) et $\forall\ m_{[\alpha_2,\beta_2]} \in B_2$, $\exists m_{[\alpha_3,\beta_3]} \in B_3 \mid [\alpha_3, \beta_3] \subseteq [\alpha_2, \beta_2]$ (d'après $B_2 \subseteq_\theta B_3$) par suite on peut déduire que $\forall\ m_{[\alpha_1,\beta_1]} \in B_1$, $\exists\ m_{[\alpha_3,\beta_3]} \in B_3 \mid [\alpha_3, \beta_3] \subseteq [\alpha_1, \beta_1]$ et donc que $B_1 \subseteq_\theta B_3$.*

Remarque 1 *La relation d'ordre partiel "$\subseteq_\theta$" définie sur $\mathfrak{P}(M \times \mathfrak{I}_\theta)$ peut être vue comme le produit de deux relations d'ordre partiel : l'inclusion $\subseteq$ entre ensembles définie sur M (éléments de $\mathfrak{P}(M)$) et l'inclusion entre intervalles de longueur inférieure ou égale à θ définis sur [0,1] (éléments de $\mathfrak{I}_\theta$).*

5.3.6 Connexion de Galois par similarité

Théorème 4 (Connexion de Galois par similarité) *Les opérateurs de dérivation $^{\uparrow_\theta}$ et $^{\downarrow_\theta}$ forment une connexion de Galois (qu'on appellera connexion de Galois par similarité) entre les ensembles partiellement ordonnés $(\mathfrak{P}(G), \subseteq)$ et $(\mathfrak{P}(M \times \mathfrak{I}_\theta), \subseteq_\theta)$.*

Preuve 5 *D'après la définition de la connexion de Galois donnée dans [Ganter and Wille, 1999], on a : $^{\uparrow_\theta}$ et $^{\downarrow_\theta}$ forment une connexion de Galois entre $(\mathfrak{P}(G), \subseteq)$ et $(\mathfrak{P}(M \times \mathfrak{I}_\theta), \subseteq_\theta)$ si et seulement les trois conditions suivantes sont vérifiées pour tous $A, A_1, A_2 \in \mathfrak{P}(G)$ et $B, B_1, B_2 \in \mathfrak{P}(M \times \mathfrak{I}_\theta)$:*

1. $A_1 \subseteq A_2 \Rightarrow A_2^{\uparrow_\theta} \subseteq_\theta A_1^{\uparrow_\theta}$
2. $B_1 \subseteq_\theta B_2 \Rightarrow B_2^{\downarrow_\theta} \subseteq B_1^{\downarrow_\theta}$
3. $A \subseteq (A^{\uparrow_\theta})^{\downarrow_\theta}$ *et* $B \subseteq_\theta (B^{\downarrow_\theta})^{\uparrow_\theta}$

La démonstration est analogue à celle du cas de l'ACF classique.

1. *Si* $m_{[\alpha,\beta]} \in A_2^{\uparrow_\theta}$ *alors* $\forall g \in A_2$, $m(g) \in [\alpha, \beta]$ *en particulier* $\forall g \in A_1$ *et par suite* $m_{[\alpha,\beta]} \in A_1^{\uparrow_\theta}$.
2. *Si* $g \in B_2^{\downarrow_\theta}$ *alors* $\forall m \in B_2$, $m(g) \in [\alpha, \beta]$ *en particulier* $\forall m \in B_1$ *et par suite* $g \in B_1^{\downarrow_\theta}$.
3. *Si* $g \in A$ *alors* $m(g) \in [\alpha, \beta]$ $\forall m \in A^{\uparrow_\theta}$. $m(g) \in [\alpha, \beta]$ *pour* $m \in A^{\uparrow_\theta}$ *implique que* $g \in (A^{\uparrow_\theta})^{\downarrow_\theta}$ *d'où* $A \subseteq (A^{\uparrow_\theta})^{\downarrow_\theta}$.

5.3.7 Concepts formels multivalués et treillis de concepts multivalués

Définition 39 (Concept formel multivalué) *Un concept formel multivalué est un couple (A, B) où A est un ensemble d'objets et B est un ensemble d'attributs avec intervalles de variation tels que $A^{\uparrow_\theta} = B$ et $B^{\downarrow_\theta} = A$. A et B sont respectivement l'extension et l'intension du concept formel multivalué (A, B).*

Dans l'exemple de contexte multivalué donné en table 5.3 et pour un seuil $\theta = 0.2$, on a :

$$(\{\textit{Vénus, Terre}\},\ \{D_{[0.94,1]},\ DS_{[0.47,0.65]},\ M_{[0.81,1]}\})$$

est un exemple de concept formel multivalué.

Définition 40 (Super-concept, sous-concept, $\leq_\theta$) *Soient (A_1, B_1) et (A_2, B_2) deux concepts formels multivalués. (A_1, B_1) est dit **sous-concept** de (A_2, B_2) si $A_1 \subseteq A_2$ (ou de façon équivalente $(B_2 \subseteq_\theta B_1)$. Dans ce cas, (A_2, B_2) est dit **super-concept** de (A_1, B_1) et on note $(A_1, B_1) \leq_\theta (A_2, B_2)$. La relation "$\leq_\theta$" est dite l'ordre hiérarchique des concepts multivalués.*

Dans l'exemple de contexte multivalué donné en table 5.3 et pour un seuil $\theta = 0.2$, on a :

$$(\{\textit{Terre}\},\ \{D_1,\ DS_{0.65},\ M_1,\ S_{0.5}\}) \leq_\theta (\{\textit{Vénus, Terre}\},\ \{D_{[0.94,1]},\ DS_{[0.47,0.65]},\ M_{[0.81,1]}\})$$

Définition 41 (Treillis de concepts multivalués) *L'ensemble de concepts multivalués d'un contexte formel multivalué (G, M, W, I), noté par $\mathfrak{B}_\theta(G, M, W, I)$, muni de la relation d'ordre hiérarchique "$\leq_\theta$", $(\mathfrak{B}_\theta(G, M, W, I), \leq_\theta)$, forme le **treillis de concepts multivalués**, noté par $\underline{\mathfrak{B}}_\theta(G, M, W, I)$, correspondant au contexte multivalué $G, M, W, I)$ pour un seuil de similarité θ.*

Le treillis de concepts multivalués correspondant au contexte multivalué donné en table 5.3 pour un seuil de variation $\theta = 0.2$, $\underline{\mathfrak{B}}_{0.2}(G, M, W, I)$, est donné dans la figure 5.1.

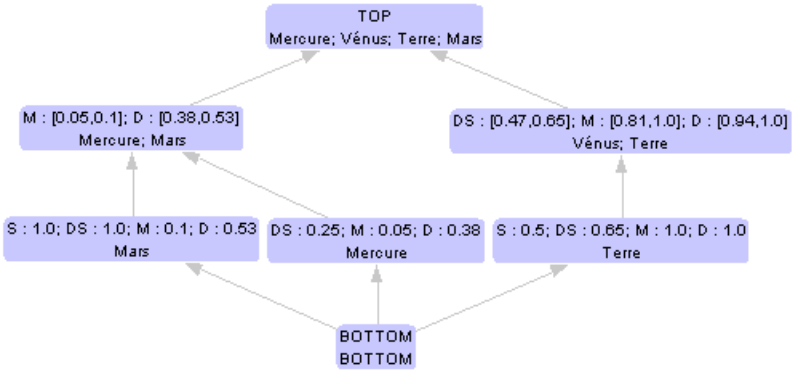


FIGURE 5.1 – Le treillis de concepts multivalués $\underline{\mathfrak{B}}_{0.2}(G, M, W, I)$ correspondant au contexte formel multivalué (G, M, W, I) donné en table 5.3.

5.3.8 Variation de la précision dans les treillis de concepts multivalués

La variation du seuil θ peut entraîner l'apparition et/ou la disparition de concepts dans le treillis de concepts multivalués obtenu à partir d'un contexte numérique. En effet une telle variation correspond à un changement dans la contrainte à vérifier lors du calcul des attributs partagés par un ensemble d'objets du contexte. L'augmentation du seuil θ correspond à un relâchement de la contrainte ce qui peut entraîner plus de partage d'attributs entre les objets. Inversement, la diminution du seuil de variation correspond à un renforcement de la contrainte ce qui entraîne moins de partage d'attributs entre les objets du contexte. Par exemple, dans le contexte numérique donné à la table 5.3, pour un seuil $\theta = 0.3$, les objets *Mercure* et *Vénus* partagent l'attribut $DS_{[0.25,0.47]}$ et les objets *Vénus* et *Terre* partagent l'attribut $DS_{[0.47,0.65]}$. Si on diminue le seuil à $\theta = 0.1$, l'attribut DS ne sera plus partagé par ces paires paires d'objets. Par contre, si on augmente le seuil maximal de variation à $\theta = 0.4$, les deux sous-ensembles d'objets seront fusionnés pour former un ensemble plus grand d'objets partageant l'attribut $DS_{[0.25,0.65]}$. Ainsi la variation du seuil de variation θ induit une modification des concepts multivalués obtenus. Ceci nous conduit à la définition suivante.

Définition 42 (Granularité d'un concept formel multivalué) *Soit un context formel multivalué (G, M, W, I) et soient deux concepts multivalués (A_1, B_1) et (A_2, B_2) obtenus à partir de (G, M, W, I) pour deux seuils de variation différents tels que leurs intensions B_1 et B_2 ont des attributs en commun si on ignore les intervalles de similarité. On dit que (A_1, B_1) est* ***de granularité plus fine*** *que (A_2, B_2) (auquel cas (A_2, B_2) est* ***de granularité plus large*** *que (A_1, B_1)) si les intervalles de similarités des attributs dans B_1 sont plus petits que ceux dans B_2. Dans ce cas (A_1, B_1) est dit* ***plus précis*** *que (A_2, B_2) ((A_2, B_2) est dit* ***plus général*** *que (A_1, B_1)).*

Dans le contexte numérique donné à la table 5.3, le concept multivalué ({*Mercure, Mars*}, $\{D_{[0.38,0.53]}, M_{[0.05,0.1]}\}$) obtenu pour un seuil de variation $\theta = 0.2$ est de granularité plus fine que le concept multivalué ({*Mercure, Mars, Vénus*}, $\{D_{[0.38,0.94]}, DS_{[0.25,1]}, M_{[0.05,0.81]}\}$) obtenu pour un seuil de variation $\theta = 0.8$.

L'augmentation de la granularité des concepts dans un treillis de concepts multivalués faisant suite à l'augmentation du seuil θ produit de nouveaux groupements d'objets dus aux relâchement de la contrainte de partage des attributs. De manière globale, cette augmentation peut se traduire par les trois effets suivants dans le treillis de concepts multivalués.

– L'augmentation de la granularité des concepts existants et l'apparition de nouveaux liens entre ces concepts. Les liens sont établis avec des sous concepts dont les objets en extensions viennent d'être ajoutés aux concepts de granularités plus larges. Dans l'exemple de contexte multivalué (G, M, W, I) donné à la table 5.3, cet effet est illustré, lors du passage du treillis de concepts multivalués obtenu pour un seuil maximal $\theta = 0.3$ à celui obtenu pour un seuil maximal $\theta = 0.5$, par le concept $(\{Mercure,\ Vénus\}, \{DS_{[0.25,0.47]}\})$ transformé en $(\{Mercure,\ Terre,\ Vénus\}, \{DS_{[0.25,0.65]}\})$ avec ajout du lien vers le sous-concept $(\{Vénus,\ Terre\}, \{D_{[0.94,1]},\ DS_{[0.47,0.65]},\ M_{[0.81,1]}\})$. Les deux treillis sont donnés à la figure 5.2.

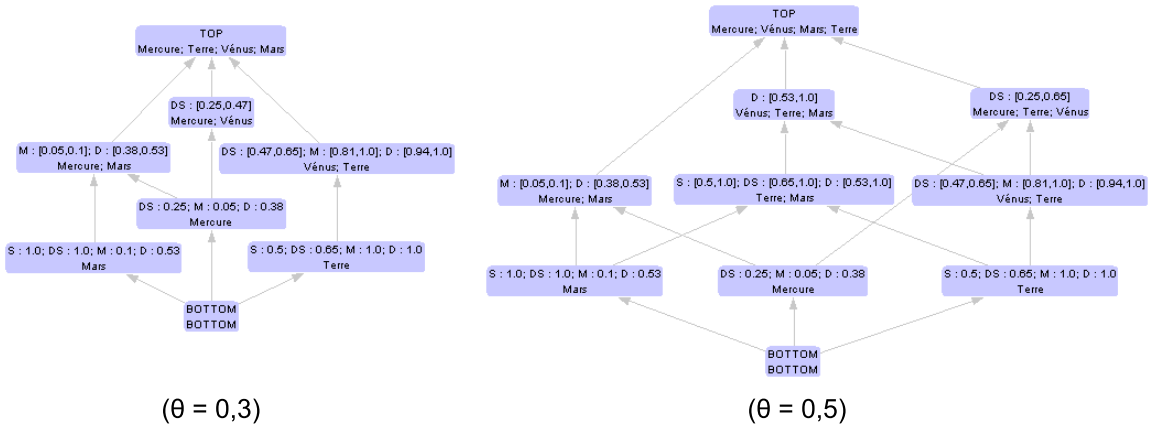


FIGURE 5.2 – Les treillis de concepts multivalués correspondant au contexte multivalué (G, M, W, I) donné en table 5.3 pour les seuils de variation $\theta = 0.3$ (gauche) et $\theta = 0.5$ (droite).

– L'apparition de nouveaux concepts. Un nouveau concept multivalué est créé dans le treillis lorsqu'un ensemble d'objets partagent un ensemble fermé d'attributs multivalués qui n'existe pas en tant qu'intension d'un concept du treillis (correspondant à l'ancien seuil θ). Dans les treillis de concepts multivalués donnés dans la figure 5.2, cet effet est illustré par l'apparition du concept multivalué $(\{Mars,\ Terre,\ Vénus\}, \{D_{[0.53,1]}\})$ dans le cas ou le seuil est $\theta = 0.5$.
– La disparition de concepts existants et l'agrandissement de la granularité d'autres concepts. Les concepts qui disparaissent sont des concepts dont l'extension et l'intension deviennent non fermés. Ils sont fusionnés dans les concepts à granularité plus large. Cet effet est illustré lors du passage du treillis de concepts multivalués obtenu pour un seuil $\theta = 0.8$ à celui obtenu pour un seuil $\theta = 0.9$ (figure 5.3). Les concepts $(\{Terre\}, \{D_1,\ DS_{0.65},\ M_1,\ S_{0.5}\})$ et $(\{Mars\}, \{D_{0.53},\ DS_1,\ M_{0.1},\ S_1\})$ sont supprimés. Ils sont fusionnés dans le concept $(\{Mars,\ Terre\}, \{D_{[0.53,1]},\ DS_{[0.65,1]},\ M_{[0.1,1]},\ S_{[0.5,1]}\})$ dont la granularité est augmentée.

De manière réciproque, la diminution de la granularité des concepts dans un treillis de concepts multivalués suite à la diminution du seuil de variation θ a pour résultat la répartition des objets dans l'extension de certains concepts en sous-ensembles respectant les nouvelles contraintes imposées par la diminution de θ. Cette répartition peut aussi avoir trois effets sur le treillis de concepts multivalués.

– La diminution de la granularité de certains concepts multivalués et la suppression de liens entre ces concepts et certains de leurs sous-concepts. La suppression de ces liens est due à la suppression des objets formant l'extension du sous-concept de l'extension du super-concept.
– La disparition de certains concepts. Les concepts supprimés sont ceux dont l'intension contient des attributs dont l'intervalle de similarité dépasse le nouveau seuil θ. Les objets

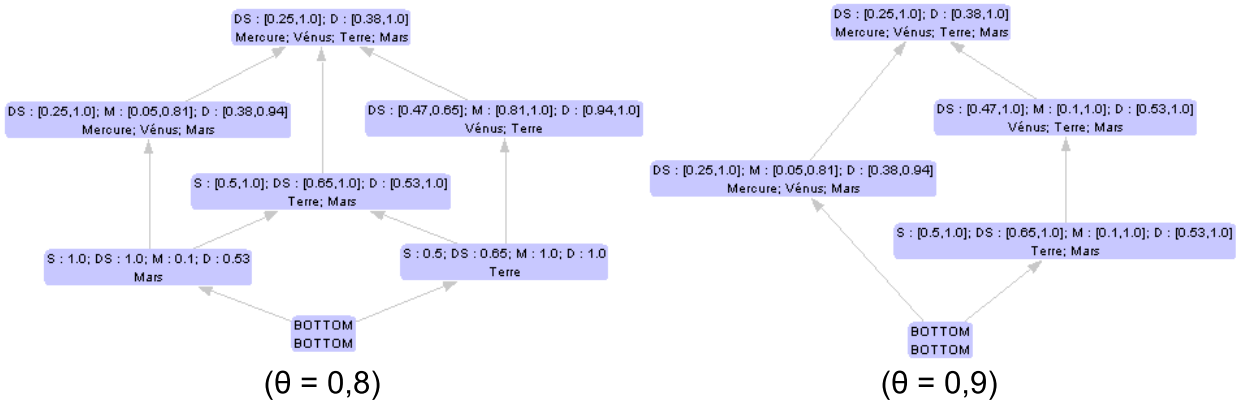


FIGURE 5.3 – Les treillis de concepts multivalués correspondant au contexte multivalué (G, M, W, I) donné en table 5.3 pour les seuils de variation $\theta = 0.8$ et $\theta = 0.9$.

dans l'extension de ces concepts ne partagent plus ces attributs.
- L'apparition de nouveau concepts multivalués à granularité fine. Ces concepts résultent de la repartition de l'extension d'un concept C en deux ou plusieurs parties qui ne sont pas extensions des sous-concepts de C.

L'illustration de ces effets est donnée par les mêmes exemples considérés précédemment en partant dans l'autre sens (du seuil le plus élevé au seuil le plus petit).

En conclusion, la variation du seuil de variation θ peut être vue comme une opération de raffinement ou de généralisation des concepts dans le treillis de concepts multivalués. En effet, l'augmentation de θ est suivie de la diminution de la précision des concepts (les intervalles de variation des valeurs des attributs dans les intensions des concepts sont agrandis). Elle correspond à une opération de regroupement des concepts du treillis pour obtenir des concepts de granularité plus large. granularité plus grande. Cependant la diminution du seuil maximal de variation θ est suivie de l'augmentation de la précision variation des valeurs des attributs dans les intensions des concepts sont réduits). Elle correspond au raffinement de ces concepts en des concepts de granularité plus fine.

5.3.9 Évolution du treillis de concepts multivalués

Les effets de la modification du seuil de variation θ sur un treillis de concepts multivalués, discutés dans la section précédente, dépendent des données contenues dans le contexte multivalué. Cependant, l'évolution du nombre de concepts dans un dans un treillis de concepts multivalués en fonction du seuil maximal de θ passe globalement par deux phases. Une première phase de croissance pour des seuils de variation bas suivie d'une deuxième phase de décroissance pour des seuils de variation élevés. On appellera ces deux phases phase de création et phase de fusion, respectivement. Cette évolution est illustrée par le graphique représentant l'évolution du nombre de concepts dans le treillis de concepts multivalués correspondant au contexte donné dans la table 5.3. Ce graphique est donné dans la figure 5.4. La première phase est obtenue pour des valeurs de θ allant de 0 à 0.6 et la deuxième est obtenue pour les valeurs allant de 0.6 à 1.

Phase de création

La phase de création est caractérisée par l'augmentation du nombre de concepts dans le treillis de concepts multivalués lorsqu'on augmente le seuil de variation θ. Cette augmentation

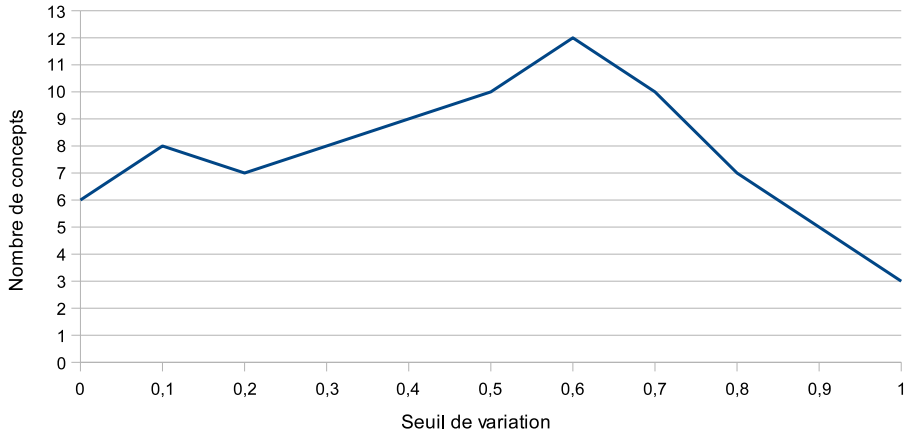


FIGURE 5.4 – L'évolution du treillis de concepts multivalués correspondant au contexte formel multivalué (G, M, W, I) donné dans la table 5.3, en terme de nombre de concepts multivalués, en fonction du seuil de variation θ.

est due à la création de nouveaux concepts regroupant des objets ayant des valeurs d'attributs qui deviennent similaires suite à suite à l'augmentation du seuil maximal de variation θ. la création de nouveaux concepts suite à l'augmentation du seuil l'augmentation du seuil maximal de variation est donnée à la figure 5.5.

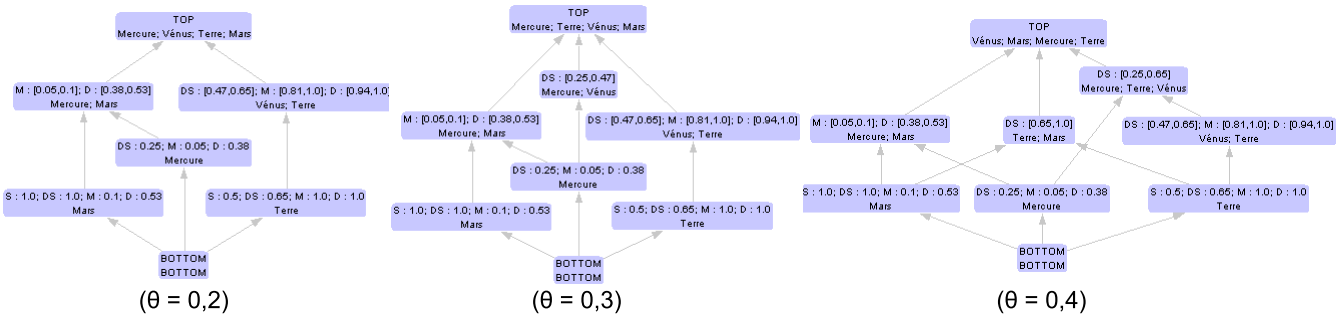


FIGURE 5.5 – L'évolution du treillis de concepts multivalués correspondant au contexte formel multivalué (G, M, W, I) donné à la table 5.3, en terme du nombre de concepts multivalués, en fonction du seuil de variation θ (θ varie entre 0.2 et 0.4) : la phase de création

Phase de fusion

La phase de fusion est caractérisée par la diminution du nombre de concepts dans le treillis de concepts multivalués lorsqu'on augmente le seuil de variation θ. Cette diminution est due à la fusion de certains concepts dans d'autres concepts de granularité plus large. L'illustration de la fusion de concepts suite à l'augmentation du seuil de variation est donné dans la figure 5.6.

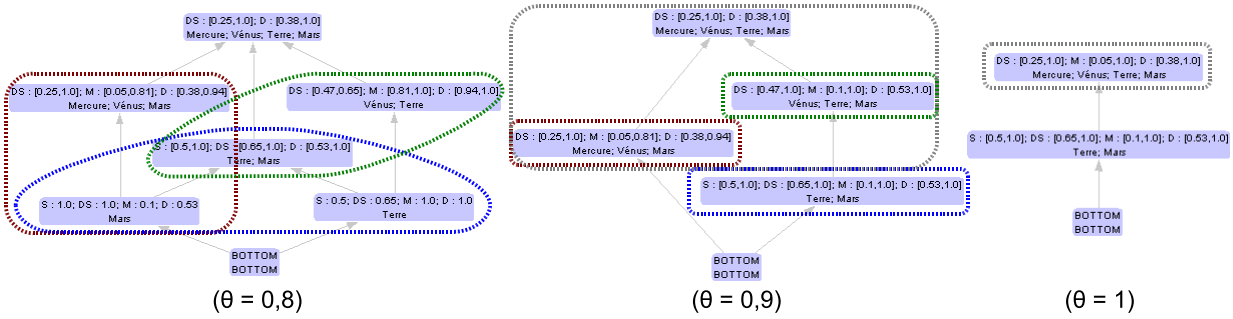


FIGURE 5.6 – L'évolution du treillis de concepts multivalués correspondant au contexte formel multivalué (G, M, W, I) donné à la table 5.3, en terme de nombre de concepts multivalués, en fonction du seuil de variation θ (θ varie entre 0.8 et 1) : la phase de fusion

5.3.10 Treillis extrêmes et liens avec l'échelonnage conceptuel

Les treillis de concepts multivalués correspondants à un contexte multivalué évoluent entre deux treillis particuliers : le treillis obtenu pour le seuil de variation minimal, $\theta = 0$, et celui et celui obtenu pour le seuil maximal de variation maximal, $\theta = 1$. premier seuil correspond au cas où aucune similarité n'existe entre les valeurs distinctes dans le contexte et le deuxième correspond au cas où toutes les valeurs sont similaires. Ces deux cas correspondent à deux échelonnages conceptuels particuliers dans le cas de l'ACF. Le premier échelonnage est celui connu sous le nom **échelonnage plan** (ou plat) [Ganter and Wille, 1999] qui consiste à produire un contexte binaire ayant comme ensemble d'attributs toutes les valeurs distinctes des attributs du contexte multivalué. Le contexte formel obtenu suite à l'application de cet échelonnage au contexte multivalué donné dans la table 5.3 est donné dans la table 5.4. Le treillis de concepts multivalués $\underline{\mathfrak{B}}_0(G, M, W, I)$ et le treillis de concepts formels correspondant à l'échelle obtenue par un échelonnage plan sont donnés dans la figure 5.7.

TABLE 5.4 – Contexte formel résultat de l'échelonnage plan du contexte multivalué donné à la table 5.3.

	Diamètre				Distance au Soleil				Masse				Satellite	
	0.38	0.53	0.94	1	0.25	0.47	0.65	1	0.05	0.1	0.81	1	0.5	1
Mercure	×				×				×					
Vénus			×			×					×			
Terre				×			×					×	×	
Mars		×						×		×				×

Le deuxième échelonnage consiste à garder le même ensemble d'attributs mais en remplaçant toutes les valeurs du contextes par "×". Le contexte formel obtenu suite à l'application de cet échelonnage au contexte multivalué donné dans la table 5.3 est donné dans la table 5.5. La seule différence entre le treillis de concepts correspondant à cet échelonnage et le treillis de concepts multivalués obtenu pour $\theta = 1$ est que dans celui-ci les intensions des concepts sont formées par des attributs accompagnés chacun par son domaine (l'intervalle maximale entre les valeurs de l'attributs). Cette différence peut être constatée sur les treillis donnés à la figure 5.8.

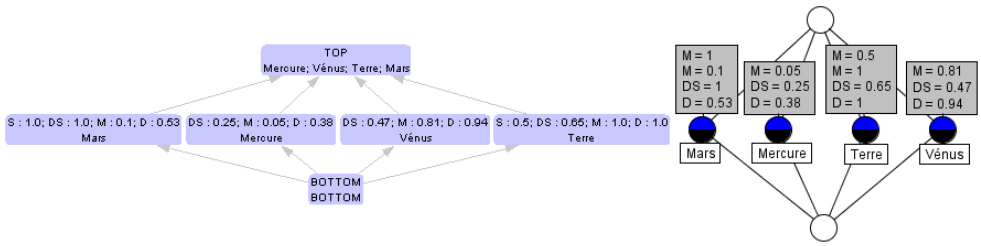


FIGURE 5.7 – Treillis de concepts multivalués $\underline{\mathfrak{B}}_0(G, M, W, I)$ (gauche) et treillis de concepts formels $\underline{\mathfrak{B}}(G, M, I)$ correspondant à l'échelle donné à la table 5.4 (droite).

TABLE 5.5 – Contexte formel résultat de l'échelonnage du contexte multivalué donné à la table 5.3 en remplaçant les valeurs par "×".

	Diamètre	Distance au Soleil	Masse	Satellite
Mercure	×	×	×	
Vénus	×	×	×	
Terre	×	×	×	×
Mars	×	×	×	×

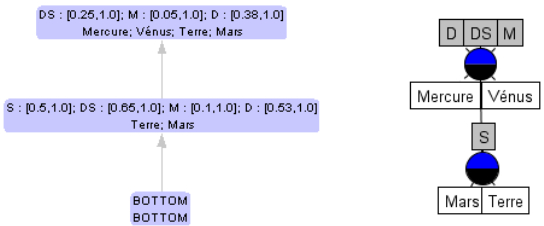


FIGURE 5.8 – Treillis de concepts multivalués $\underline{\mathfrak{B}}_1(G, M, W, I)$ (gauche) et treillis de concepts formels $\underline{\mathfrak{B}}(G, M, I)$ correspondant à l'échelle donné à la table 5.5 (droite).

5.4 Études des contextes symboliques

5.4.1 Contextes symboliques

Un contexte symbolique est un contexte multivalué dont les valeurs sont des données symboliques [Bock and Diday, 2000].

Un exemple de contexte symbolique, tiré de BioRegistry, est donné à la table 5.6. Les objets sont des bases de données biologiques et les attributs sont les ressources sémantiques utilisées pour l'annotation de ces bases de données : "*NCBI taxonomy*[30]", "*MeSH thesaurus*[31]" et la hiérarchie de catégories de NAR "*Nucleic Acid Research (NAR)*". Les valeurs prises par les attributs NCBI et MeSH pour une base de données sont des "*termes*" appartenant aux ressources sémantiques correspondantes utilisées pour annoter la base de données considérée. La valeur prise par l'attribut NAR donne la catégorie à laquelle appartient la base de données. Une valeur de l'attribut NAR est interprétée comme *catégorie.sous-catégorie*. Par exemple la valeur "1.2" correspond à la deuxième sous-catégorie "*Gene structure, introns and exons, splice sites*", de la première catégorie, "*Nucleotide Sequence Databases*"[32].

Table 5.6 – Un exemple de contexte symbolique représentant un extrait des données de BioRegistry.

	NCBI	MeSH	NAR
ExInt	Eukaryotes	Gene components	1.2
HSD	Human	Proteins	7.3
rRNDB	Prokaryotes	Genomics	5.2
SpliceDB	Mammals	Gene components	1.2
CropNet	Plants		13
GOLD		Genomics	5.2
INE	Rice		13
TRANSCompel	Vertebrates	Transcription factors	1.2

5.4.2 Similarité entre les valeurs dans un contexte symbolique

Les récents développements et les travaux considérables effectués dans le domaine du Web sémantique ont abouti à la mise en place de plusieurs ressources sémantiques fiables telles que l'ontologie Gene Ontology les thésaurus MeSH et WordNet, etc. Ces ressources sémantiques ont par la suite été largement utilisées pour annoter les sources de données hétérogènes et complexes accessibles via le Web. Le but du recours aux ressources sémantiques est de faciliter l'organisation, l'identification et l'interaction avec les ressources du Web. Dans le cadre de notre travail, l'utilisation de ces ressources sémantiques nous permet de prendre en compte les relations sémantiques entre les valeurs des attributs multivalués dans un contexte symbolique. En s'appuyant sur les relations sémantiques entre les valeurs dans un contexte symbolique, il est possible de calculer la similarité entre toute paire de valeurs d'un attribut du contexte.

30. http ://www.ncbi.nlm.nih.gov/
31. http ://www.nlm.nih.gov/mesh/
32. La hiérarchie complète des catégories NAR avec les noms complets des catégories est disponible à l'adresse suivante http ://www.oxfordjournals.org/nar/database/c

Dans l'exemple de contexte symbolique donné dans la table 5.6, les fragments des ontologies illustrant les relations sémantiques entre les valeurs des attributs du contexte sont donnés dans la figure 5.9.

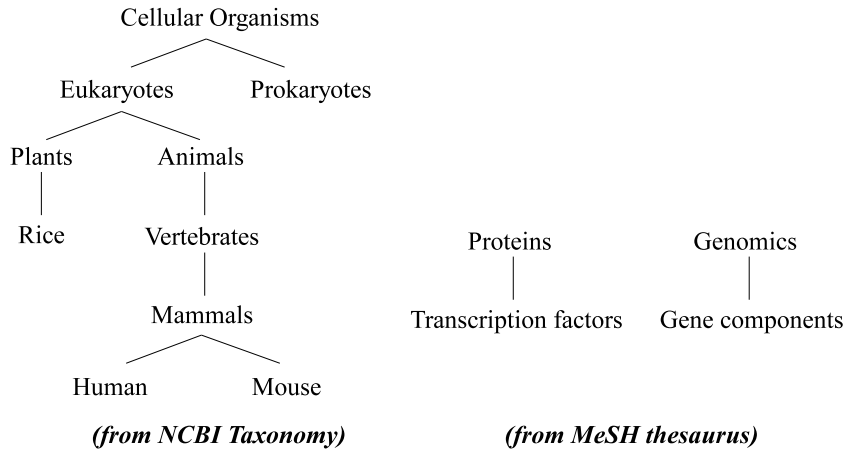


FIGURE 5.9 – Fragments de hiérarchies contenant les valeurs des attributs du contexte multivalué donné à la table 5.6.

Il existe plusieurs méthodes de calcul de la similarité dans une ontologie. Une comparaison détaillée de ces méthodes a été faite dans [Hliaoutakis et al., 2006]. Dans le cas du contexte donné à la table 5.6, nous utilisons l'une de premières méthodes de calcul de similarité dans une hiérarchie de termes à savoir celle de Rada [Rada et al., 1989]. Cette méthode considère que la similarité entre deux termes dans une hiérarchie est inversement proportionnelle à la distance entre les nœuds qui les représentent dans la hiérarchie. Dans l'exemple donné à la figure 5.9, la similarité entre *Mammals* et *Vertebrates* est supérieure à celle entre *Mammals* et *Animals*. Pour la formalisation du calcul de la similarité selon cette méthode, nous supposons qu'une hiérarchie de termes se présente sous la forme d'un arbre qu'on notera par $\mathcal{T} = (V, E)$ où V est l'ensemble des termes et E est l'ensemble des liens entre ces termes en suivant un chemin ascendant ou descendant dans la hiérarchie. On notera par $d(v_1, v_2)$ la distance entre les deux termes $v_1, v_2 \in V$ dans $\mathcal{T}$. Par exemple, $d(Mammals, Vertebrates) = 1$ et $d(Mammals, Animals) = 2$.

La possibilité de définir la similarité entre les valeurs des attributs multivalués dans un contexte symbolique, permet de procéder de manière analogue au cas des contextes numériques pour définir les concepts multivalués et le treillis de concepts multivalués à partir d'un contexte symbolique.

Définition 43 *Étant donné une hiérarchie de termes $\mathcal{T}$, un contexte symbolique (G, M, W, I) et deux valeurs w_i et w_j d'un même attribut m. w_i et w_j sont dits similaires si et seulement si $d(w_i, w_j) \leq S$ (S étant un seuil de similarité choisi préalablement). Dans ce cas on note $w_i \simeq w_j$.*

Dans le contexte symbolique donné dans la table 5.6 et pour un seuil de similarité $S = 2$ dans les hierarchies données dans la figure 5.9, on a *Human* $\simeq$ *Mammals* et *Human* $\simeq$ *Vertebrates* tandis que *Human* $\not\simeq$ *Animals* et *Prokaryotes* $\not\simeq$ *Eukaryotes*.

Partage d'attributs entre les objects dans un contexte symbolique

Étant donné qu'il est possible de calculer la similarité entre les valeurs prises par un attribut pour différents objets dans un contexte symbolique, on peut définir le partage d'attributs entre objets par analogie avec les contextes numériques en s'appuyant sur la similarité entre les valeurs des attributs. De manière intuitive, un ensemble d'objets partagent un attribut dans un contexte multivalué lorsque les valeurs prises par cet attribut pour ces objets sont similaires.

Définition 44 (Partage d'attributs) *Soit un contexte symbolique* (G, M, W, I).

1. *Deux objets* $g_i, g_j \in G$ *partagent un attribut* $m \in M$ *si et seulement si* $m(g_i) \simeq m(g_j)$.
2. *Ainsi les objets* g_i *et* g_j *partagent les valeurs* $m(g_i)$ *et* $m(g_j)$ *de l'attribut* m *et* $\{m(g_i), m(g_j)\}$ *est l'ensemble des valeurs similaires de* m *pour les objets* g_i *et* g_j *:* g_i *et* g_j *partagent* $m_{\,|\,\{m(g_i),m(g_j)\}}$.
3. *Plus généralement, les objets d'un ensemble d'objets* $A \subseteq G$ *partagent un attribut* m *si et seulement si toute paire d'objets dans* A *partage* m. *L'ensemble des valeurs similaires de* m *pour* A *est* $\{m(g), g \in A\}$. *On dit que les objets dans* A *partagent* $m_{\,|\,\{m(g),g\in A\}}$

Dans le contexte symbolique donné dans la table 5.6 et pour un seuil de similarité $S = 2$, les objets *HSD* et *SpliceDB* partagent l'attribut $NCBI_{\,|\,\{Human,Mammals\}}$ puisque $Human \simeq Mammals$. Par contre, ces deux objets ne partagent pas l'attribut *MeSH* puisque $Proteins \not\simeq Gene\ components$.

En considérant la définition 44, un attribut peut être partagé par deux ou plusieurs ensembles d'objets séparément sans être partagé par tous les objets des deux ensembles. Ceci est le cas de l'attributs *NCBI* dans le contexte donné dans la table 5.6. En effet, *HSD* et *SpliceDB* partagent $NCBI_{\,|\,\{Human,Mammals\}}$ et *CropNet* et *INE* partagent $NCBI_{\,|\,\{Plants,Rice\}}$ alors que *HSD* et *INE* ne partagent pas *NCBI* puisque $Rice \not\simeq Human$.

Dans la suite, la définition de partage d'attributs entre objets est utilisée pour définir la connexion de Galois par similarité dans les contextes symboliques.

5.4.3 Ordres Partiels, opérateurs de dérivation et connexion de Galois par similarité

La connexion de Galois à définir dans le cas des contextes symboliques doit établir le lien entre ensembles d'objets et ensembles d'attributs avec valeurs de similarités. Formellement, étant donné un contexte symbolique (G, M, I, W), les ensembles d'objets sont des parties de G, donc des éléments de $\mathfrak{P}(G)$, et les ensembles d'attributs avec valeurs de similarité sont des parties de $M \times \mathfrak{P}(W)$, des éléments de $\mathfrak{P}(M \times \mathfrak{P}(W))$, (chaque attribut de M est accompagné d'un ensemble de valeurs inclus dans W). Les éléments de $M \times \mathfrak{P}(W)$ sont des paires de la formes (m, W_m) où $m \in M$ est un attribut et $W_m \subseteq W$ est un ensemble de valeurs possibles de m. Pour plus de lisibilité et pour assurer la cohérence avec la notation introduite en définition 44, nous utiliserons la notation $m_{\,|\,W_m}$ au lieu (m, W_m). Un exemple d'éléments de $\mathfrak{P}(M \times \mathfrak{P}(W))$ dans le cas du contexte symbolique donné dans la table 5.6 est $\{NCBI_{\,|\,\{Vertebrates,Mammals\}}, NAR_{\,|\,\{1.2\}}\}$.

Ordres partiels

Il est possible de définir plusieurs relations d'ordre partiel sur $\mathfrak{P}(G)$ et sur $\mathfrak{P}(M \times \mathfrak{P}(W))$. Une relation d'ordre partiel possible sur $\mathfrak{P}(G)$ est l'inclusion "$\subseteq$" entre ensembles comme dans le cas de l'ACF. Par exemple, dans le contexte symbolique donné dans la table 5.6, {*ExInt, HSD*} $\subseteq$ {*ExInt, HSD, rRNDB*}. $(\mathfrak{P}(G), \subseteq)$ est ainsi un ensemble partiellement ordonné.

Une relation d'ordre partiel sur $\mathfrak{P}(M \times \mathfrak{P}(W))$ qui s'appuie sur "$\subseteq$" et prend en compte les valeurs de similarité des attributs étant donné un seuil de similarité S est notée par "$\subseteq_S$". Nous appelons "$\subseteq_S$" opérateur d'**inclusion par similarité** et nous définissons l'inclusion par similarité comme suit.

Définition 45 ($\subseteq_S$) *Soient $B_1,\ B_2 \in \mathfrak{P}(M \times \mathfrak{P}(W))$. On dit que B_1 est inclus dans B_2 et on note $B_1 \subseteq_S B_2$ si et seulement si $\forall m_1 \,|_{W_1} \in B_1,\ \exists\ m_2 \,|_{W_2} \in B_2$ tel que $m_1 = m_2$ et $W_2 \subseteq W_1$.*

Dans le contexte symbolique donné dans la table 5.6 et pour un seuil de similarité $S = 2$ dans les hiérarchies données dans la figure 5.9, on a $\{NCBI\,|_{\{Vertebrates,Mammals\}},\ NAR\,|_{\{1.2\}}\} \subseteq_S \{NCBI\,|_{\{Mammals\}},\ NAR\,|_{\{1.2\}},\ MeSH\,|_{\{Gene\ Comp.\}}\}$.

Proposition 5 *($\mathfrak{P}(M \times \mathfrak{P}(W)), \subseteq_S$) est un ensemble partiellement ordonné (i.e. "$\subseteq_S$" est réflexif, anti-symétrique et transitif).*

Preuve 6
- *Pour tout $B \in \mathfrak{P}(M \times \mathfrak{P}(W))$, $B \subseteq_S B$ et "$\subseteq_S$" est réflexif.*
- *Soient $B_1,\ B_2 \in \mathfrak{P}(M \times \mathfrak{P}(W))$ tels que $B_1 \subseteq_S B_2$ (1) et $B_1 \neq B_2$ (2). (1) et (2) impliquent qu'il existe au moins un attribut $m_2\,|_{W_2} \in B_2$ qui vérifie l'une des conditions suivantes : (i) il n'existe aucun $m_1\,|_{W_1}$ dans B_1 tel que $m_2 = m_1$ (ii) il existe $m_1\,|_{W_1}$ dans B_1 tel que $m_2 = m_1$ et $W_2 \subset W_1$. Dans les deux cas (i) et (ii), la définition 45 n'est pas vérifiée pour $B_2 \subseteq_S B_1$. Par conséquent "$\subseteq_S$" est anti-symétrique.*
- *Soient $B_1,\ B_2,\ B_3 \in \mathfrak{P}(M \times \mathfrak{P}(W))$ tels que $B_1 \subseteq_S B_2$ et $B_2 \subseteq_S B_3$. $B_1 \subseteq_S B_2$ implique que $\forall m_1\,|_{W_1} \in B_1,\ \exists\ m_2\,|_{W_2} \in B_2$ tel que $m_1 = m_2$ et $W_2 \subseteq W_1$ (1). $B_2 \subseteq_S B_3$ implique que $\forall m_2\,|_{W_2} \in B_2,\ \exists\ m_3\,|_{W_3} \in B_3$ tel que $m_2 = m_3$ et $W_3 \subseteq W_2$ (1). (1) et (2) impliquent que $\forall m_1\,|_{W_1} \in B_1,\ \exists\ m_3\,|_{W_3} \in B_3$ tel que $m_1 = m_3$ et $W_3 \subseteq W_1$ ce qui signifie, par la définition 45, que $B_1 \subseteq_S B_3$. Par conséquent "$\subseteq_S$" est transitif.*

Opérateurs de dérivation

Les opérateurs de dérivation à définir sur un contexte symbolique doivent associer à un ensemble d'objets l'ensemble de leurs attributs communs avec l'ensemble des valeurs similaires et inversement. Ils sont définis entre $\mathfrak{P}(G)$ et $\mathfrak{P}(M \times \mathfrak{P}(W))$ comme suit.

Définition 46 (Opérateurs de dérivation) *Soit un contexte symbolique (G, M, W, I),*
- *Pour tout ensemble d'objets $A \subseteq G$,*

$$A^{\uparrow_S} = \{m\,|_{\{m(g), g \in A\}} \in M \times \mathfrak{P}(W) \text{ tel que } \forall g_i, g_j \in A, m(g_i) \simeq m(g_j)\}$$

  est l'ensemble de tous les attributs de $M \times \mathfrak{P}(W)$ partagés par tous les objets de A.
- *De manière duale, pout tout ensemble d'attributs $B \subseteq M \times \mathfrak{P}(W)$,*

$$B^{\downarrow_S} = \{g \in G \text{ tel que } \forall\ m\,|_{W_m} \in B, m(g) \simeq w\ \forall w \in W_m\}$$

  est l'ensemble de tous les objects de G ayant les attributs de B.

Dans le contexte symbolique donné dans la table 5.6 et pour un seuil de similarité $S = 2$, on a :

$$\{INE, CropNet\}^{\uparrow_S} = \{NCBI_{\,|\,\{Plants,Rice\}}, NAR_{\,|\,\{13\}}\}$$

et

$$\{NCBI_{\,|\,\{Plants,Rice\}}, NAR_{\,|\,\{13\}}\}^{\downarrow_S} = \{INE, CropNet\}.$$

Connexion de Galois par similarité

Théorème 5 (Connexion de Galois par similarité) *Les opérateurs de dérivation $^{\uparrow_S}$ et $^{\downarrow_S}$ (définition 46) forment une connexion de Galois, appelée* ***connexion de Galois par similarité****, entre $(\mathfrak{P}(G), \subseteq)$ et $(\mathfrak{P}(M \times \mathfrak{P}(W)), \subseteq_S)$.*

Preuve 7 *D'après la définition de la connexion de Galois donnée dans [Ganter and Wille, 1999], on a : $^{\uparrow_S}$ et $^{\downarrow_S}$ forment une connexion de Galois entre $(\mathfrak{P}(G), \subseteq)$ et $(\mathfrak{P}(M \times \mathfrak{P}(W)), \subseteq_S)$ si et seulement les trois conditions suivantes sont vérifiées pour tout $A, A_1, A_2 \in \mathfrak{P}(G)$ et $B, B_1, B_2 \in \mathfrak{P}(M \times \mathfrak{P}(W))$:*

1. *$A_1 \subseteq A_2$ implique $A_2^{\uparrow_S} \subseteq_S A_1^{\uparrow_S}$*
2. *$B_1 \subseteq_S B_2$ implique $B_2^{\downarrow_S} \subseteq B_1^{\downarrow_S}$*
3. *$A \subseteq (A^{\uparrow_S})^{\downarrow_S}$ et $B \subseteq_S (B^{\downarrow_S})^{\uparrow_S}$*

1. *Si $m_{2\,|\,W_2} \in A_2^{\uparrow_S}$, alors d'après la définition 46 on a, $m_2(g_i) \simeq m_2(g_j)\ \forall g_i, g_j \in A_2$ en particulier $m_2(g_i) \simeq m_2(g_j)\ \forall g_i, g_j \in A_1$ puisque $A_1 \subseteq A_2$. En conséquence $m_{2\,|\,\{m_2(g),\, g \in A_1\}} \in A_1$. De plus $\{m_2(g),\ g \in A_1\} \subseteq \{m_2(g),\ g \in A_2\}$ since $A_1 \subseteq A_2$. Finalement, si on note $\{m_2(g),\ g \in A_2\}$ par W_2 et $\{m_2(g),\ g \in A_1\}$ par W_1, on obtient : $\forall m_{2\,|\,W_2} \in A_2^{\uparrow_S}\ \exists m_{1\,|\,W_1} \in A_1^{\uparrow_S}$ tel que $m_1 = m_2$ et $W_1 \subseteq W_2$ ce qui signifie que $A_2^{\uparrow_S} \subseteq_S A_1^{\uparrow_S}$.*
2. *Si $g \in B_2^{\downarrow_S}$ alors $\forall m_{2\,|\,W_2} \in B_2,\ m_2(g) \simeq w\ \forall w \in W_2$ en particulier, puisque $B_1 \subseteq_S B_2$, $\forall m_{1\,|\,W_1} \in B_1,\ m_1(g) \simeq w\ \forall w \in W_1$ ce qui signifie que $g \in B_1^{\downarrow_S}$. Par conséquent $B_2^{\downarrow_S} \subseteq B_1^{\downarrow_S}$.*
3. *Si $g \in A$ alors $\forall m_{\,|\,W} \in A^{\uparrow_S},\ m(g) \simeq w\ \forall w \in W$ ce qui implique (par définition de "$\downarrow_S$") que $g \in (A^{\uparrow_S})^{\downarrow_S}$. Par conséquent $A \subseteq (A^{\uparrow_S})^{\downarrow_S}$. De manière duale, si $m_{\,|\,W} \in B$ then $\forall g \in B^{\downarrow_S},\ m(g) \simeq w\ \forall w \in W$ ce qui implique (par définition de "$\uparrow_S$") que $m_{\,|\,W} \in (B^{\downarrow_S})^{\uparrow_S}$. Par conséquent $B \subseteq_S (B^{\downarrow_S})^{\uparrow_S}$.*

La connexion de Galois par similarité est à l'origine de la définition des concepts formels multivalués et des treillis de concepts multivalués dans le cas des contextes symboliques.

5.4.4 Concepts formels multivalués et treillis de concepts multivalués

Définition 47 (Concept multivalué) *Soient (G, M, I, W) un contexte symbolique et S un seuil de similarité des valeurs dans W. Un concept formel multivalué (ou concept multivalué) est une paire (A, B), où $A \subseteq G$ et $B \subseteq M \times \mathfrak{P}(W)$, telle que $A^{\uparrow_S} = B$ et $B^{\downarrow_S} = A$. A et B sont respectivement l'extension et l'intension de (A, B). L'ensemble de tous les concepts multivalués dans (G, M, I, W), étant donné un seuil de similarité S, est noté par $\mathfrak{B}_S(G, M, I, W)$.*

Dans le contexte symbolique donné dans la table 5 et pour un seuil de similarité $S = 2$,

$$(\{INE, CropNet\}, \{NCBI_{|\{Plants,Rice\}}, NAR_{|\{13\}}\})$$

et

$$(\{HSD, SpliceDB, TRANSCompel\}, \{NCBI_{|\{Human,Mammals,Vertebrates\}}\})$$

sont deux exemples de concepts multivalués.

Définition 48 ($\leq_S$) *Soient (G, M, W, I) un contexte symbolique, S un seuil de similarité et (A_1, B_1) et (A_2, B_2) deux concepts multivalués. (A_1, B_1) est dit sous-concept de (A_2, B_2) lorsque $A_1 \subseteq A_2$ (ou de manière équivalente $B_2 \subseteq_S B_1$). Dans ce cas (A_2, B_2) est dit super-concept de (A_1, B_1) et on note $(A_1, B_1) \leq_S (A_2, B_2)$. La relation $\leq_S$ est appelée relation d'ordre hiérarchique sur les concepts multivalués.*

Dans le contexte symbolique donné dans la table 5 et pour un seuil de similarité $S = 2$, on a :

$$(\{SpliceDB, TRANSCompel\}, \{NCBI_{|\{Mammals,Vertebrates\}}, NAR_{|\{1.2\}}\})$$

$$\leq_S$$

$$(\{HSD, SpliceDB, TRANSCompel\}, \{NCBI_{|\{Human,Mammals,Vertebrates\}}\})$$

Définition 49 (Treillis de concepts multivalués) *Soient (G, M, I, W) un contexte symbolique et S un seuil de similarité. L'ensemble de concepts multivalués $\mathfrak{B}_S(G, M, I, W)$ ordonné selon la relation d'ordre hiérarchique entre concepts multivalués $\leq_S$, $(\mathfrak{B}_S(G, M, I, W), \leq_S)$, forme un treillis de concepts appelé treillis de concepts multivalués et noté par $\underline{\mathfrak{B}}_S(G, M, W, I)$.*

Le treillis de concepts multivalués correspondant au contexte symbolique donné dans la table 5.6 pour un seuil de similarité $S = 2$, $\underline{\mathfrak{B}}_2(G, M, W, I)$ est donné dans la figure 5.10.

5.4.5 Variation de la granularité des concepts multivalués et évolution du treillis de concepts multivalués

De la même manière que dans le cas des contextes numériques, la variation du seuil de similarité S dans un contexte symbolique entraîne la variation de la condition du partage d'attributs multivalués entre les objets du contexte. Ceci nous conduit à définir la granularité d'un concept multivalué dans un contexte symbolique en fonction du seuil de similarité S.

Définition 50 (Granularité d'un concept multivalué) *Soit (G, M, W, I) un context symbolique et soient (A_1, B_1) et (A_2, B_2) deux concepts multivalués obtenus à partir de (G, M, W, I) pour deux seuils de similarité S_1 et S_2 différents tels que B_1 et B_2 ont des attributs en commun si on ignore les valeurs de similarité. On dit que (A_1, B_1) est **de granularité plus fine** que (A_2, B_2) (auquel cas (A_2, B_2) est **de granularité plus grossière** que (A_1, B_1)) si les valeurs de similarités des attributs dans B_1 sont plus petites que celles dans B_2. Dans ce cas (A_1, B_1) est dit **plus précis** que (A_2, B_2) ((A_2, B_2) est dit **plus général** que (A_1, B_1)).*

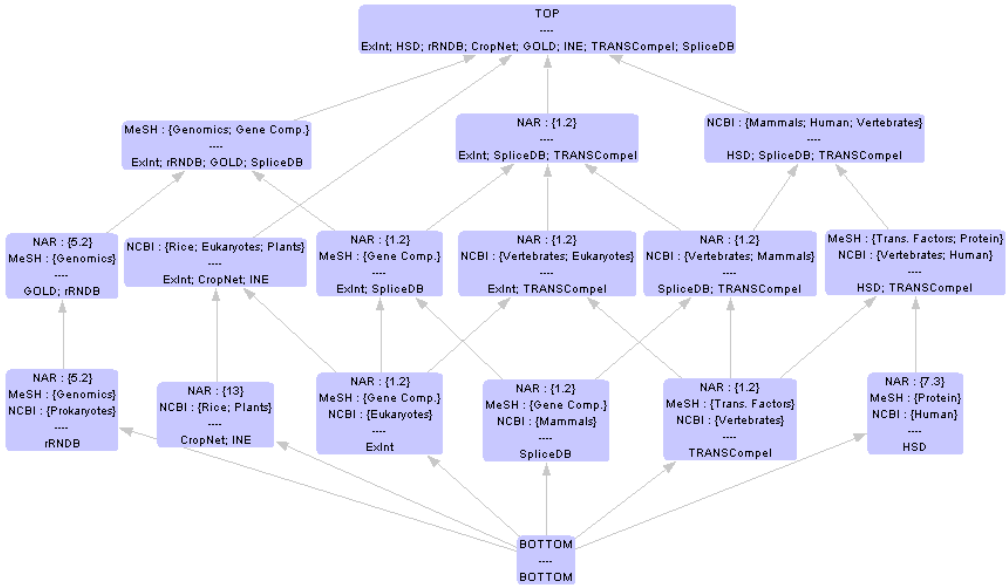


FIGURE 5.10 – $\underline{\mathfrak{B}}_2(G, M, W, I)$ correspondant au contexte symbolique (G, M, W, I) donné dans la table 5.6.

Le treillis de concepts multivalués correspondant à un contexte symbolique évolue en fonction du seuil de similarité S choisi de la même manière que les treillis de concepts multivalués dans le cas de contextes numérique. L'évolution se fait entre deux treillis extrêmes, celui obtenu pour $S = 0$ et celui obtenu pour la valeur maximale de S. La valeur maximale de S correspond au chemin le plus long possible dans la hiérarchie de termes entre deux valeurs d'un attribut dans le contexte. Dans l'exemple donné dans la figure 5.9, la valeur maximale de S est 4. Le treillis de concepts multivalués $\underline{\mathfrak{B}}_0(G, M, W, I)$ correspondant au contexte symbolique donné dans la table 5.6 pour un seuil $S = 0$ est donné dans la figure 5.11. $\underline{\mathfrak{B}}_0(G, M, W, I)$ est le treillis le plus spécifique possible correspondant au contexte considéré. Dans ce treillis, la majorité des concepts ont des extensions qui contiennent un seul objet. Les seuls groupements d'objets dans les extensions des concepts sont dus aux partage d'une même valeur d'un même attribut. Ceci est le cas des objets *CropNet* et *INE* qui ont tous les deux la valeur 13 de l'attribut *NAR*. Ce treillis de concepts multivalués particulier est isomorphe au treillis de concepts formels obtenu après l'échelonnage plat du contexte symbolique donné dans la table 5.6.

Le treillis de concepts multivalués $\underline{\mathfrak{B}}_4(G, M, W, I)$ correspondant au contexte symbolique donné dans la table 5.6 pour un seuil $S = 4$ est donné dans la figure 5.11. $\underline{\mathfrak{B}}_4(G, M, W, I)$ est le treillis le plus général possible correspondant au contexte symbolique considéré compte tenu des similarités possibles entre les valeurs des attributs dans la figure 5.9. Comparé à $\underline{\mathfrak{B}}_0(G, M, W, I)$, le treillis $\underline{\mathfrak{B}}_4(G, M, W, I)$ contient plus de concepts dont les intensions sont formés par plusieurs objets. Ceci s'explique par la relâchement de la condition de partage d'attributs entre concepts.

Si nous admettons que pour des seuils de similarité élevés on considère que toutes les valeurs d'un même attribut dans le contexte symbolique (table 5.6) sont similaires, nous pouvons obtenir un treillis de concepts multivalués encore plus général que $\underline{\mathfrak{B}}_4(G, M, W, I)$. Ce treillis, noté par $\underline{\mathfrak{B}}_{max}(G, M, W, I)$, est donné à la figure 5.13. $\underline{\mathfrak{B}}_{max}(G, M, W, I)$ est isomorphe au treillis de concepts formels obtenu après l'échelonnage conceptuel particulier qui consiste à remplacer toute valeur non vide dans le contexte symbolique par "×".

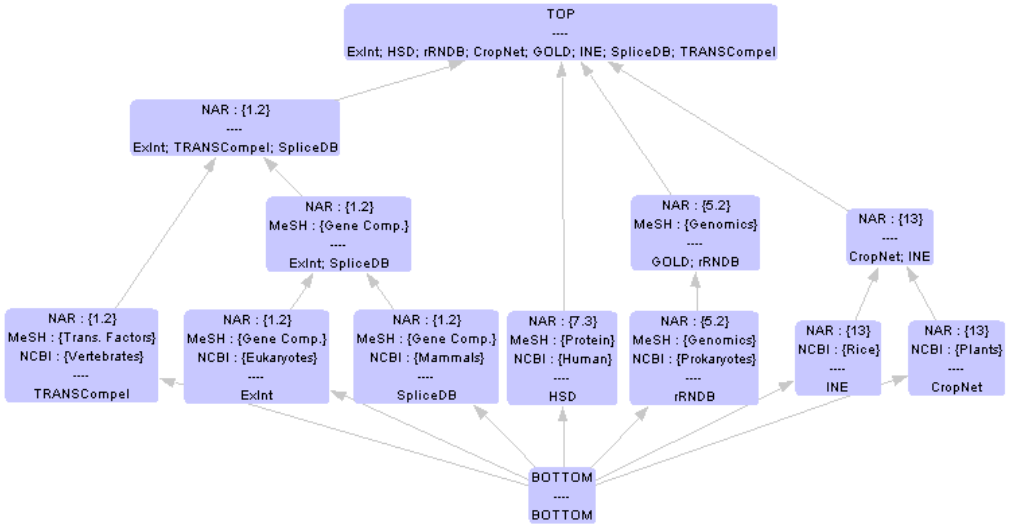


FIGURE 5.11 – $\underline{\mathfrak{B}}_0(G, M, W, I)$ correspondant au contexte symbolique (G, M, W, I) donné dans la table 5.6.

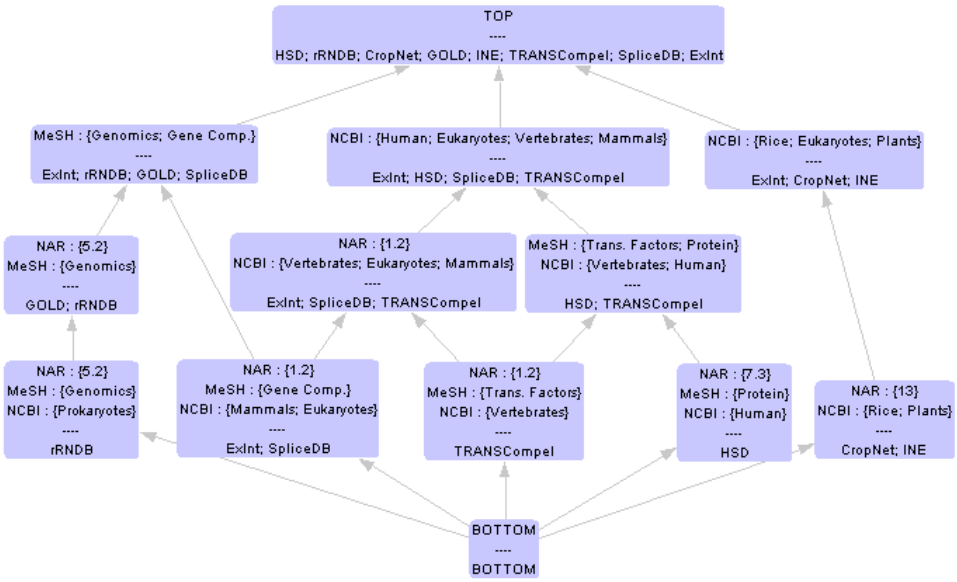


FIGURE 5.12 – $\underline{\mathfrak{B}}_4(G, M, W, I)$ correspondant au contexte symbolique (G, M, W, I) donné à la table 5.6.

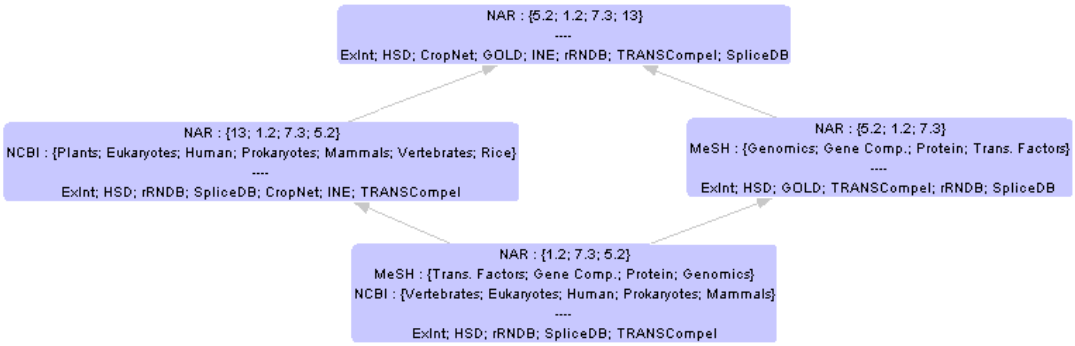


FIGURE 5.13 – $\underline{\mathfrak{B}}_{max}(G, M, W, I)$ correspondant au contexte symbolique (G, M, W, I) donné dans la table 5.6.

Entre les treillis de concepts multivalués correspondant aux valeurs extrêmes du seuil de similarité ($S = 0$ et $S = 4$ ou $S = max$), l'évolution du nombre de concepts dans le treillis passe par les deux phases suivantes :

1. **La phase de création** caractérisée par l'augmentation du nombre de concepts dans le treillis de concepts multivalués lorsqu'on augmente le seuil de similarité S. Cette augmentation est due à la création de nouveaux concepts regroupant des objets ayant des valeurs d'attributs qui deviennent similaires.
2. **La phase de fusion** caractérisée par la diminution du nombre de concepts dans le treillis de concepts multivalués lorsqu'on augmente le seuil de similarité S. Cette diminution est due à la fusion de certains concepts dans d'autres concepts de granularité plus large.

L'évolution du nombre de concepts en fonction du seuil dans l'exemple de contexte symbolique considéré est donné dans la table 5.7.

TABLE 5.7 – Le nombre de concepts multivalués extraits à partir du contexte donné dans la table 5.6 en fonction du seuil de similarité S.

Seuil de similarité	0	1	2	3	4	max
Nombre de concepts	13	16	17	13	12	4

5.5 Généralisation aux contextes plurivalués et aux contextes hétérogènes

Dans certains cas, les contextes formels multivalués sont insuffisants pour représenter de manière exhaustive un ensemble de données complexes. En effet, la relation entre un objet et un attribut peut prendre plusieurs valeurs plutôt qu'une valeur unique. Ceci est le cas des données de BioRegistry (un exemple est donné dans la table 4.1). Afin de permettre la prise en compte des données de cette forme, nous définissons les contextes plurivalués. Ces contextes constituent une généralisation des contextes multivalués dans la mesure où une case dans la représentation tabulaire du contexte peut contenir un ensemble de valeurs. Formellement, on définit un contexte plurivalué comme suit.

Définition 51 (Contexte plurivalué) *Un contexte plurivalué est un quadruplet* (G, M, I, W) *où* G *est un ensemble d'objets,* M *un ensemble d'attributs (plurivalués),* W *un ensemble de valeurs et* I *est une relation ternaire telle que* $I(g, m) \subseteq W$ *est l'ensemble de valeurs prises par* m *pour l'objet* g*.*

Dans le cas général, un contexte plurivalué permet de représenter des données hétérogènes et très variées. Cependant, selon la nature des valeurs des attributs du contextes (éléments de W) et selon les relations entre ces valeurs, des formes typiques de contextes plurivalués peuvent être distinguées. Par analogie avec les contextes multivalués étudiés précédemment, il est possible de distinguer les contextes plurivalués numériques, les contextes plurivalués symboliques et les contextes plurivalués hétérogènes. Pour chaque type de contexte, les relations entre les éléments de W (totalement ou partiellement ordonnés) ainsi que la forme des valeurs des attributs dans le contexte (intervalles ou ensembles discrets d'éléments de W) peuvent être considérées lors de la définition de méthodes d'exploitation de ces contextes.

L'approche d'analyse de données par similarité définie précédemment pour les contextes multivalués peut être généralisée aux contextes plurivalués. La généralisation s'appuie sur la définition du partage d'attributs plurivalués entre objets qui prend en compte la similarité entre les ensembles de valeurs des attributs dans le contexte plurivalué. Le calcul de la similarité entre les valeurs des attributs dans un contexte plurivalué dépend de la nature de ces valeurs et des relations entre elles. Par analogie avec le cas des contextes multivalués, nous illustrons l'extension de l'analyse de données par similarité sur les contextes plurivalués numériques et les contextes plurivalués symboliques.

5.5.1 Contexte plurivalué numérique

Un contexte plurivalué numérique est un contexte plurivalué où les valeurs des attributs peuvent être des ensembles de valeurs réelles. Ces ensembles de valeurs peuvent être des intervalles ou des ensembles discrets. Un exemple de tel contexte est donné à la table 5.8.

Table 5.8 – Un extrait des mesures (approximatives) des caractéristiques relatives au pesticide appelé *glyphosate*. Ces mesures sont données par différentes sources qui ont étudié cette matière active. La dernière ligne du tableau représente les valeurs données par les experts de l'INRA Colmar. DT50 : *temps de demi vie au champ*, Aquatox : *toxicité* et DJA : *dose journalière admise*.

Variable / Base	DT50 (jour)	Aquatox (mg/l)	DJA (mg/l)
PM10	[3,60]	[86,120]	0.3
INRA	[38,60]	[38,78]	0.05
Com98	[38,60]	15	
Agritox 2002	[8,30]	0.64	0.3
Expertise	32	15	0.3

L'application de l'approche d'analyse de données par similarité détaillée à la section 5.3 à un tel contexte plurivalué est directe. Seule la définition du partage d'attributs entre objets, qui repose sur la similarité entre les valeurs des attributs, doit être adaptée pour prendre en compte la similarité entre deux ensembles de valeurs. En s'appuyant sur la définition du partage d'attributs multivalués (définition 35), on définit le partage d'attributs plurivalués comme suit.

Définition 52 *Soit un contexte plurivalué numérique* (G, M, W, I).

1. *Soient deux objets* g_i *et* g_j *dans* G *et un attribut* m *dans* M *tels que* $m(g_i) \neq \varnothing$ *et* $m(g_j) \neq \varnothing$. *On dit que les valeurs dans* $m(g_i)$ *et* $m(g_j)$ *sont similaires et on note* $m(g_i) \simeq m(g_j)$ *si et seulement si* $\forall w_i \in m(g_i)$ *et* $w_j \in m(g_j)$, $|w_i - w_j| \leq \theta_m$ *(où* θ_m *désigne le seuil de similarité choisi pour* m*).*
2. *On dit que deux objets* g_i *et* g_j *partagent un attribut* $m \in M$ *si et seulement si* $m(g_i) \simeq m(g_j)$. *Dans ce cas, on dit que* g_i *et* g_j *partagent* $m_{\{m(g_i) \cup m(g_j)\}}$. *Lorsque* $m(g_i)$ *et* $m(g_j)$ *sont sous la forme d'intervalles* $(m(g_i) = [\alpha_i, \beta_i]$ *et* $m(g_j) = [\alpha_j, \beta_j]$, g_i *et* g_j *partagent* $m_{[min(\alpha_i,\alpha_j),max(\beta_i,\beta_j)]}$ *lorsque* $max(\beta_i, \beta_j) - min(\alpha_i, \alpha_j) \leq \theta_m$.
3. *De manière générale, on dit qu'un ensemble d'objets* $A \subseteq G$ *partage un attribut* $m \in M$ *si et seulement si toute paire d'objets de* A *partage* m. *Dans ce cas, on dit que les objets dans* A *partagent* $m_{\{m(g_i),\ g_i \in A\}}$. *Lorsque* $m(g_i) = [\alpha_i, \beta_i]$ *pour* $g_i \in A$, *on dit que les objets dans* A *partagent* $m_{[min(\alpha_i),max(\beta_i)]}$.

Étant donné cette définition de partage d'attributs plurivalués, l'application des opérateurs de dérivation introduits dans la définition 37 aux contextes plurivalués numérique est directe. La réécriture de ces opérateurs dans les conditions des contextes plurivalués est donnée dans la définition suivante.

Définition 53 *Soit un contexte plurivalué numérique* (G, M, W, I). *Pour tout ensemble d'objets* $A \in \mathfrak{P}(G)$,

$$A^{\uparrow_\theta} = \{m_{[\alpha,\beta]} \in M \times \mathfrak{I}_\theta \text{ tel que } \forall g_i, g_j \in A,\ m(g_i) \simeq m(g_j)\}$$

est l'ensemble d'attributs dans $M \times \mathfrak{I}_\theta$ *communs à tous les objets de* A. α *étant le minimum des bornes inférieures des intervalles* $[\alpha_i, \beta_i] = m(g_i)$, $g_i \in A$ *et* β *étant le maximum des bornes supérieures de ces intervalles.*

De façon duale, pour tout ensemble d'attributs $B \in \mathfrak{P}(M \times \mathfrak{I}_\theta)$,

$$B^{\downarrow_\theta} = \{g \in G \text{ tel que } \forall m_{[\alpha,\beta]} \in B, m(g) \subseteq [\alpha, \beta]\}$$

est l'ensemble d'objets dans G *qui partagent tous les attributs dans* B.

Le théorème 4 est applicable aux opérateurs de dérivation "$\uparrow_\theta$" et "$\downarrow_\theta$" dans le cas des contextes plurivalués numériques. La connexion de Galois formée par ces opérateurs permet de déterminer les concepts plurivalués et les treillis de concepts plurivalués à partir d'un contexte plurivalué numérique qui étendent respectivement les concepts multivalués et les treillis de concepts multivalués correspondant à un contexte multivalué numérique.

Le treillis de concepts plurivalués correspondant au contexte plurivalué numérique donné dans la table 5.8 pour les seuils $\theta_{DT50} = 30$, $\theta_{Aquatox} = 100$ et $\theta_{DJA} = 0.1$ est donné à la figure 5.14.

Les treillis de concepts plurivalués ne diffèrent pas des treillis de concepts multivalués. De ce fait, les aspects relatifs aux treillis de concepts multivalués discutés dans les sections précédentes (granularité des concepts, évolution du treillis en fonction des seuils de similarité) restent valables dans le cas des treillis de concepts plurivalués.

5.5.2 Contexte plurivalué symbolique

Un contexte plurivalué symbolique est un contexte plurivalué où les valeurs des attributs peuvent être des ensembles de valeurs symboliques. Un exemple de contexte sous cette forme est donné dans la table 5.9.

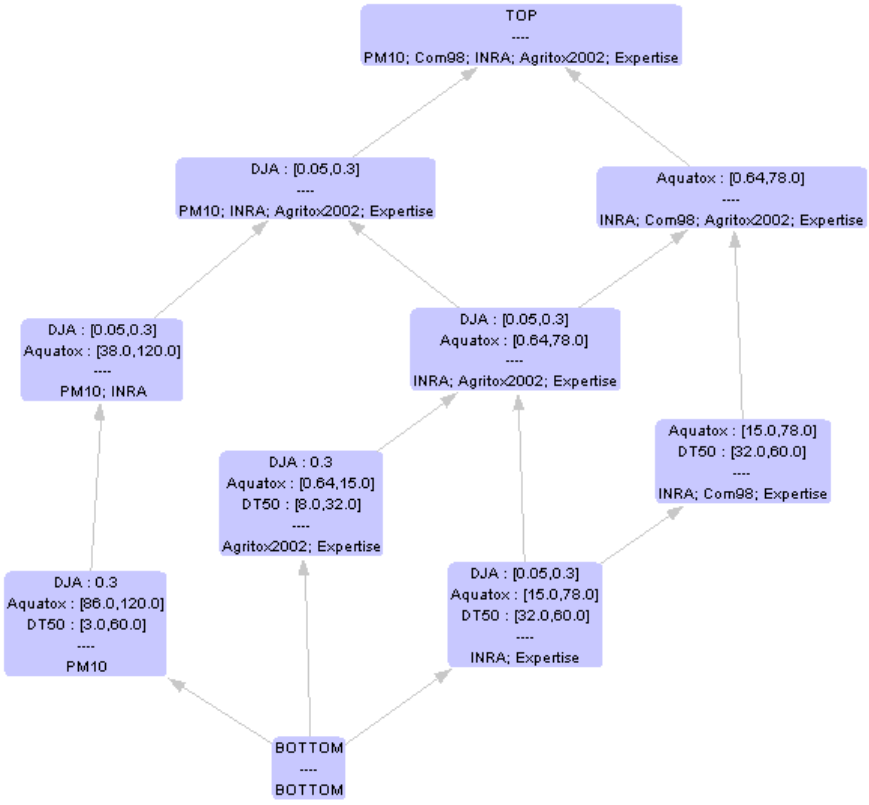


FIGURE 5.14 – Le treillis de concepts plurivalués correspondant au contexte plurivalué numérique donné à la table 5.8 pour les seuils $\theta_{DT50} = 30$, $\theta_{Aquatox} = 100$ et $\theta_{DJA} = 0.1$.

TABLE 5.9 – Un exemple de contexte plurivalué symbolique représentant une partie des données de BioRegistry

	Organism	Content Quality	Ontology
DB_1	Amphibians, Fishes	Complete	NCBI
DB_2	Amphibians, Fishes	Complete, Updated	NCBI
DB_3	Amphibians, Mammals		NCBI
DB_4	Birds, Mammals	Updated	
DB_5	Amphibians, Mammals	Complete, Updated	
DB_6	Birds, Mammals	Complete	GO
DB_7	Birds, Mammals	Complete, Updated	GO, NCBI
DB_8	Birds, Mammals		NCBI

De la même manière que pour les contextes plurivalués numériques, l'extension des définitions et résultats de l'analyse de données par similarité introduits dans le cas des contextes multivalués symboliques nécessite l'extension du calcul de la similarité entre paires de valeurs (cas multivalué) à des paires d'ensembles de valeurs (cas plurivalué). En considérant la similarité entre les ensembles de valeurs des attributs plurivalués dans le contexte, la définition de partage d'attributs dans les contextes multivalués symboliques est généralisée aux contextes plurivalués comme suit.

Définition 54 *Soit un contexte plurivalué symbolique* (G, M, W, I).

1. *Deux objets* g_i *et* g_j *partagent un attribut* $m \in M$ *si et seulement si* $m(g_i) \simeq m(g_j)$. *Dans ce cas,* g_i *et* g_j *partagent* $m_{|\, m(g_i) \cup m(g_j)}$.
2. *De manière générale, un ensemble d'objets* $A \subseteq G$ *partage un attribut* $m \in M$ *si et seulement si toute paire d'objets de* A *partage* m. *Dans ce cas, on dit que les objets dans* A *partagent* $m_{|\, \bigcup_{g_i \in A} m(g_i)}$.

Les opérateurs de dérivation introduits dans la définition 46 sont généralisés aux contextes plurivalués comme suit.

Définition 55 (Opérateurs de dérivation) *Soit un contexte symbolique* (G, M, W, I),

– *Pour tout ensemble d'objets* $A \subseteq G$,

$$A^{\uparrow_S} = \{m_{|\, \bigcup_{g \in A} m(g)} \in M \times \mathfrak{P}(W) \text{ tel que } \forall g_i, g_j \in A, m(g_i) \simeq m(g_j)\}$$

est l'ensemble de tous les attributs de $M \times \mathfrak{P}(W)$ *partagés par tous les objets de* A.

– *De manière duale, pout tout ensemble d'attributs* $B \subseteq M \times \mathfrak{P}(W)$,

$$B^{\downarrow_S} = \{g \in G \text{ tel que } \forall\, m_{|\, W_m} \in B, m(g) \simeq W_m\}$$

est l'ensemble de tous les objects de G *ayant les attributs de* B.

5.5.3 Contexte hétérogène

Nous appelons contexte hétérogène un contexte formel sous sa forme la plus générale où il peut y avoir des attributs binaires et des attributs (multi/pluri)valués à valeurs numériques ou symboliques. L'application de l'analyse de données par similarité à un contexte hétérogène nécessite tout d'abord la définition d'une fonction de similarité adaptée à chaque type d'attribut. Ces fonctions de similarité sont ensuite utilisées dans la définition du partage d'attributs entre objets ainsi que des opérateurs de dérivation permettant ainsi de déterminer l'ensemble des concepts à partir d'un contexte hétérogène et de les organiser en treillis de concepts.

Chapitre 6

Découverte de ressources par treillis de concepts multivalués

Sommaire

6.1 Introduction 113
6.2 Classification à granularité variable 113
6.3 Navigation 114
6.3.1 Navigation statique 114
6.3.2 Navigation dynamique : les zooms avant/arrière dans les treillis de concepts multivalués 114
6.4 Interrogation 116
6.4.1 Les requêtes multivaluées 116
6.4.2 Pertinence des objets par rapport à une requête multivaluée 118
6.4.3 Stratégie de recherche des objets pertinents 119
6.4.4 Choix du treillis de concepts multivalués à interroger 120
6.4.5 Illustration dans le cas de contextes numériques 121
6.4.6 Illustration dans le cas de contextes symboliques 124

6.1 Introduction

L'approche d'analyse de données complexes par similarité définie dans le chapitre précédente permet d'obtenir des treillis de concepts multivalués à partir de contextes multivalués. Ces treillis peuvent être utilisés dans plusieurs applications qui s'appuient sur la notion de granularité des concepts multivalués. Dans la suite de chapitre nous nous nous intéressons à la découverte de ressources par treillis de concepts multivalués. Nous détaillerons, dans la section 6.2, le potentiel des treillis de concepts multivalués pour la classification à granularité variable. Dans la section 6.3, nous nous intéressons à la navigation dans un treillis de concepts multivalués. Dans la section 6.4, nous détaillons l'interrogation d'un treillis de concepts multivalués.

6.2 Classification à granularité variable

La motivation de l'utilisation des treillis de concepts comme outil de classification est renforcée dans le cas des treillis de concepts multivalués par deux aspects majeurs. Le premier aspect

est la possibilité d'effectuer une classification directe à partir de données complexes et hétérogènes qui peuvent être présentées de manière naturelle sous la forme de contextes multivalués (ou plurivalués) sans avoir à les transformer en contextes binaires. Le deuxième aspect est la possibilité de produire pour un même ensemble de données complexes plusieurs treillis à différents niveaux de précision (d'où l'appellation *classification à granularité variable*). La notion de granularité d'un concept et de treillis de concepts multivalués (section 5.3.8 et section 5.4.5) et l'évolution des treillis de concepts multivalués en fonction des seuils de similarité choisis (section 5.3.9 et section 5.4.5) sont parmi les points clés de cette méthode. En plus de l'obtention du treillis correspondant aux attentes exprimées en terme de seuils de similarité, la classification à granularité variable permet d'effectuer une classification progressive commençant par un treillis de concepts multivalués à granularité large et en générant ensuite des treillis de concepts multivalués à granularité plus fine en fonction du besoin en précision. Cette classification progressive est particulièrement intéressante dans le cas de données volumineuses où il est important d'avoir d'abord une idée globale sur l'organisation des données pour effectuer par la suite une classification plus précise avec un choix approprié des seuils de similarité. Un intérêt supplémentaire de la classification progressive est d'observer l'évolution en fonction de la variation de la similarité de certains attributs plutôt que d'autre. Ceci correspond à fixer le seuil de similarité de certains attributs et de faire varier progressivement le seuil d'autres attributs. De cette manière l'évolution de la classification correspond aux préférences qu'on pourrait accorder à certain attributs dans le contexte multivalué considéré.

6.3 Navigation

Nous distinguons deux modes de navigation dans les treillis de concepts multivalués : la navigation statique et la navigation dynamique.

6.3.1 Navigation statique

Nous appelons navigation statique la navigation dans un seul treillis de concepts multivalués. Elle s'effectue comme dans la navigation classique où il s'agit de suivre les liens de spécialisation et/ou de généralisation entre les concepts. La différence est la possibilité de choisir le niveau de précision du treillis de concepts multivalués à explorer avant de procéder la navigation. Le changement de la précision du treillis à explorer dans le cadre d'une navigation statique signifie la reprise à zéro de la navigation dans le nouveau treillis choisi.

6.3.2 Navigation dynamique : les zooms avant/arrière dans les treillis de concepts multivalués

Nous appelons navigation dynamique la navigation dans plusieurs treillis ou parties de treillis de concepts multivalués à la fois. Le processus de navigation dynamique consiste à parcourir les concepts multivalués d'un treillis de concepts multivalués d'un niveau de précision donné en suivant les liens de spécialisation/généralisation entre ces concepts et à effectuer ensuite, en fonction du besoin de précision, des zooms avant ou arrière pour avoir une idée plus précise sur l'organisation des objets dans le concept concerné par le zoom. Le zoom avant d'un concept multivalué consiste à passer dans la partie correspondante à ce concept dans un treillis de concepts multivalués plus précis. Un zoom avant sur un concept donné permet de montrer comment les objets dans l'extension du concept courant sont organisés lorsqu'on augmente les contraintes de similarité entre les valeurs des attributs du contexte. Réciproquement, le zoom arrière consiste à

passer à la partie correspondante au concept courant dans un treillis de concepts multivalués plus général. Un zoom arrière permet de donner une idée sur la classe générale à laquelle appartiennent les objets dans l'extension du concepts en cours lorsqu'on relâche les contraintes de similarité entre les valeurs des attributs du contexte. Dans les deux cas, le processus de navigation n'est pas repris à zéro lors du changement de la précision du treillis.

L'illustration du zoom avant/arrière dans le cadre d'un processus de navigation dynamique est donnée à la figure 6.1. Les treillis considérés sont $\underline{\mathfrak{B}}_3(G, M, W, I)$ (à droite) et $\underline{\mathfrak{B}}_2(G, M, W, I)$ (à gauche) correspondant au contexte multivalué symbolique donné à la table 5.6. Le zoom avant est donné dans le cadre d'une navigation dynamique dans le treillis $\underline{\mathfrak{B}}_3(G, M, W, I)$ pour le concept multivalué $(\{Exint, SpliceDB, TRANSCompel\}, \{NCBI_{|\{Vertebarates,Eukaryotes,Mammals\}}, NAR_{|\{1.2\}}\})$. Le zoom avant d'un niveau de ce concept montre la répartition des objets dans son extension sur trois concepts multivalués ainsi que l'organisation hiérarchique de ces concepts. Cette opération est illustrée à la figure 6.1 de droite à gauche.

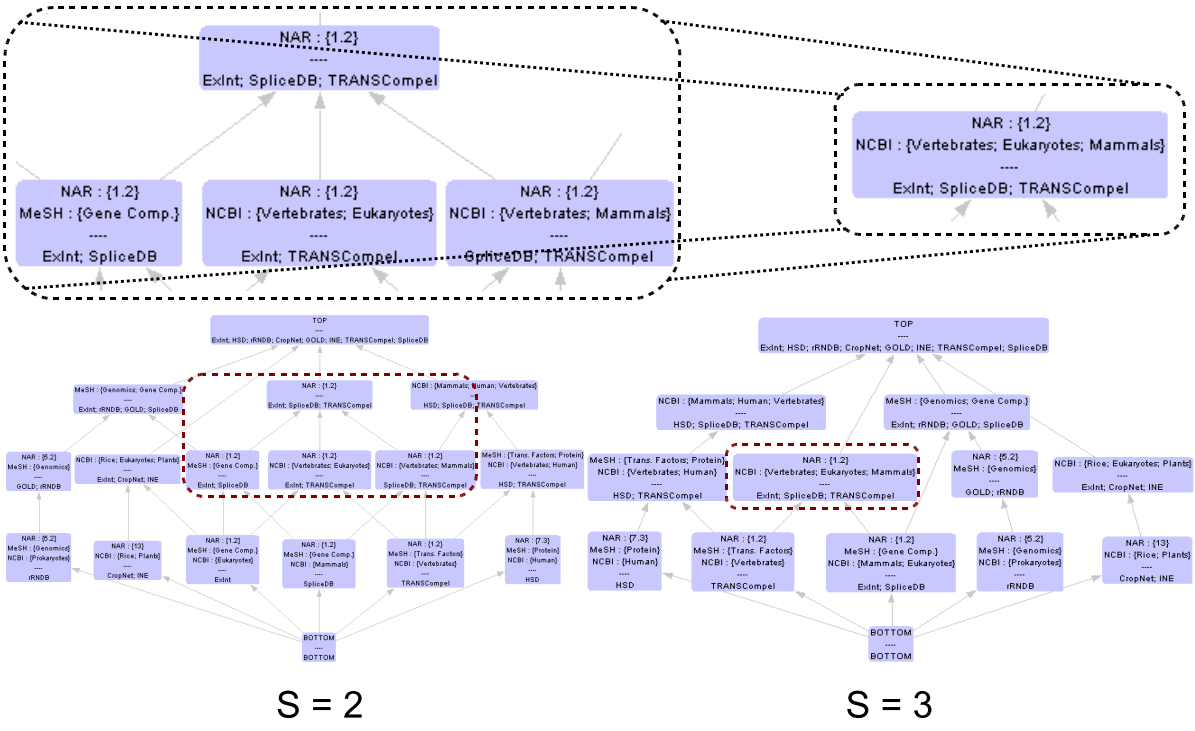


FIGURE 6.1 – Un exemple de zoom avant/arrière des concepts multivalués dans un processus de navigation dynamique.

Réciproquement, le zoom arrière est illustré à la figure 6.1 (de gauche à droite) dans le cadre de la navigation dynamique dans le treillis de concepts multivalués $\underline{\mathfrak{B}}_2(G, M, W, I)$ pour l'un des quatre concepts $(\{Exint, SpliceDB, TRANSCompel\}, \{NAR_{|\{1.2\}}\})$ ou $(\{Exint, TRANSCompel\}, \{NCBI_{|\{Vertebarates,Eukaryotes\}}, NAR_{|\{1.2\}}\})$ ou $(\{SpliceDB, TRANSCompel\}, \{NCBI_{|\{Vertebarates,Mammals\}}, NAR_{|\{1.2\}}\})$ ou $(\{Exint, SpliceDB\}, \{MeSH_{|\{GeneComponents\}}, NAR_{|\{1.2\}}\})$.

Un autre exemple illustrant les opération de zoom avant/arrière est donné à la figure 5.6 dans

le cas du contexte multivalué numérique donné en table 5.3.

Dans le cadre d'un processus de navigation dynamique, plusieurs treillis (ou parties de treillis) de concepts multivalués sont considéré(e)s à la fois. Ceci nécessite la construction préalable de ces treillis de concepts multivalués. Dans la pratique cette solution peut être gérable dans le cas de contextes relativement petits. Cependant, dans le cadre d'applications réelles avec des contextes de grandes tailles, la construction de tous les treillis de concepts multivalués d'un contexte multivalués peut d'une part être coûteuse en terme de temps de calcul et d'espace mémoire et d'autre part affecter les performances du processus de navigation. Une meilleure façon d'assurer la navigation dynamique tout en en minimisant les calculs consiste à considérer un seul treillis global d'un niveau de précision choisi et de répondre aux zooms par la construction dynamique de "petits" treillis correspondant seulement au lignes du contextes correspondantes aux objets dans l'extension du concept sur lequel le zoom est effectué.

Dans l'exemple donné à la figure 6.1, le zoom avant du concept ($\{$*Exint, SpliceDB, TRANSCompel*$\}$, $\{NCBI_{|\{Vertebarates,Eukaryotes,Mammals\}}, NAR_{|\{1.2\}}\}$) correspond à la construction du treillis de concepts multivalués pour le contexte multivalué réduit donné à la table 6.1. Le treillis correspondant à ce contexte est donné à la figure 6.2.

TABLE 6.1 – Le contexte multivalué réduit aux objets du concept multivalué ($\{$*Exint, SpliceDB, TRANSCompel*$\}$, $\{NCBI_{|\{Vertebarates,Eukaryotes,Mammals\}}, NAR_{|\{1.2\}}\}$).

	NCBI	MeSH	NAR
ExInt	Eukaryotes	Gene components	1.2
SpliceDB	Mammals	Gene components	1.2
TRANSCompel	Vertebrates	Transcription factors	1.2

6.4 Interrogation

La découverte de ressources par interrogation d'un treillis de concepts multivalués est une extension de l'approche définit à la section 4.3.2 qui s'appuie sur les treillis de concepts formels. Elle offre plus de flexibilité dans la définition des requêtes d'une part et plus de précision dans l'identification et le classement des objets pertinents pour ces requêtes d'autre part. En effet, des fonctions de pertinence sont définies en s'appuyant sur les connexions de Galois par similarité définies dans les sections précédentes.

Dans le cas de l'interrogation d'un treillis de concepts formels, une requête est définie par un ensemble d'attributs (binaires). Une telle requête n'est pas toujours suffisante pour décrire de façon précise les objets à retrouver dans le cas des contextes multivalués. En effet, un attribut peut prendre des valeurs très différentes. De ce fait, il est indispensable de donner des précisions supplémentaires sur les attributs.

6.4.1 Les requêtes multivaluées

La définition de la forme générale des requêtes dans un processus d'interrogation de treillis de concepts multivalués s'appuie sur les trois idées suivantes :

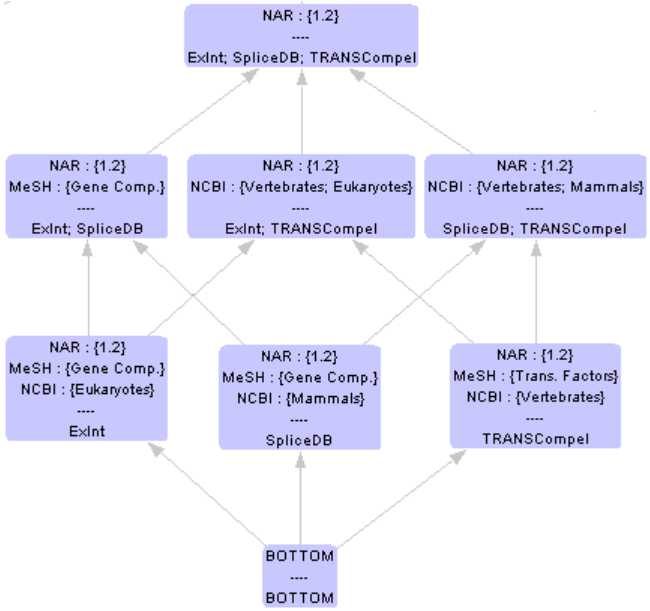


FIGURE 6.2 – Le treillis de concepts multivalués correspondant au contexte réduit donné à la table 6.1 pour un seuil de similarité $S = 2$.

- Une requête est sensée être la description d'un "objet idéal" à trouver. Et comme un objet est décrit par un ensemble d'attributs valués dans un contexte multivalué, la requête doit respecter cette forme particulière.
- Par analogie avec l'interrogation des treillis de concepts formels, un concept multivalué peut être considéré comme une classe d'objets (l'extension du concept) pertinents pour la requête formée par la conjonction des attributs valués constituant l'intension du concept. De ce fait, une requête doit aussi respecter la forme générale des intensions des concepts multivalués.
- Puisque l'interrogation de treillis de concepts multivalués est une généralisation de l'interrogation de treillis de concepts formels, la définition de requêtes sur un treillis de concepts multivalués doit couvrir naturellement les requêtes définies sur un treillis de concepts formels.

La combinaison de ces trois formes nous amène à définir les requêtes dans le cas de l'interrogation de treillis de concepts multivalués comme suit.

Définition 56 (Requête multivaluée) *Soit un contexte multivalué (G, M, W, I). Une requête multivaluée est un ensemble $Y \subseteq M \times \mathfrak{P}(W)$ d'attributs multivalués. Soit $y \in Y$ un attribut de la requête multivaluée. y peut avoir l'une des trois formes suivantes :*

1. *$y = m \in M$: y est un attribut sans restriction sur la valeur qu'il peut prendre. Ce cas correspond à la définition de requêtes dans le cas de l'interrogation de treillis de concepts formels. Cette forme de requête correspond à la recherche des objets pour lesquels l'attribut m a une valeur non vide dans le contexte (G, M, W, I).*
2. *$y = m_{|\{w\}} \in M \times W$: y est un attribut accompagné d'une valeur. La valeur w est utilisée pour orienter la recherche aux objets pour lesquels l'attribut m prend une valeur égale ou similaire à w dans le contexte (G, M, W, I).*

3. $y = m_{|W_m} \in M \times \mathfrak{P}(W)$: *y est un attribut accompagné d'un ensemble de valeurs de précision. L'ensemble de valeurs W_m indique que les objets à retrouver sont ceux pour lesquels l'attribut m prend une valeur dans W_m ou similaire à celles dans W_m dans le contexte (G, M, W, I).*

La définition précédente donne un cadre général pour les requêtes dans un processus d'interrogation de treillis de concepts multivalués. Elle peut être raffinée en respectant la particularité de certains contextes multivalués (numérique, symbolique, etc.). De ce fait, les illustrations seront données dans les sections qui suivent avec des exemples de contextes spécifiques.

6.4.2 Pertinence des objets par rapport à une requête multivaluée

Dans le cas de l'interrogation d'un treillis de concepts formels, la définition de base de la pertinence des objets du contexte s'appuie sur le calcul du nombre d'attributs partagés avec la requête considérée. Un objet est pertinent lorsqu'il partage au moins un attribut avec la requête. Plus un objet partage d'attributs avec la requête plus il est pertinent. Cette définition est étendue dans l'interrogation de treillis de concepts multivalués. Intuitivement, un objets est pertinent s'il possède des attributs de la requête avec des valeurs similaires à celles données dans la requête. Si l'on considère la définition de partage d'attribut dans un contexte multivalué, un objet est pertinent s'il partage au moins un attribut avec la requête multivaluée considérée. Formellement, nous définissons la pertinence d'un objet par rapport à une requête multivaluée comme suit.

Définition 57 (Pertinence) *Soient (G, M, W, I) un contexte multivalué et $Y \subseteq M \times \mathfrak{P}(W)$ une requête multivaluée. Un objet $g \in G$ est pertinent pour la requête Y si et seulement si $\{g\}^{\uparrow} \cap Y \neq \varnothing$.*

L'opérateur "$\uparrow$" utilisé dans la définition précédente est un opérateur de dérivation générique sur les contextes multivalués. Il correspond à "$\uparrow_\theta$" dans le cas des contextes numériques ou à "$\uparrow_S$" dans le cas des contextes symboliques, etc. De même pour l'opérateur d'intersection "$\cap$" qui dénote l'intersection entre sous ensembles de $M \times \mathfrak{P}(W)$.

La définition du partage d'un attribut y de la requête par g dépend de la forme de y. Conformément à la définition de requête multivaluée, on distingue trois cas possibles pour chaque attribut $y \in Y$:

1. Si $y = m \in M$ alors y est partagé par un objet $g \in G$ si et seulement si la valeur de prise par l'attribut m pour g dans le contexte (G, M, W, I) est non vide (i.e. $m(g) \neq \varnothing$).
2. Si $y = m_{|\{w\}} \in M \times W$ alors y est partagé par $g \in G$ si et seulement si la valeur prise par m pour g est similaire à la valeur w donné dans la requête (i.e. $m(g) \simeq w$).
3. Si $y = m_{|W_m} \in M \times \mathfrak{P}(W)$ alors y est partagé par $g \in G$ si et seulement si la valeur prise par m pour g est similaire aux valeurs données dans W_m (i.e. $\forall w \in W_m\ m(g) \simeq w$).

Dans les conditions 2 et 3 énumérées précédemment, on ne considère pas de définition particulière de la similarité entre les valeurs des attributs. De cette manière, ces conditions sont applicables dans l'interrogation de treillis de concepts multivalués correspondant à un contexte multivalué indépendamment de la nature des données qu'il contient (numériques, symboliques, plurivalués et hétérogènes).

6.4.3 Stratégie de recherche des objets pertinents

L'approche d'interrogation de treillis de concepts formels détaillée dans la section 4.3.2 du chapitre précédent, qui s'appuie sur l'algorithme BR-Explorer, peut être adaptée à l'interrogation de treillis de concepts multivalués.

Après construction du treillis de concepts multivalués, l'interrogation s'effectue en suivant les étapes suivantes :

1. Définition d'une requête multivaluée.
2. Insertion de la requête dans le treillis de concepts multivalués.
3. Localisation de la requête dans le treillis de concepts multivalués obtenu.
4. Recherche des sources de données pertinentes dans le voisinage de la requête dans le treillis.
5. Présentation des objets pertinents classés selon leur pertinence pour la requête.

La définition de la requête consiste à spécifier une requête multivaluée décrivant les objets à retrouver. L'insertion de la requête dans le treillis de concepts multivalués s'effectue de la même manière que dans l'algorithme BR-Explorer. Il est donc nécessaire de définir le concept requête multivaluée qui sera inséré dans le treillis de concepts multivalués. Par analogie avec le cas classique (définition 23), on définit le concept requête multivaluée comme suit.

Définition 58 (Concept requête multivaluée) *Un concept requête multivaluée Q correspondant à une requête multivaluée Y est le couple (X, Y) où X est l'ensemble formé par un objet virtuel x supposé partager tous les attributs de Y (i.e. $X^{\uparrow} = Y$).*

Après l'insertion du concept requête multivaluée (X, Y) dans le treillis, (X, Y) donne naissance au concept *pivot* défini par $P = (Y^{\uparrow\downarrow}, Y)$ de manière analogue au concept pivot dans BR-Explorer (définition 25). Une fois que le pivot est localisé dans le treillis, la recherche des objets pertinents commence en considérant le pivot puis de manière récursives ses subsumants jusqu'au concept le plus général du treillis. À chaque étape, les objets émergeants sont ajoutés au résultat dans le rang approprié. Un premier classement des objets est déduit directement à partir du treillis, qui s'appuie sur le nombre d'attributs partagés avec la requête. Bien qu'adéquat, ce classement peut être amélioré par le calcul de la proximité des objets à la requête, qui est fonction de la similarité entre les valeurs des attributs partagés entre la requête et les objets pertinents.

Pour formaliser la notion de proximité entre les objets et les requêtes on définit une fonction de similarité, qu'on notera par Sim, permettant de calculer la similarité entre deux valeurs w_1 et w_2 d'un même attribut m et de renvoyer une valeur $Sim(w_1, w_2) \in [0, 1]$ indiquant à quel point w_1 et w_2 sont similaires. $Sim(w_1, w_2) = 1$ lorsque $w_1 = w_2$ (similarité maximale) et $Sim(w_1, w_2) = 0$ lorsque w_1 et w_2 ne sont pas de tout similaires. En s'appuyant sur la fonction Sim, on définit la distance, qu'on notera par $dist$, d'un objet g à un attribut de la requête, $y \in Y$, selon la forme de y comme suit.

Définition 59 (Distance entre un objet à un attribut de la requête) *Soient (G, M, W, I) un contexte multivalué, Y une requête multivaluée, $g \in G$ et $y \in Y$.*

– *Si $y = m$ alors*

$$dist(g, y) = 0.$$

Cette valeur est justifiée par le fait que toute valeur non vide prise par m est acceptée lorsque l'attribut de la requête n'est pas accompagné d'une valeur de précision.

– *Si* $y = m_{|\{w\}}$ *alors*

$$dist(g, y) = 1 - Sim(m(g), w).$$

Lorsque la similarité entre la valeur de l'attribut dans la requête, w, et celle prise pour l'objet g, $m(g)$ est maximale, la distance de g à y est nulle.

– *Si* $y = m_{|W_m}$ *alors*

$$dist(g, y) = 1 - max_{w \in W_m}(Sim(m(g), w)).$$

Dans le cas où l'attribut de la requête est accompagné par plusieurs valeurs de précision, on considère la valeur la plus proche de celle prise pour l'objet g dans le contexte. Lorsque $m(g) \in W_m$, la distance est nulle ce qui veut dire que l'objet correspond parfaitement au critère de la requête exprimé par l'attribut considéré.

À partir du calcul des distances d'un objet aux attributs de la requête, il est possible de savoir à quel point cet objet est pertinent. Plus l'objet est proche des attributs de la requête plus il est pertinent. Cette condition s'ajoute à la condition de pertinence définie dans le cas classique à savoir le partage d'attributs avec la requête. De ce fait, le calcul de la distance entre l'objet et les attributs d'une requête se limite aux attributs partagés entre les deux. Le calcul de la proximité entre les objets et les requêtes nous conduit à définir de manière précise le degré de pertinence d'un objet à une requête comme suit.

Définition 60 (Degré de pertinence d'un objet pour une requête) *Soient un contexte multivalué (G, M, W, I) , une requête multivaluée $Y = \{y_1, ..., y_n\}$ représentée par le concept requête multivaluée $Q = (X, Y)$ et un objet $g \in G$ qui partage p attributs avec Y. Le degré de pertinence de g pour Q, noté par $\mathcal{P}(g, Q)$, est donné par :*

$$\mathcal{P}(g, Q) = \frac{1}{n} \sum_{i=1}^{p} (1 - dist(g, y_i))$$

La définition précédente tient compte à la fois du nombre d'attributs partagés avec la requête et de la similarité entre les valeurs des attributs partagés. La pertinence est maximale (égale à 1) lorsque l'objet possède tous les attributs de la requête et est à une distance nulle de chacun des ces attributs ($p = n$ et $dist(g, y_i) = 0$). Dans le cas où l'objet considéré ne partage aucun attribut avec la requête, son degré de pertinence est égale à 0.

Dans un processus d'interrogation de treillis de concepts multivalués, le calcul des degrés de pertinence des objets s'effectue après avoir identifié les objets pertinents pour la requête, les super-concepts du concept pivot. De cette manière, ce calcul n'est effectué que pour des objets qui partagent des attributs avec la requête multivaluée dans le but de fournir un classement plus précis des objets pertinents. L'illustration de l'apport de ce calcul sur le classement des objets pertinents à une requête multivaluée est détaillée dans les deux sections qui suivent.

6.4.4 Choix du treillis de concepts multivalués à interroger

La découverte de ressources par interrogation de treillis de concepts multivalués peut bénéficier de la possibilité de la construction de treillis de concepts multivalués de granularités variables. En effet, le niveau de granularité des concepts du treillis considéré est en relation étroite avec la condition de pertinence des objets et par la suite avec la réponse obtenue. Dans le cas ou les concepts sont de granularité fine, la condition de partage d'attributs entre objets

est strict. Et comme le partage d'attributs de la requête reprend la même définition, il est à son tour stricte. Ceci implique que seul les objets les plus proches de la requête (définitions 59 et 60) seront pertinents. Dans le cas où les concepts sont de granularité plus large, les attributs de la requête sont partagés par un nombre plus élevé d'objets mais de degrés de pertinence plus faible.

6.4.5 Illustration dans le cas de contextes numériques

Dans cette section nous donnons une illustration de l'interrogation d'un treillis de concepts multivalués pour un contexte numérique. Soit le contexte numérique (G, M, W, I) représentant les mesures relatives aux planètes du système solaire donné dans la table 5.3. Un exemple d'interrogation pourrait consister à "*chercher la (les) planètes ayant un diamètre d'environ 12 milles km, une masse d'environ 50* 10^{23} *kg et à une distance au soleil d'environ 150 millions km*". Ceci se traduit par la requête $Y = \{Diamètre_{12000},\ Masse_{50\ 10^{23}},\ Distance\ au\ soleil_{150M}\}$. Cette requête est transformée pour être conforme aux données originales relatives aux planètes (table 5.1). La représentation équivalente de cette requête est $Y = \{D_{0.94},\ M_{0.83},\ DS_{0.65}\}$.

La requête est alors représentée par un concept requête multivaluée qui est ensuite inséré dans le treillis de concepts multivalués. Soit le treillis $\underline{\mathfrak{B}}_{0.1}(G, M, W, I)$ obtenu pour un seuil $\theta = 0.1$ (figure 6.3). L'insertion du concept requête multivaluée (X, Y) dans $\underline{\mathfrak{B}}_{0.1}(G, M, W, I)$ produit un

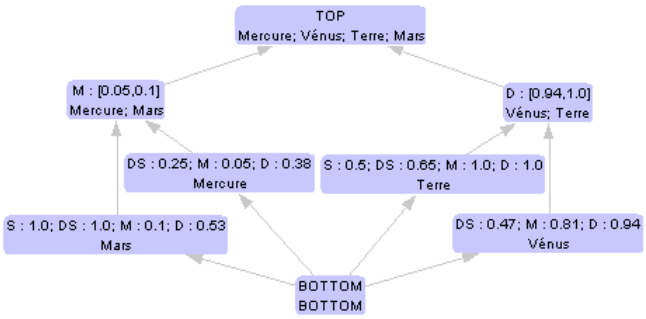


FIGURE 6.3 – Treillis de concepts multivalués $\underline{\mathfrak{B}}_{0.1}(G, M, W, I)$.

nouveau treillis (figure 6.4).

D'après la définition de la pertinence (définition 57), les objets pertinents pour la requête considérée sont les objets $g \in G$ vérifiant $\{g\}^{\uparrow_\theta} \cap Y \neq \varnothing$ où $\theta = 0.1$ en conformité avec le treillis $\underline{\mathfrak{B}}_{0.1}(G, M, W, I)$. Après l'insertion de la requête dans le treillis, la recherche des objets pertinents consiste d'abord à localiser le concept pivot dans le treillis. Ensuite la réponse est construite en ajoutant de manière récursive les objets constituant les extensions du concept pivot et de ses super-concepts. L'algorithme BR-Explorer détaillé dans le chapitre précédent est applicable ici. La modification à apporter est la comparaison entre intensions de concepts multivalués au lieu des intensions de concepts formels.

Le concept pivot correspondant à la requête $Y = \{D_{0.94}, M_{0.83}, DS_{0.65}\}$ est $(\{x\}, \{D_{0.94}, M_{0.83}, DS_{0.65}\})$. Ce concept est indiqué en rouge sur le treillis donné à la figure 6.5. L'extension de ce concept ne contient aucun objet du contexte : aucun objet de G ne partage tous les attributs de la requête. Les étapes suivantes consistent à considérer les super-concepts du concept pivot. Les super-concepts du pivot sont indiqués en jaune à la figure 6.5. Tout d'abord les super-concepts directs du pivot sont considérés : $C_1 = (\{Vénus, x\}, \{D_{0.94}, M_{[0.81,0.83]}\})$ et $C_2 = (\{Terre, x\}, \{D_{[0.94,1]}, DS_{0.65}\})$. À cette étape, les objets *Vénus* et *Terre* sont ajoutés à la réponse. L'étape suivante consiste à considérer les super-concepts de C_1 et de C_2 : $C_3 =$

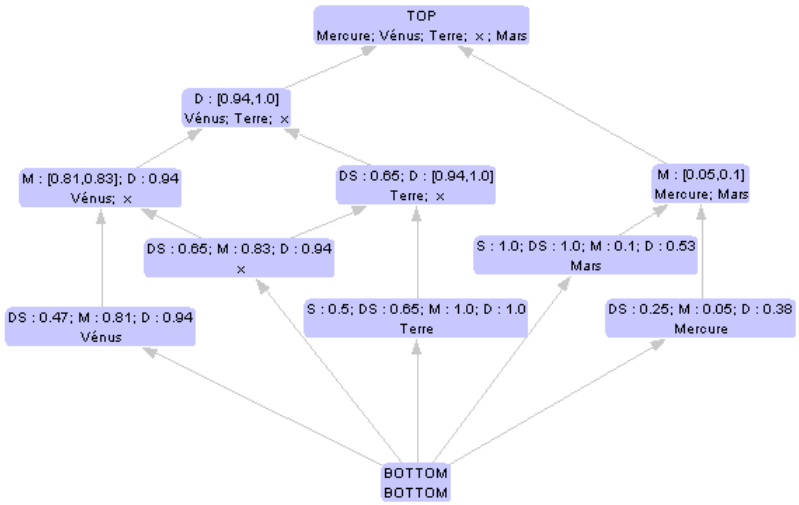


FIGURE 6.4 – Treillis de concepts multivalués résultant de l'insertion de la requête $Y = \{D_{0.94}, M_{0.83}, DS_{0.65}\}$ dans $\underline{\mathfrak{B}}_{0.1}(G, M, W, I)$.

$(\{Vénus, Terre, x\}, \{D_{[0.94,1]}\})$. L'extension de C_3 contient les objets *Vénus* et *Terre* qui ont déjà été ajoutés à la réponse. De ce fait, aucun objet n'est ajouté à la réponse. L'étape suivante consiste à considérer les super-concepts de C_3 : ici Top, qui a une intension nulle. Aucun objet n'est ajouté à la réponse et la recherche s'arrête.

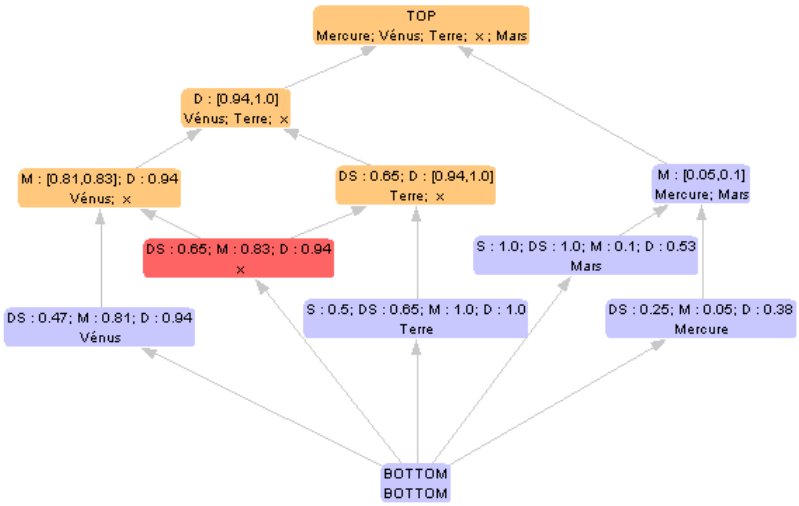


FIGURE 6.5 – Recherche des objets pertinents dans le treillis obtenu

Finalement le résultat retourné est :

1 - *Vénus* : partage $D_{0.94}$ et $M_{[0.81,0.83]}$ avec la requête.
- *Terre* : partage $D_{[0.94,1]}$ et $DS_{0.65}$ avec la requête.

Les deux objets, Vénus et Terre, sont classés au même rang dans la réponse parce qu'ils ont été retrouvés à la même étape. Ce classement peut être amélioré en considérant le degré de

pertinence de chacun des deux objets à la requête (définitions 57 et 60). Ici, deux valeurs sont similaires si leur différence est inférieur à un seuil θ donné. La similarité entre les valeurs d'un attribut se définit comme suit :

$$Sim(w_1, w_2) = 1 - |w_1 - w_2|.$$

La condition de partage d'attributs, "*la différence est inférieure à* θ", est équivalente à *la similarité est supérieure à* $1 - \theta$. Ainsi, les distances des objets, constituant la réponse, aux attributs de la requête sont données par le calcul suivant :

$$\begin{aligned}
dist(Vénus, D_{0.94}) &= 1 - Sim(D(Vénus), 0.94) \\
&= 1 - 1 + |0.94 - 0.94| \\
&= 0 \\
dist(Vénus, M_{0.83}) &= 1 - Sim(M(Vénus), 0.83) \\
&= 1 - 1 + |0.81 - 0.83| \\
&= 0.02 \\
dist(Terre, D_{0.94}) &= 1 - Sim(D(Terre), 0.94) \\
&= 1 - 1 + |1 - 0.94| \\
&= 0.06 \\
dist(Terre, DS_{0.65}) &= 1 - Sim(DS(Terre), 0.65) \\
&= 1 - 1 + |0.65 - 0.65| \\
&= 0
\end{aligned}$$

Les degrés de pertinence des objets Vénus et Terre à la requête $Y = \{D_{0.94}, M_{0.83}, DS_{0.65}\}$ sont obtenus en appliquant la formule de la définition 60 :

$$\mathcal{P}(Vénus, Q) = \frac{(1 + 0.98)}{3} = 0.66$$

$$\mathcal{P}(Terre, Q) = \frac{(0.94 + 1)}{3} = 0.64$$

En considérant ces degrés de pertinence, le classement est affiné comme suit :

1 - *Vénus* : partage $D_{0.94}$ et $M_{[0.81,0.83]}$ avec la requête (pertinence : 0.66).
2 - *Terre* : partage $D_{[0.94,1]}$ et $DS_{0.65}$ avec la requête (pertinence : 0.64).

Les planètes Vénus et Terre ont des valeurs proches de celles exprimées dans la requête pour les attributs Diamètre, Distance au soleil et Masse. La planète Vénus a un diamètre de 12 104 km et une masse de 48.7 10^{23} kg toutes les deux similaires aux valeurs souhaitées et exprimées dans la requête (12 000 km et 50 10^{23} kg respectivement). De la même manière, la planète Terre a un diamètre de 12 756 km et une distance au soleil de 150 million km. Ces deux valeurs sont similaires à celles exprimées dans la requête (12 000 km et 150 million km respectivement). Grâce à l'utilisation du treillis de concepts multivalués, l'appariement entre la requête et les objets pertinents ne nécessite pas l'identité des valeurs des attributs. Un objet ayant un attribut dont la valeur est différente de celle indiquée dans la requête peut être considéré comme pertinent. La différence maximale autorisée est contrôlée par le seuil de similarité, ici $\theta = 0.1$. Dans le cas de l'attribut diamètre, cela revient à accepter les planètes ayant un diamètre entre 10 800 et 13 200 km.

L'interprétation précédente met en évidence le lien entre le choix du treillis de concepts multivalués à interroger et la précision dans les réponses obtenues pour les requêtes. En effet, un tel choix s'effectue selon la marge maximale permise entre les valeurs des attributs dans

une requête et celles caractérisant les objets retrouvés. Si aucune marge n'est permise, c'est-à-dire que les objets retrouvés doivent avoir exactement les mêmes valeurs des attributs de la requête, le treillis à interroger est celui obtenu pour un seuil de similarité nul ($\theta = 0$). Dans l'exemple considéré précédemment, il s'agit d'interroger $\underline{\mathfrak{B}}_{0.1}(G, M, W, I)$ avec la requête $Y = \{D_{0.94}, M_{0.83}, DS_{0.65}\}$. La réponse obtenue est la suivante :

1 - *Vénus* : partage $D_{0.94}$ avec la requête (pertinence : 1/3).
- *Terre* : partage et $DS_{0.65}$ avec la requête (pertinence : 1/3).

Dans cet exemple, la réponse est formée par le même ensemble d'objet que celle donnée dans le cas de $\underline{\mathfrak{B}}_{0.1}(G, M, W, I)$ mais la pertinence des objets a diminué car le nombre d'attributs partagés avec la requête est plus réduit. Si, au contraire, on élargie la marge permise entre les valeurs des attributs dans une requête et celles caractérisant les objets retrouvés, on se retrouve avec plus d'objets dans la réponse (avec une augmentation de la pertinence des objets présents dans la réponse pour des marges inférieurs). Ceci est le cas si on considère la même requête $Y = \{D_{0.94}, M_{0.83}, DS_{0.65}\}$ dans le treillis $\underline{\mathfrak{B}}_{0.4}(G, M, W, I)$. La réponse obtenue est la suivante :

1 - *Vénus* : partage $D_{[0.94,1]}$, $M_{[0.81,1]}$ et $DS_{[0.47,0.65]}$ avec la requête (pertinence : 0.93).
2 - *Terre* : partage $D_{[0.94,1]}$, $M_{[0.81,1]}$ et $DS_{[0.47,0.65]}$ avec la requête (pertinence : 0.92)
3 - *Mars* : partage $DS_{[0.65,1]}$ avec la requête (pertinence : 0.21).
4 - *Terre* : partage $DS_{[0.65,1]}$ avec la requête (pertinence : 0.20).

Le cas particulier du treillis $\underline{\mathfrak{B}}_1(G, M, W, I)$ est adapté aux requêtes constitués d'attributs sans valeurs de précision. Dans ce cas, les objets à trouver sont ceux pour lesquels les attributs considérés dans la requête ont une valeur non nulle.

Considérons par exemple la requête qui consiste à trouver "*les planètes qui ont des satellites*". Formellement, la requête est $Y = \{S\}$. La réponse à cette requête est la suivante :

1 - *Terre* : partage S avec la requête.
2 - *Mars* : partage S avec la requête.

6.4.6 Illustration dans le cas de contextes symboliques

Dans cette illustration, nous considérons le contexte symbolique donné à la table 5.6 et le treillis de concepts multivalués correspondant obtenu pour une distance de similarité égale à 2 dans la hiérarchie donnée à la figure 5.9 (figure 5.10). Un exemple de requête multivaluée est : "'*quelles sont les bases de données biologiques qui contiennent des données relatives aux facteurs de transcription chez les vertébrés*". Formellement, la requête est $Y = \{MeSH_{\,|\,\{Trans.fact.\}}$' $NCBI_{\,|\,\{Vertebrates\}}\}$. La recherche de bases de données pertinentes commence par l'insertion de la requête dans le treillis (figure 6.6). Ensuite, le concept pivot est localisé. Dans cet exemple, l'ensemble Y n'est pas fermé pour l'opérateur "$^{\downarrow}$". De ce fait, le treillis de concepts multivalués ne contient pas de concept (X, Y) et il peut y avoir plus d'un pivot. L'identification de ces pivots consiste à chercher les concepts (A, B) les plus spécifiques tels que $x \in A$. Ces concepts sont indiqués en rouge à la figure 6.6. Enfin la réponse à la requête est construite en considérant récursivement les objets dans les extensions des pivots et de leurs super-concepts dans le treillis :

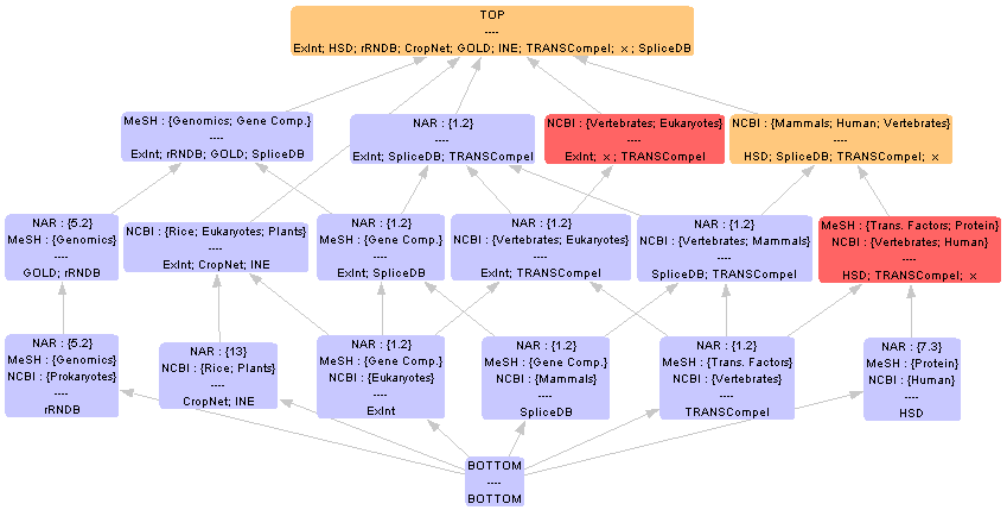


FIGURE 6.6 – Interrogation du treillis $\underline{\mathfrak{B}}_2(G, M, W, I)$ avec la requête $Y = \{MeSH_{\,|\,\{Trans.fact.\}}, NCBI_{\,|\,\{Vertebrates\}}\}$.

1 - *TRANSCompel* : partage $MeSH_{\,|\,\{Trans.\ fact.,\ Proteins\}}$ et $NCBI_{\,|\,\{Vertebrates,\ Human\}}$ avec la requête.
- *HSD* : partage $MeSH_{\,|\,\{Trans.\ fact.,\ Proteins\}}$ et $NCBI_{\,|\,\{Vertebrates,\ Human\}}$ avec la requête.

2 - *SpliceDB* : partage $NCBI_{\,|\,\{Vertebrates,\ Human,\ Mammals\}}$ avec la requête.
- *ExInt* : partage $NCBI_{\,|\,\{Vertebrates,\ Eukaryotes\}}$ avec la requête.

Le classement des bases de données dans cette réponse s'appuie uniquement sur le nombre d'attributs partagés avec la requête. De ce fait, les bases de données *TRANSCompel* et *HSD* sont classées au même rang alors qu'elles devraient être distinguées puisque la première possède les mêmes valeurs d'attributs que la requête alors que la deuxième possède des valeurs d'attributs similaires à celles des attributs de la requête. Cette "limite" dans le classement des objets pertinents est résolue par le recours à la fonction de pertinence (définition 60). L'application de la fonction de pertinence donnée dans cette définition nécessite le redefinition de la similarité, donnée de manière générique dans la définition 59, pour qu'elle soit adaptée aux données considérées. Comme dans le cas des contextes numériques, il est possible d'exploiter la similarité considérée dans le partage d'attributs. Dans cet exemple, deux valeurs w_1 et w_2 d'un attribut dans sont similaires si leur distance, $d(w_1, w_2)$, dans la hiérarchie de la figure 5.9 est inférieure à un seuil S. La similarité entre les valeurs d'un attribut se définit comme suit :

$$Sim(w_1, w_2) = \frac{1}{1 + d(w_1, w_2)}$$

La condition de partage d'attributs, "*la distance* $d(w_1, w_2)$ *est inférieure à* S", est équivalente à "*la similarité est supérieure à* $\frac{1}{1+S}$". Ainsi, les distances des objets constituant la réponse aux attributs de la requête sont données :

$$
\begin{aligned}
dist(TRANSCompel, MeSH\,|_{\{Trans.fact.\}}) &= 1 - Sim(MeSH(TRANSCompel), Trans.fact.) \\
&= 1 - Sim(Trans.fact., Trans.fact.) \\
&= 1 - frac11 + 0 \\
&= 0 \\
dist(TRANSCompel, NCBI\,|_{\{Vertebrates\}}) &= 1 - Sim(NCBI(TRANSCompel), Vertebrates) \\
&= 1 - Sim(Vertebrates, Vertebrates) \\
&= 1 - \tfrac{1}{1+0} \\
&= 0 \\
dist(HSD, MeSH\,|_{\{Trans.fact.\}}) &= 1 - Sim(MeSH(HSD), Trans.fact.) \\
&= 1 - Sim(Proteins, Trans.fact.) \\
&= 1 - \tfrac{1}{1+1} \\
&= 0.5 \\
dist(HSD, NCBI\,|_{\{Vertebrates\}}) &= 1 - Sim(NCBI(HSD), Vertebrates) \\
&= 1 - Sim(Human, Vertebrates) \\
&= 1 - \tfrac{1}{1+2} \\
&= 0.66 \\
dist(SpliceDB, NCBI\,|_{\{Vertebrates\}}) &= 1 - Sim(NCBI(SpliceDB), Vertebrates) \\
&= 1 - Sim(Mammals, Vertebrates) \\
&= 1 - \tfrac{1}{1+1} \\
&= 0.5 \\
dist(ExInt, NCBI\,|_{\{Vertebrates\}}) &= 1 - Sim(NCBI(ExInt), Vertebrates) \\
&= 1 - Sim(Eukaryotes, Vertebrates) \\
&= 1 - \tfrac{1}{1+2} \\
&= 0.66
\end{aligned}
$$

Les degrés de pertinence des objets *TRANSCompel*, *HSD*, *ExInt* et *SpliceDB* à la requête $Y = \{MeSH\,|_{\{Trans.fact.\}}, NCBI\,|_{\{Vertebrates\}}\}$ sont obtenus en appliquant la définition 60 :

$$\mathcal{P}(TRANSCompel, Q) = \frac{(1+1)}{2} = 1$$

$$\mathcal{P}(HSD, Q) = \frac{(0.5+0.66)}{2} = 0.58$$

$$\mathcal{P}(ExInt, Q) = \frac{(0.66)}{2} = 0.33$$

$$\mathcal{P}(SpliceDB, Q) = \frac{(0.5)}{2} = 0.25$$

En considérant les degrés de pertinence, il est possible d'affiner le classement :

1 - *TRANSCompel* : partage $MeSH\,|_{\{Trans.\ fact.,\ Proteins\}}$ et $NCBI\,|_{\{Vertebrates,\ Human\}}$ avec la requête. (pertinence = 1)

2 - *HSD* : partage $MeSH\,|_{\{Trans.\ fact.,\ Proteins\}}$ et $NCBI\,|_{\{Vertebrates,\ Human\}}$ avec la requête. (pertinence = 0.58)

3 - *ExInt* : partage $NCBI\,|_{\{Vertebrates,\ Eukaryotes\}}$ avec la requête. (pertinence = 0.33)

4 - *SpliceDB* : partage $NCBI\,|_{\{Vertebrates,\ Human,\ Mammals\}}$ avec la requête. (pertinence = 0.25)

L'interprétation de cette réponse est la suivante. La base de données *TRANSCompel* correspond parfaitement aux critères exprimées dans la requête. De ce fait, elle est classée au premier rang avec le degré de pertinence maximal. Les bases de données *HSD*, *SpliceDB* et *ExInt* ont des valeurs d'attributs similaires à celles exprimées dans la requête. Elles sont considérées dans la réponse car elles peuvent être vues comme des réponses plus générales ou plus spécifiques selon que leurs valeurs d'attributs sont plus générales ou plus spécifiques que celles des attributs de la requête.

De la même manière que dans le cas des contextes numériques, le choix du treillis de concepts multivalués à interroger est déterminant pour les réponses aux requêtes considérées. Ce choix reflète la flexibilité autorisée par rapport aux valeurs des attributs dans la requête. L'interrogation d'un treillis spécifique (un seuil S proche de 0) signifie que seuls les objets dont les valeurs des attributs sont très similaires à celles des attributs de la requête figurent dans la réponse. Pour illustrer cette remarque considérons la requête donnée précédemment dans le treillis de concepts multivalués obtenue pour $S = 1$ donné à la figure 6.7. L'insertion de la requête dans ce treillis

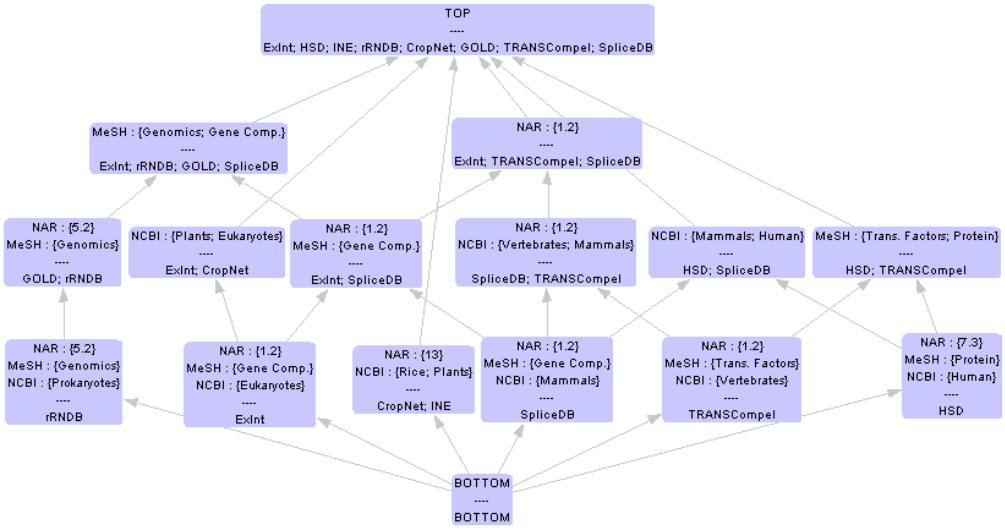


FIGURE 6.7 – Treillis de concepts multivalués, $\underline{\mathfrak{B}}_1(G, M, W, I)$, obtenu pour un seuil $S = 1$ à partir du contexte symbolique donné à la table 5.6.

produit le nouveau treillis donné à la figure 6.8. La réponse obtenue est la suivante.

1 - *TRANSCompel* : partage $MeSH_{|\{Trans.\ fact.\}}$ et $NCBI_{|\{Vertebrates\}}$ avec la requête. (pertinence = 1)
2 - *HSD* : partage $MeSH_{|\{Trans.\ fact.,\ Proteins\}}$ avec la requête. (pertinence = 0.25)
2 - *SpliceDB* : partage $NCBI_{|\{Vertebrates,\ Mammals\}}$ avec la requête. (pertinence = 0.25)

Après comparaison de cette réponse avec celle obtenue dans le cas du treillis $\underline{\mathfrak{B}}_2(G, M, W, I)$, on remarque que l'objet *ExInt* a été écarté et que les objets *HSD* et *SpliceDB* ont des degrés de pertinence moins importants. Ceci est dû au fait que les valeurs Vertebrates et *Eukarayotes* et *Human* ne sont plus similaires à la valeur *Vertebrates* et qu'en conséquence les objets *ExInt*, *HSD* et *SpliceDB* ne partagent plus l'attribut *NCBI* avec la requête.

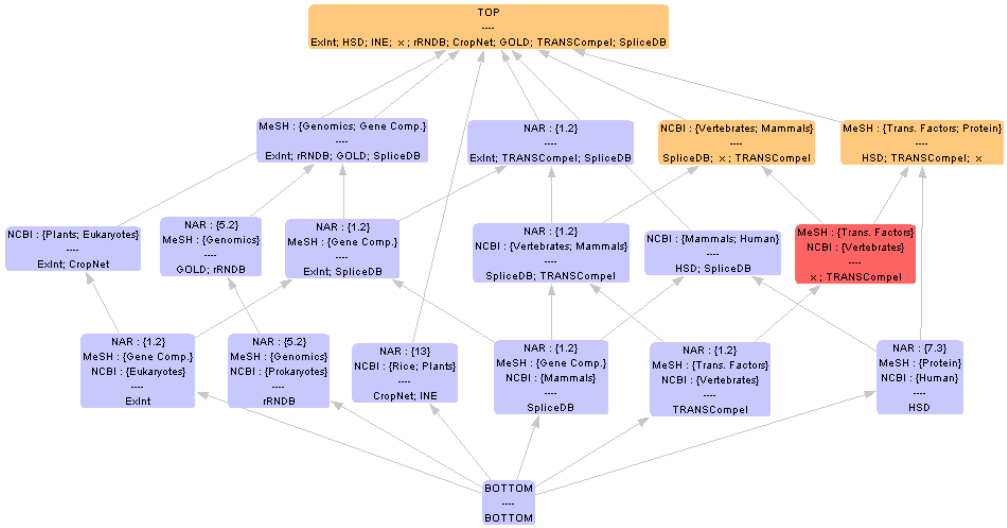


FIGURE 6.8 – Interrogation du treillis $\underline{\mathfrak{B}}_1(G, M, W, I)$ avec la requête $Y = \{MeSH_{|\{Trans.fact.\}}, NCBI_{|\{Vertebrates\}}\}$.

Réciproquement, l'interrogation d'un treillis général signifie que des objets possédant des valeurs d'attributs plus éloignées de celles des attributs de la requête peuvent aussi faire partie de la réponse. Si on considère la même requête dans le treillis de concepts multivalués le plus général, $\underline{\mathfrak{B}}_{max}(G, M, W, I)$, donné à la figure 5.13, on obtient la réponse suivante formée par tous les objets du contexte à différents degrés de pertinence selon la similarité entre les valeurs des attributs qu'ils possèdent et celles des attributs de la requête.

L'interrogation du treillis $\underline{\mathfrak{B}}_{max}(G, M, W, I)$ est particulièrement intéressante dans le cas où la requête considérée est formée d'un ensemble d'attributs sans valeurs de précisions. Des exemples de requêtes sous cette forme sont $Y = \{MeSH\}$ et $Y = \{NCBI\}$. Ces requêtes correspondent respectivement aux questions "*Quelles sont les bases de données annotées par des terms MeSH*" et "*Quelles sont les bases de données qui contiennent des données relatives aux organismes NCBI*". La réponse à la première requête est formée par les objets du contexte qui ont une valeur pour l'attribut *MeSH*. Ces objets sont *ExInt*, *HSD*, *rRNDB*, *SpliceDB*, *CropNet*, *INE* et *TRANSCompel*. De manière analogue, la réponse à la deuxième requête est formée par les objets du contexte qui ont une valeur pour l'attribut *NCBI*. Ces objets sont *ExInt*, *HSD*, *rRNDB*, *SpliceDB*, *GOLD* et *TRANSCompel*.

Chapitre 7

Mise en oeuvre

Sommaire

7.1 Introduction 129
7.2 Le prototype BR-Explorer 129
7.2.1 Architecture générale du système BR-Explorer 129
7.2.2 Le noyau du système BR-Explorer 130
7.2.3 L'interface utilisateurs du système BR-Explorer 130
7.2.4 Expérimentation 131
7.3 Le système SimBA 131
7.4 Architecture générale du système SimBA 131
7.4.1 Expérimentation 132

7.1 Introduction

La mise en oeuvre de l'approche de découverte de ressources guidée par des connaissance a produit deux prototypes : BR-Explorer [33] (**B**io**R**egistry **Explorer**) et SimBA [34] (**Sim**ilarity **B**ased concept **A**nalysis). Ces prototypes sont décrits en section 7.2 et section 7.3, respectivement.

7.2 Le prototype BR-Explorer

7.2.1 Architecture générale du système BR-Explorer

Le système BR-Explorer est le résultat de l'implémentation (en JAVA) de l'approche de découverte de ressource guidée par des connaissances de domaine détaillée dans le chapitre 4. Le système BR-Explorer se place comme interface entre les utilisateurs et l'annuaire BioRegistry pour assurer une exploitation efficace de cet annuaire. Le fonctionnement de BR-Explorer s'effectue sur un contexte formel extrait à partir des métadonnées dans BioRegistry. Les métadonnées sont d'abord exportées de la base de données sous la forme d'un fichier XML. Ensuite, un module d'analyse syntaxique, utilisant l'API JDOM [35], annexe au système BR-Explorer permet d'extraire les métadonnées souhaitées qui vont constituer le contexte formel. Ce contexte est par la suite considéré comme le corpus de recherche des sources pertinentes. BR-Explorer

33. http ://www.loria.fr/ messai/en/BR-Explorer.htm
34. http ://www.loria.fr/ messai/en/SimBA.htm
35. http ://www.jdom.org/

commence par construire le treillis de concepts correspondant à ce contexte. Ensuite des sessions de recherche peuvent être effectuées par navigation dans le treillis obtenu ou par la définition de requêtes à traiter à l'aide de l'algorithme BR-Explorer. Ce fonctionnement est assuré par les deux principaux modules du système : le noyau et l'interface utilisateurs.

7.2.2 Le noyau du système BR-Explorer

Construction du treillis de concepts

Ce module s'appuie sur une implémentation de l'algorithme AddIntent [der Merwe et al., 2004] pour la construction incrémentale de treillis de concepts. Ce choix est motivé par l'efficacité de cet algorithme comparé à d'autres algorithmes existants (voir la section 3.3.6). Le caractère incrémental de cet algorithme permet par la suite l'interrogation du treillis de concepts par insertion de requêtes.

Interrogation du treillis de concepts

Ce module s'appuie sur AddIntent pour l'insertion de la requête dans le treillis de concepts et sur une implementation de l'algorithme BR-Explorer détaillé en section 4.4 pour la recherche des sources pertinents. Ce module assure également la prise en compte des hiérarchies d'attributs pour guider la recherche des sources pertinentes conformément au fonctionnement détaillé en section 4.6.4. Pour chacune des sources pertinentes retournées en réponse à une requête, la ligne correspondante dans le contexte formel est consultée afin de récupérer les métadonnées relatives à la source et non partagées avec la requête. Ces métadonnées sont par la suite présentées avec le résultat afin de faciliter la reformulation de la requête pour une nouvelle recherche.

Exploration du treillis de concepts par navigation

Ce module assure la construction du diagramme de Hasse du treillis de concepts (avant et après insertion des requêtes). Il s'appuie sur l'API JGraph [36] pour la construction d'éléments graphiques correspondant aux concepts et aux liens hiérarchiques dans le treillis. Ce module assure également la construction du diagramme correspondant uniquement aux concepts formant le voisinage de la requête dans le treillis. Ce voisinage est centré autour de la requête et formé par les concepts les plus proche de la requête.

Parcours d'ontologies pour enrichir la requête

Ce module assure le parcours des concepts d'une hiérarchie de classes codé en OWL pour l'identification des métadonnées ayant des relations sémantiques avec celles de la requête dans une ontologie donnée. Il s'appuie sur l'API Jena [37] pour implémenter certaines méthodes permettant d'identifier, dans l'ontologie, les concepts correspondants aux métadonnées de la requête et de récupérer leurs concepts les plus proches sémantiquement. Les métadonnées correspondantes à ces concepts dans le contexte sont par la suite utilisées pour enrichir la requête.

7.2.3 L'interface utilisateurs du système BR-Explorer

Ce module a pour rôle de faciliter l'utilisation des différentes fonctionnalités assurées par le noyau. Il s'appuie sur une interface graphique développée en JAVA (voir Annexe A) et permet

36. http ://www.jgraph.com/
37. http ://jena.sourceforge.net/

d'effectuer les opérations suivantes :

- La définition de requêtes : Les requêtes sont exprimées sous la forme d'un ensemble de métadonnées (avec possibilité de considérer la négation en utilisant l'opérateur logique NOT dans le cas où les sources souhaitées ne doivent pas être annotées par une ou plusieurs métadonnées). Le choix des métadonnées est facilité d'une part par la présentation de l'ensemble de toutes les métadonnées du contexte sous la forme d'une liste déroulante et d'autre part par l'aide à la saisie (en donnant des propositions pour compléter la saisie d'une métadonnée). L'interface de définition de requêtes dans BR-Explorer est schématisée à la figure A.2.
- La définition de dépendances entre attributs : Les attributs principaux sont d'abord choisis ensuite pour chaque attribut il est possible de choisir un ou plusieurs attributs secondaires. Le tout est ensuite visualisé sous la forme d'une hiérarchie. L'interface de définition de hiérarchies d'attributs dans BR-Explorer est schématisée à la figure A.5.
- L' enrichissement de requête : Les métadonnées proposées selon le mode d'enrichissement choisi (généralisation, spécialisation ou mixte) sont présentées afin de choisir celles qui seront retenues pour la nouvelle requête avant de la soumettre au noyau.
- Affichage des réponses et reformulation de requêtes : Les sources de données pertinentes retournées par le noyau et ordonnées selon leur pertinence sont affichées. Pour chaque source il est possible de visualiser l'ensemble de métadonnées qu'elle partage avec la requête et l'ensemble des métadonnées qui la décrive et qui ne font pas partie de la requête. La visualisation de ces métadonnées est faite dans le but de faciliter la reformulation de requêtes par suppression de certaines métadonnées et/ou ajout d'autres. A partir de la réponse affichée il est aussi possible d'accéder directement au site d'une source et de procéder à son exploitation. Ceci est assuré par la prise en compte de l'url de chaque source lors de l'exportation du contexte formel à partir de la base de données BioRegistry. La figure A.3 schématise le résultat de l'exécution de BR-Explorer sur un exemple de requête.
- Visualisation des treillis et zoom sur le voisinage de la requête : Le diagramme de Hasse du treillis de concepts est affiché pour faciliter la navigation dans le treillis. Dans le cas de treillis de grande taille, cette opération devient difficile. Dans ce cas, il est possible d'afficher uniquement les concepts les plus proches de la requête. Le choix de ces concepts dépend de leurs distances à la requête (en terme de nombre de liens dans le treillis). La figure A.4 montre le voisinage d'une requête.

7.2.4 Expérimentation

Le système BR-Explorer tourne sur un corpus de 729 sources décrites par 231 métadonnées relatives aux sujets et aux organismes concernés par le contenu des sources. Le treillis de concepts obtenu est formé de 638 concepts formels. L'exécution de BR-Explorer sur un exemple de requête est donné dans l'annexe A.

7.3 Le système SimBA

7.4 Architecture générale du système SimBA

Le système SimBA est le résultat de l'implémentation (en JAVA) des connexion de Galois multivaluées introduites dans le chapitre 5. Dans son état actuel, SimBA contient les modules suivants.

- Un module de construction de treillis de concepts multivalués pour différents types de contextes. Cette étape s'appuie sur un algorithme qui étend la méthode de Malgrange pour le calcul des concepts par intersections successives entre leurs intensions et extensions.
- Un module de visualisation de treillis de concepts multivalués. Ce module s'appuie sur l'API prefuse [38] et facilite la navigation dans le treillis en coloriant de manière dynamique les concepts visités et leurs voisinages, en réduisant le treillis au voisinage du concept courant et en affichant les concepts sous différentes formes (intension uniquement, extension uniquement, support des intensions, etc.).
- Un module d'interrogation des treillis de concepts multivalués qui implémente la méthode détaillée en section 6.4.

7.4.1 Expérimentation

Différentes connexions de Galois ont été implémentées au sein de SimBA. La première connexion de Galois est définie pour les contextes multivalués numériques et détaillée dans le chapitre 5 et le chapitre 6 sur le contexte multivalué des planètes du système solaire. La deuxième connexion de Galois est définie pour les contextes multivalués symboliques et détaillée dans le chapitre 5 et le chapitre 6 sur le contexte multivalué symbolique donné dans la table 5.6.

L'expérimentation de SimBA sur les données de BioRegistry est assuré sur un contexte plurivalué symbolique formé par 48 sources de données et leurs termes MeSH appartenant à 5 branches du thésaurus MeSH ("Organisms [B]", "Chemicals and Drugs [D]", "Analytical, Diagnostic and Therapeutic Techniques and Equipment [E]", "Biological Sciences [G]" et "Natural Sciences [H]"). Il y a 39 différents termes MeSH décrivant les 48 sources et répartis dans le contexte avec une densité de 34%.

La similarité entre les termes MeSH dans le contexte est calculée en s'appuyant sur leurs identifiants MeSH (tree number). L'identifiant d'un terme MeSH détermine de manière unique sa position du terme dans la hiérarchie et se présente comme une suite de numéros correspondants aux prédécesseurs du termes à partir de la racine jusqu'au terme lui même. Par exemple, le terme "Mammals" a comme identifiant MeSH "B01.150.900.649" qui signifie que "Mammals" appartient à la branche B du MeSH et ses prédécesseurs sont les termes identifiés par B01.150.900 ("Vertebrates"), B01.150 ("Chordata") et B01 ("Animals"). Ainsi il est possible d'identifier l'ensemble des prédécesseurs d'un terme dans MeSH sans avoir à chercher dans le thésaurus. Nous définissons la similarité entre deux termes t1 et t2 comme la proportion du nombre de prédécesseurs communs à t1 et t2 parmi tous les termes identifié à partir de t1 et t2. Par exemple, la similarité entre Mammals (B01.150.900.649) et Reptiles (B01.150.900.833) est égale à $7/8 = 0.875$. La similarité entre deux ensembles de termes est définie comme étant la moyenne des similarités des termes des deux ensembles. Etant donné deux ensembles de termes $L_1 = \{t_{11}, t_{12}...t_{1n}\}$ et $L_2 = \{t_{21}, t_{22}, ...t_{2k}\}$, la similarité entre L_1 et L_2 est donnée par la formule suivante :

$$sim(L_1, L_2) = \frac{1}{n * k} \sum_{i=1}^{n} \sum_{j=1}^{k} sim(t_{1i}, t_{2j}).$$

En s'appuyant sur cette définition de la similarité, plusieurs treillis de concepts multivalués sont construits pour différents seuil de similarité. Le treillis de concepts multivalués est formé de 43 concepts pour un seuil de similarité $S = 1$, 41 concepts pour $S = 0.7$, 37 concepts pour $S = 0.5$, 35 concepts pour $S = 0.4$ et 25 concepts pour $S = 0.1$. Les treillis de concepts multivalués obtenu pour $S = 0.5$ et $S = 0.1$ sont donnés respectivement à la figure 7.1 et la figure 7.2.

38. http ://prefuse.org/

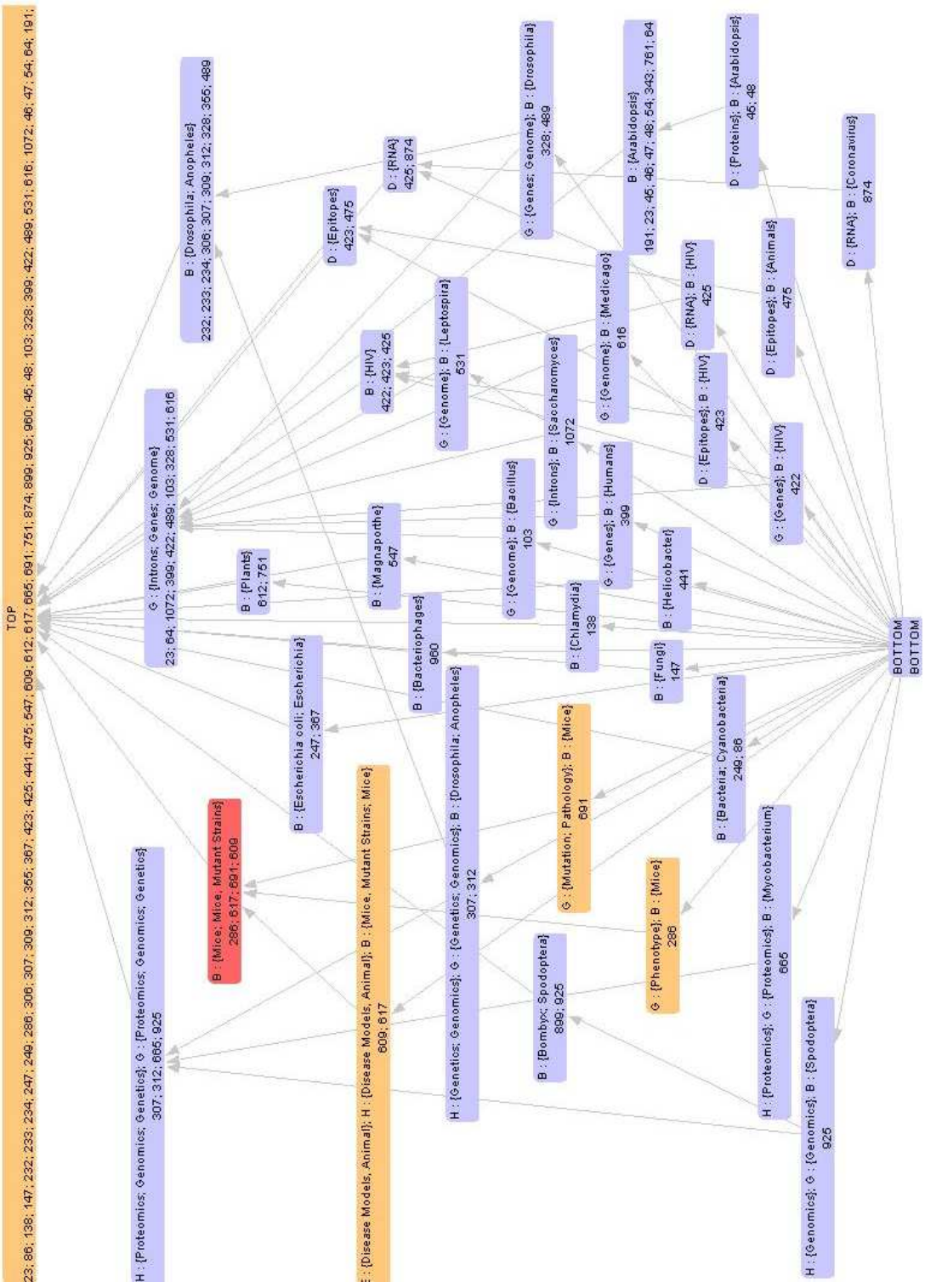


FIGURE 7.1 – Un treillis de concepts multivalués correspondant au contexte multivalué formé par 48 sources de données dans BioRegistry pour un seuil $S = 0.5$.

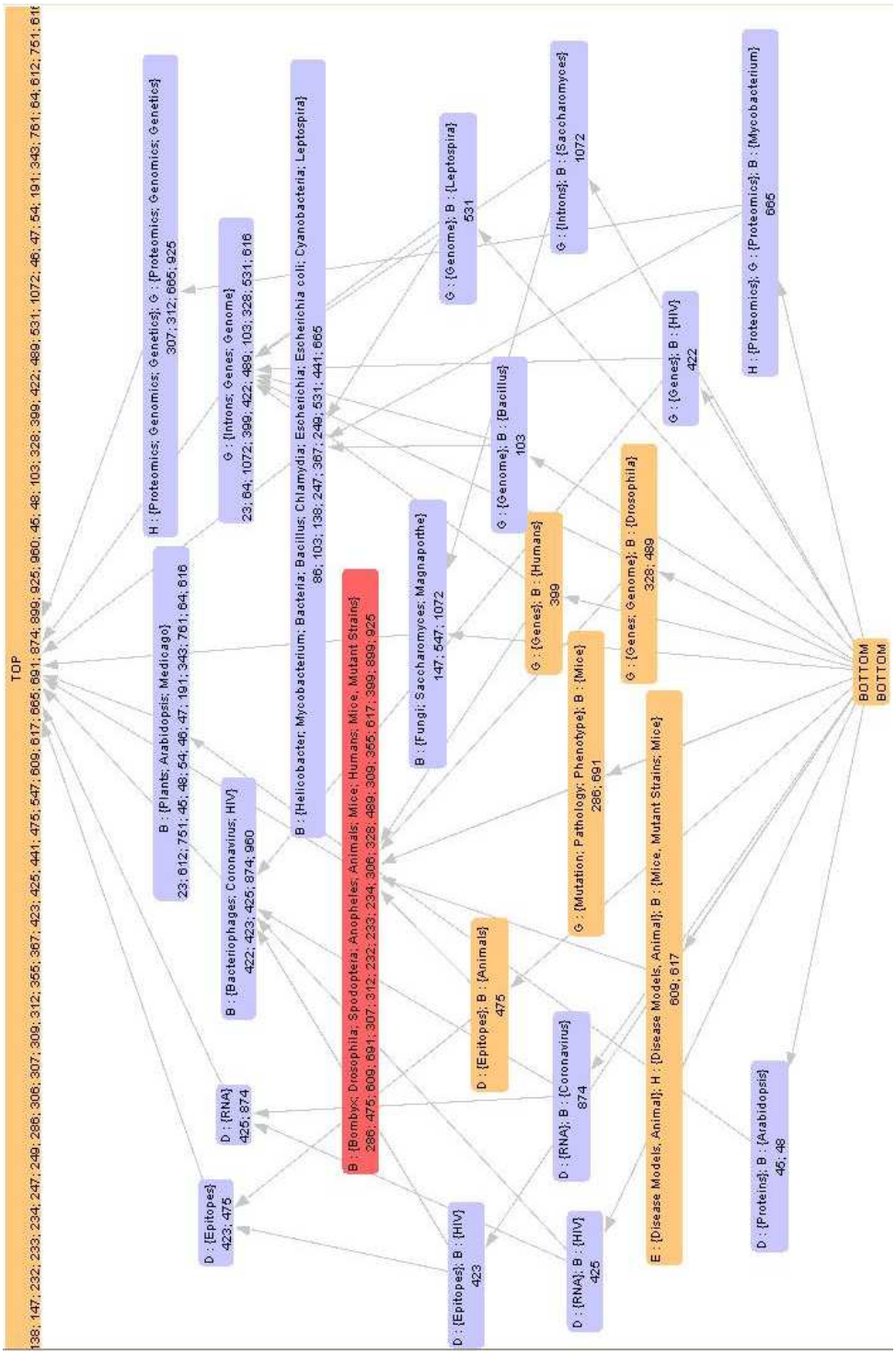


FIGURE 7.2 – Un treillis de concepts multivalués correspondant au contexte multivalué formé par 48 sources de données dans BioRegistry pour un seuil $S = 0.1$.

Chapitre 8

Conclusion et perspectives

8.1 Conclusion générale

Les travaux effectués dans le cadre de cette thèse ont porté sur l'exploitation des connaissances de domaines dans un processus de découvertes de ressources pertinentes pour répondre à des requêtes complexes. Plus précisément, il s'agit de tenir compte du contenu des ressources génomiques de façon sémantique pour aboutir à un appariement efficace entre les requêtes posées et les ressources qui répondent au mieux à ces requêtes. Ces travaux ont abouti à la mise en place d'une approche originale qui étend l'Analyse de Concepts Formels et la combine avec des ontologies de domaines pour la découverte de ressources hétérogènes. Cette approche est caractérisées par deux apports majeurs par rapport à l'état de l'art.

Le premier apport consiste à utiliser des connaissances de domaines dans toutes les étapes du processus de découverte de ressources : annotation et organisation des ressources, navigation dans cette organisation, définition et traitement des requêtes complexes. Dans l'étape d'annotation, des termes provenant d'ontologies de domaines sont utilisés dans un modèle conforme aux standards d'annotation pour décrire de manière optimale les caractéristiques et le contenu des ressources. En s'appuyant sur cette annotation, les ressources sont organisées dans un treillis de concepts. Ce treillis constitue par la suite un support de recherche de ressources pertinentes ou deux mode de recherche peuvent être combinés : la recherche par navigation et la recherche par interrogation. La recherche par navigation consiste à partir de l'ensemble de toutes les sources et de le restreindre en choisissant progressivement les caractéristiques des ressources recherchées. La recherche par interrogation consiste à exprimer ces caractéristiques dans une requête qui sera insérée dans le treillis. Une mesure de similarité combinée avec un parcours du voisinage de la requête dans le treillis résultant permet d'identifier les ressources pertinentes. Pour permettre la prise en compte de requêtes complexes, nous avons proposé deux solutions complémentaires. La première consiste à effectuer l'enrichissement sémantiques de requête et la deuxième consiste à prendre en compte les préférences dans les requêtes.

Le deuxième apport consiste à étendre l'analyse de concepts formels pour le traitement de données complexes et hétérogènes afin de prendre en compte l'hétérogénéité des ressources génomiques et la complexité des objets biologiques auxquels elles font référence d'une part et de définir des requête précises et complexes d'autre part. L'extension consiste à proposer une nouvelle connexion de Galois pour le traitement de contextes formels multivalués. La connexion de Galois multivaluée s'appuie sur les relations sémantiques entres les attributs pour former les concepts formels multivalués et les organiser en treillis. Les treillis de concepts multivalués peuvent avoir des niveaux de précision variés selon les relations sémantiques considérées ce qui

leur permet d'être un bon outil pour la classification à granularité variable où on aimerait avoir une première représentation assez générale et de la raffiner par la suite en fonction des besoins en précision. La connexion de Galois multivaluée est à la base d'une méthode de recherche d'information par treillis de concepts multivalués qui combine la navigation dans le treillis avec une mesure de similarité bien définie pour traiter efficacement des requêtes complexes. En effet, il est possible de définir des requêtes pondérés où le poids attribué à un mot clé dans une requête donnée peut être de deux formes : (i) une valeur précise exigée dans les réponses à trouver ou (ii) un ensemble de valeurs représentant la marge des valeurs acceptables dans les réponses recherchées.

8.2 Perspectives

Les perspectives de ce travail sont à la fois nombreuses et prometteuses. Tout d'abord, l'ACF guidée par des connaissances de domaine peut apporter des solutions à différents problèmes rencontrés par les applications qui s'appuient sur l'ACF classique. En particulier, la flexibilité des treillis multivalués permet d'éviter le problème de passage à l'échelle et la manipulation des treillis de concepts de grande taille. Il conviendrait donc d'examiner l'application de l'ACF guidée par des connaissances de domaine à ces applications et de voir son apport effectif. On pourrait s'intéresser ici à la recherche d'information sur le Web, à l'étude des données d'expressions de gènes afin d'extraire des groupes de gènes co-exprimés qui peuvent potentiellement caractériser des processus biologiques, etc. De plus, la prise de l'aspect sémantique permet à l'ACF guidée par des connaissances de domaine d'être en parfaite conformité avec les récents développement du Web sémantique et aussi de contribuer à ce développement. Cette contribution peut être assurée par l'application de l'ACF guidée par des connaissances de domaine à la découverte de nouvelles connaissances et à la construction d'ontologies. Dans ce cadre, les treillis de concepts multivalués peuvent être exploités pour la mise en place d'une approche de fouille de données complexes et hétérogènes.

Afin d'aborder avec succès ces différentes pistes et d'autres aussi, il est important de disposer d'un appuie algorithmique performant et des implémentations efficaces des fonctions de base de l'ACF guidée par des connaissances de domaine (construction des treillis, manipulation des ressources sémantiques, visualisation, etc.).

Bibliographie

[DBL, 2005] (2005). Data Integration in the Life Sciences, Second InternationalWorkshop, DILS 2005, San Diego, CA, USA, July 20-22, 2005, Proceedings.

[Azmeh et al., 2008] Azmeh, Z., Huchard, M., Tibermacine, C., Urtado, C., and Vauttier, S. (2008). Wspab : A tool for automatic classification and selection of web services using formal concept analysis. ECOWS'08 IEEE Sixth European Conference on Web Services, pages 31–40.

[Baeza-Yates and Ribeiro-Neto, 1999] Baeza-Yates, R. A. and Ribeiro-Neto, B. A. (1999). Modern Information Retrieval. ACM Press / Addison-Wesley.

[Barbut and Monjardet, 1970] Barbut, M. and Monjardet, B. (1970). Ordre et classification : Algèbre et combinatoire, Tome II. Hachette, Paris.

[Baziz, 2005] Baziz, M. (2005). Indexation Conceptuelle Guidée par Ontologie pour la Recherche d'Information. PhD thesis, Université Paul Sabatier (Toulouse III).

[Baziz et al., 2006] Baziz, M., Boughanem, M., Prade, H., and Pasi, G. (2006). A fuzzy logic approach to information retrieval using an ontology-based representation of documents. In Sanchez, E., editor, Fuzzy Logic and the semantic web, chapter 18, pages 363–377. Elsevier.

[Belkin, 2008] Belkin, N. J. (2008). Some (what) grand challenges for information retrieval. In [Macdonald et al., 2008], page 1.

[Belohlávek, 1999] Belohlávek, R. (1999). Lattices generated by binary fuzzy relations. Tatra Mountains Mathematical Publications, 16 :11 – 19.

[Belohlavek, 2002] Belohlavek, R. (2002). Fuzzy Relational Systems : Foundations and Principles. Kluwer Academic Publishers, Norwell, MA, USA.

[Belohlávek and Sklenar, 2005] Belohlávek, R. and Sklenar, V. (2005). Formal concept analysis constrained by attribute-dependency formulas. In [Ganter and Godin, 2005], pages 176–191.

[Belohlávek et al., 2004] Belohlávek, R., Sklenar, V., and Zacpal, J. (2004). Formal concept analysis with hierarchically ordered attributes. International Journal of General Systems, 33(4) :283 – 294.

[Belohlávek and Vychodil, 2005] Belohlávek, R. and Vychodil, V. (2005). What is a fuzzy concept lattice ? In 3rd Internationnal Conference on Concept Lattices and Their Applications CLA 2005, September 7-9, pages 34 – 45, Olomouc, Czech Republic.

[Bendaoud et al., 2008] Bendaoud, R., Napoli, A., and Toussaint, Y. (2008). Formal concept analysis : A unified framework for building and refining ontologies. In [Gangemi and Euzenat, 2008], pages 156–171.

[Berendt et al., 2002] Berendt, B., Hotho, A., and Stumme, G. (2002). Towards semantic web mining. In Proceedings of the First International Semantic Web Conference on The Semantic Web, pages 264–278. Springer-Verlag.

[Berners-Lee et al., 2001] Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic web. Scientific American, 284(5) :35–43.

[Berthold et al., 2005] Berthold, M. R., Glen, R. C., Diederichs, K., Kohlbacher, O., and Fischer, I., editors (2005). Computational Life Sciences, First International Symposium, CompLife 2005, Konstanz, Germany, September 25-27, 2005, Proceedings, volume 3695 of LNCS. Springer.

[Birkhoff, 1967] Birkhoff, G. (1967). Lattice Theory, volume 25 of ASM Colloquium Publications. AMS, Providence, RI, 3rd edition. 1st ed., 1940 ; 2nd ed., 1948.

[Birkland and Yona, 2006] Birkland, A. and Yona, G. (2006). BIOZON : a system for unification, management and analysis of heterogeneous biological data. BMC Bioinformatics, 7(70) :1–24.

[Bock, 2000] Bock (2000). Analysis of Symbolic Data : Exploratory Methods for Extracting Statistical Information from Complex Data. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[Bock and Diday, 2000] Bock, H. H. and Diday, E., editors (2000). Analysis of Symbolic Data : Exploratory Methods for Extracting Statistical Information from Complex Data, volume 15 of Studies in Classification, Data Analysis, and Knowledge Organization, Secaucus, NJ, USA. Springer-Verlag New York, Inc.

[Bordat, 1986] Bordat, J.-P. (1986). Calcul pratique du treillis de galois d'une correspondance. Mathématiques et Sciences Humaines, 96 :31–47.

[Bouzeghoub et al., 2000] Bouzeghoub, M., Klusch, M., Nutt, W., and Sattler, U., editors (2000). Proceedings of the 7th International Workshop on Knowledge Representation meets Databases (KRDB 2000), Berlin, Germany, August 21, 2000, volume 29 of CEUR Workshop Proceedings. CEUR-WS.org.

[Brito and Polaillon, 2005] Brito, P. and Polaillon, G. (2005). Structuring probablilistic data by Galois lattices. Mathématiques et Sciences humaines / Mathematics and Social Sciences, (169) :77–104.

[Buttler et al., 2002] Buttler, D., Coleman, M., Critchlow, T., Fileto, R., Han, W., Pu, C., Rocco, D., and Xiong, L. (2002). Querying Multiple Bioinformatics Information Sources : Can Semantic Web Research Help ? SIGMOD Record, 31(4) :59–64.

[Carmel et al., 2002] Carmel, D., Farchi, E., Petruschka, Y., and Soffer, A. (2002). Automatic query refinement using lexical affinities with maximal information gain. In SIGIR '02 : Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, pages 283–290. ACM Press.

[Carpineto et al., 2006] Carpineto, C., Pietra, A. D., Mizzaro, S., and Romano, G. (2006). Mobile clustering engine. In [Lalmas et al., 2006], pages 155–166.

[Carpineto and Romano, 1993] Carpineto, C. and Romano, G. (1993). Galois : An order-theoretic approach to conceptual clustering. Proceedings of 10th International Conference on Machine Learning, Amherst, pages 33–40.

[Carpineto and Romano, 1996] Carpineto, C. and Romano, G. (1996). A lattice conceptual clustering system and its application to browsing retrieval. Machine Learning, 24(2) :95–122.

[Carpineto and Romano, 2000] Carpineto, C. and Romano, G. (2000). Order-theoretical ranking. Journal of the American Society for Information Science, 51(7) :587–601.

[Carpineto and Romano, 2004a] Carpineto, C. and Romano, G. (2004a). Concept Data Analysis : Theory and Applications. John Wiley & Sons.

[Carpineto and Romano, 2004b] Carpineto, C. and Romano, G. (2004b). Exploiting the potential of concept lattices for information retrieval with credo. Journal of Universal Computer Science, 10(8) :985–1013.

[Carpineto and Romano, 2005] Carpineto, C. and Romano, G. (2005). Using concept lattices for text retrieval and mining. In [Ganter et al., 2005a], pages 161–179.

[Cellier, 2008] Cellier, P. (2008). DeLLIS : Débogage de programmes par localisation de fautes avec un système d'information logique. Thèse, Université de Rennes 1.

[Challam and Gauch, 2004] Challam, V. and Gauch, S. (2004). Contextual information retrieval using ontology-based user profiles. Nineteenth National Conference on Artificial Intelligence (AAAI-04), San Jose, CA.

[Charlet et al., 2003] Charlet, J., Bachimont, B., and Troncy, R. (2003). Ontologies pour le web sémantique. In Charlet, J., Laublet, P., and Reynaud, C., editors, Action spécifique 32 CNRS/STIC Web sémantique Rapport final, volume 2, pages 43–63.

[Chaudron and Maille, 1998] Chaudron, L. and Maille, N. (1998). 1st Order Logic Formal Concept Analysis : from logic programming to theory. Computer and Information Science, 3(13).

[Chein, 1969] Chein, M. (1969). Algorithme de recherche des sous-matrices premières d'une matrice. Bull. Math. Soc. Sci. Math. R.S. Roumanie, 13 :21–25.

[Cormen et al., 1992] Cormen, T., Leiserson, C., and Rivest, R. (1992). Introduction à l'algorithmique. Dunod.

[Cormen et al., 1990] Cormen, T. H., Leiserson, C. E., and Rivest, R. L. (1990). Introduction to algorithms. MIT Electrical Engineering and Computer Science. MIT Press, Cambridge, MA, USA.

[Dau and Klinger, 2005] Dau, F. and Klinger, J. (2005). From formal concept analysis to contextual logic. In [Ganter et al., 2005b], pages 81–100.

[Dau et al., 2005] Dau, F., Mugnier, M.-L., and Stumme, G., editors (2005). Conceptual Structures : Common Semantics for Sharing Knowledge : 13th International Conference on Conceptual Structures, ICCS 2005, Kassel, Germany, July 18-22, 2005. Proceedings, volume 3596 of LNCS. Springer-Verlag.

[Davey and Priestley, 2002] Davey, B. A. and Priestley, H. A. (2002). Introduction to Lattices and Order. Cambridge University Press, 2nd edition edition.

[Davidson et al., 2001] Davidson, S. B., Crabtree, J., Brunk, B. P., Schug, J., Tannen, V., Overton, G. C., and Stoeckert, C. J. (2001). K2/Kleisli and GUS : experiments in integrated access to genomic data sources. IBM systems journal, 40(2) :512–531.

[Dehaspe, 1999] Dehaspe, L. (1999). Frequent pattern discovery in first-order logic. AI Commun., 12(1-2) :115–117.

[Dehaspe and Toivonen, 1999] Dehaspe, L. and Toivonen, H. (1999). Discovery of frequent DATALOG patterns. Data Mining and Knowledge Discovery, 3(1) :7–36.

[Dekkers and Weibel, 2003] Dekkers, M. and Weibel, S. (2003). State of the Dublin Core Metadata Initiative. D-Lib Magazine, 9(4).

[Delugach and Stumme, 2001] Delugach, H. S. and Stumme, G., editors (2001). Conceptual Structures : Broadening the Base, 9th International Conference on Conceptual Structures, ICCS 2001, Stanford, CA, USA, July 30-August 3, 2001, Proceedings, volume 2120 of Lecture Notes in Computer Science. Springer.

[der Merwe et al., 2004] der Merwe, D. V., Obiedkov, S. A., and Kourie, D. G. (2004). AddIntent : A New Incremental Algorithm for Constructing Concept Lattices. In Eklund, P. W., editor, ICFCA Concept Lattices, Second International Conference on Formal Concept Analysis, ICFCA 2004, Sydney, Australia, February 23-26, 2004, Proceedings, volume 2961, pages 372–385. Springer.

[Devignes et al., 2004] Devignes, M.-D., Messai, N., Napoli, A., Osman, S., and Smail-Tabbone, M. (2004). Intelligent access to genomic sources on the web. In W3C Workshop on Semantic Web and Life Sciences (position paper), Cambridge, MA, USA.

[Devignes et al., 2002] Devignes, M.-D., Smail, M., and Boudjlida, N. (2002). Collecte de donnés biologiques à partir de sources multiples et hétérogènes. vers une structure de médiation conviviale et orientée source. In Journées scientifiques sur le Web sémantique, Paris.

[Discala et al., 2000] Discala, C., Benigni, X., Barillot, E., and Vaysseix, G. (2000). DBCAT : a catalog of 500 biological databases. Nucleic Acids Research, 28(1) :8–9.

[Dowling, 1993] Dowling, C. E. (1993). On the irredundant generation of knowledge spaces. Journal of Mathematical Psychology, 37(1) :49–62.

[Ducrou and Eklund, 2007] Ducrou, J. and Eklund, P. W. (2007). Searchsleuth : The conceptual neighbourhood of an web query. In [Eklund et al., 2008].

[Ducrou and Eklund, 2008] Ducrou, J. and Eklund, P. W. (2008). An intelligent user interface for browsing and searching mpeg-7 images using concept lattices. Int. J. Found. Comput. Sci., 19(2) :359–381.

[Eklund et al., 2008] Eklund, P. W., Diatta, J., and Liquiere, M., editors (2008). Proceedings of the Fifth International Conference on Concept Lattices and Their Applications, CLA 2007, Montpellier, France, October 24-26, 2007, volume 331 of CEUR Workshop Proceedings. CEUR-WS.org.

[Eklund and Haemmerlé, 2008] Eklund, P. W. and Haemmerlé, O., editors (2008). Conceptual Structures : Knowledge Visualization and Reasoning, 16th International Conference on Conceptual Structures, ICCS 2008, Toulouse, France, July 7-11, 2008, Proceedings, volume 5113 of Lecture Notes in Computer Science. Springer.

[Etzold and Argos, 1993] Etzold, T. and Argos, P. (1993). SRS–an indexing and retrieval tool for flat file data libraries. Computer Applications In The Biosciences : CABIOS, 9(1) :49–57.

[Ferré, 2002] Ferré, S. (2002). Systèmes d'information logiques : un paradigme logico-contextuel pour interroger, naviguer et apprendre. Thèse d'université, Université de Rennes 1.

[Ferré, 2007] Ferré, S. (2007). Camelis : Organizing and browsing a personal photo collection with a logical information system. In [Eklund et al., 2008].

[Ferré and Ridoux, 2000] Ferré, S. and Ridoux, O. (2000). A logical generalization of formal concept analysis. In [Ganter and Mineau, 2000], pages 371–384.

[Ferré and Ridoux, 2001] Ferré, S. and Ridoux, O. (2001). Searching for objects and properties with logical concept analysis. In [Delugach and Stumme, 2001], pages 187–201.

[Ferré and Ridoux, 2004] Ferré, S. and Ridoux, O. (2004). Introduction to logical information systems. Inf. Process. Manage., 40(3) :383–419.

[Franconi and Kifer, 1999] Franconi, E. and Kifer, M., editors (1999). Proceedings of the 6th International Workshop on Knowledge Representation meets Databases (KRDB'99), Linköping, Sweden, July 29-30, 1999, volume 21 of CEUR Workshop Proceedings. CEUR-WS.org.

[Galperin, 2004] Galperin, M. Y. (2004). The Molecular Biology Database Collection : 2004 update. Nucleic Acids Research, 32 :D4–D22.

[Galperin and Cochrane, 2009] Galperin, M. Y. and Cochrane, G. R. (2009). Nucleic acids research annual database issue and the nar online molecular biology database collection in 2009. Nucleic Acids Research, 37(Database-Issue) :D1–D4.

[Ganesan et al., 2003] Ganesan, P., Garcia-Molina, H., and Widom, J. (2003). Exploiting hierarchical domain structure to compute similarity. ACM Transactions on Information Systems (TOIS), 21(1) :64–93.

[Gangemi and Euzenat, 2008] Gangemi, A. and Euzenat, J., editors (2008). Knowledge Engineering : Practice and Patterns, 16th International Conference, EKAW 2008, Acitrezza, Italy, September 29 - October 2, 2008. Proceedings, volume 5268 of Lecture Notes in Computer Science. Springer.

[Ganter, 1984] Ganter, B. (1984). Two basic algorithms in concept analysis. FB4-Preprint 831, Technische Hochschule Darmstadt.

[Ganter and Godin, 2005] Ganter, B. and Godin, R., editors (2005). Formal Concept Analysis, Third International Conference, ICFCA 2005, Lens, France, February 14-18, 2005, Proceedings, volume 3403 of Lecture Notes in Computer Science. Springer.

[Ganter and Kuznetsov, 2001] Ganter, B. and Kuznetsov, S. O. (2001). Pattern structures and their projections. In [Delugach and Stumme, 2001], pages 129–142.

[Ganter and Mineau, 2000] Ganter, B. and Mineau, G. W., editors (2000). Conceptual Structures : Logical, Linguistic, and Computational Issues, 8th International Conference on Conceptual Structures, ICCS 2000, Darmstadt, Germany, August 14-18, 2000, Proceedings, volume 1867 of Lecture Notes in Computer Science. Springer.

[Ganter and Reuter, 1991] Ganter, B. and Reuter, K. (1991). Finding all closed sets : A general approach. Order, 8(3) :283–290.

[Ganter et al., 2005a] Ganter, B., Stumme, G., and Wille, R., editors (2005a). Formal Concept Analysis, Foundations and Applications, volume 3626 of Lecture Notes in Computer Science. Springer.

[Ganter et al., 2005b] Ganter, B., Stumme, G., and Wille, R., editors (2005b). Formal Concept Analysis, Foundations and Applications, volume 3626 of Lecture Notes in Computer Science. Springer.

[Ganter and Wille, 1999] Ganter, B. and Wille, R. (1999). Formal Concept Analysis. Springer, mathematical foundations edition.

[Goble et al., 2001] Goble, C. A., Stevens, R., Ng, G., Bechhofer, S., Paton, N. W., Baker, P. G., Peim, M., and Brass, A. (2001). Transparent Access to Multiple Bioinformatics Information Sources. IBM Systems Journal, 40(2) :532–551.

[Godin et al., 1995a] Godin, R., Mineau, G., and Missaoui, R. (1995a). Incremental structuring of knowledge bases. In Ellis, G., Levinson, R. A., Fall, A., and Dahl, V., editors, Proceedings of the 1st International Symposium on Knowledge Retrieval, Use, and Storage for Efficiency (KRUSE'95), Santa Cruz (CA), USA, pages 179–193. Department of Computer Science, University of California at Santa Cruz.

[Godin et al., 1995b] Godin, R., Mineau, G. W., and Missaoui, R. (1995b). Méthodes de classification conceptuelle basées sur les treillis de Galois et applications. Revue d'intelligence artificielle, 9(2) :105–137.

[Godin et al., 1995c] Godin, R., Missaoui, R., and Alaoui, H. (1995c). Incremental Concept Formation Algorithms Based on Galois (Concept) Lattices. Computational Intelligence, 11 :246–267.

[Godin et al., 1993] Godin, R., Missaoui, R., and April, A. (1993). Experimental comparison of navigation in a galois lattice with conventional information retrieval methods. International Journal of Man-machine Studies, 38 :747–767.

[Godin and Valtchev, 2005] Godin, R. and Valtchev, P. (2005). Formal concept analysis-based class hierarchy design in object-oriented software development. In [Ganter et al., 2005a], pages 304–323.

[Gruber, 1993] Gruber, T. (1993). A translation approach to portable ontology specification. Knowledge Acquisition, 5(2) :199–220.

[Gruninger and Lee, 2002] Gruninger, M. and Lee, J. (2002). Ontology : applications and design. Communications of the ACM, 45(2) :39–41.

[Guarino et al., 1999] Guarino, N., Masolo, C., and Vetere, G. (1999). Ontoseek : Content-based access to the web. IEEE Intelligent Systems, 14(3) :70–80.

[Guenoche, 1990] Guenoche, A. (1990). Construction du treillis de galois d'une relation binaire. Mathmatiques, Informatique et Sciences Humaines, 28ème année(109) :41–53.

[Guenoche and Mechelen, 1993] Guenoche, A. and Mechelen, I. V. (1993). Galois approach to the induction of concepts. In Mechelen, I. V., Hampton, J., Michalski, R., and Theuns, P., editors, Categories and Concepts. Theoretical Views and Inductive Data Analysis, pages 287–308. Academic Press, London.

[Halevy, 2001] Halevy, A. Y. (2001). Answering queries using views : A survey. The VLDB Journal, 10(4) :270–294.

[Hernandez and Kambhampati, 2004] Hernandez, T. and Kambhampati, S. (2004). Integration of biological sources : current systems and challenges ahead. ACM SIGMOD Record, 33(3) :51–60.

[Hliaoutakis et al., 2006] Hliaoutakis, A., Varelas, G., Voutsakis, E., Petrakis, E. G. M., and Milios, E. E. (2006). Information retrieval by semantic similarity. International Journal on Semantic Web and Information Systems, 2(3) :55–73.

[II et al., 2003] II, R. J. C., Eklund, P. W., and Stumme, G. (2003). Document retrieval for e-mail search and discovery using formal concept analysis. Applied Artificial Intelligence, 17(3) :257–280.

[Jaoua, 2007] Jaoua, A. (2007). Conceptual structured browsing : Applications for information retrieval, document summarization and meta-search engine design. In Kuznetsov, S. O. and Schmidt, S., editors, 5th International Conference, ICFCA 2007, Clermont-Ferrand, France, February 12-16, 2007, Supplementary Proceedings.

[Jaoua and Elloumi, 2002] Jaoua, A. and Elloumi, S. (2002). Galois connection, formal concepts and galois lattice in real relations : application in a real classifier. Journal of Systems and Software, 60(2) :149–163.

[Josuttis, 2007] Josuttis, N. M. (2007). SOA in Practice : The Art of Distributed System Design (Theory in Practice). O'Reilly Media, Inc.

[Klapaftis and Manandhar, 2005] Klapaftis, I. P. and Manandhar, S. (2005). Google and wordnet based word sense disambiguation. In Proceedings of the 22nd International Conference on Machine Learning (ICML05) Workshop on Learning and Extending Ontologies by using Machine Learning Methods, Bonn, Germany.

[Koester, 2006] Koester, B. (2006). Conceptual knowledge retrieval with fooca : Improving web search engine results with contexts and concept hierarchies. In [Perner, 2006], pages 176–190.

[Kohler et al., 2003] Kohler, J., Philippi, S., and Lange, M. (2003). SEMEDA : ontology based semantic integration of biological databases. Bioinformatics, 19(18) :2420–2427.

[Kosala and Blockeel, 2000] Kosala, R. and Blockeel, H. (2000). Web mining research : a survey. SIGKDD Exploration Newsletter, 2(1) :1–15.

[Kuznetsov, 1993] Kuznetsov, S. (1993). A fast algorithm for computing all intersections of objects in a finite semi-lattice. Automatic Documentation and Mathematical Linguistics, 27(5) :11–21.

[Kuznetsov, 2005] Kuznetsov, S. O. (2005). Galois connections in data analysis : Contributions from the soviet era and modern russian research. In [Ganter et al., 2005a], pages 196–225.

[Kuznetsov and Obiedkov, 2002] Kuznetsov, S. O. and Obiedkov, S. A. (2002). Comparing Performance of Algorithms for Generating Concept Lattices. Journal of Experimental & Theoretical Artificial Intelligence, 14 :189–216.

[Lacroix et al.,] Lacroix, Z., Parekh, K., Vidal, M.-E., Cardenas, M., and Marquez, N. BioNavigation : Selecting Optimum Paths Through Biological Resources to Evaluate Ontological Navigational Queries. In Ludäscher, B. and Raschid, L., editors, Data Integration in the Life Sciences, Second InternationalWorkshop, DILS'05, San Diego, CA, USA, July 20-22, 2005. Proceedings, volume 3615 of Lecture Notes in Computer Science, pages 275–283. Springer.

[Lakhal and Stumme, 2005] Lakhal, L. and Stumme, G. (2005). Efficient mining of association rules based on formal concept analysis. In [Ganter et al., 2005a], pages 180–195.

[Lalmas et al., 2006] Lalmas, M., MacFarlane, A., Rüger, S. M., Tombros, A., Tsikrika, T., and Yavlinsky, A., editors (2006). Advances in Information Retrieval, 28th European Conference on IR Research, ECIR 2006, London, UK, April 10-12, 2006, Proceedings, volume 3936 of Lecture Notes in Computer Science. Springer.

[Lenat, 1995] Lenat, D. B. (1995). Cyc : a large-scale investment in knowledge infrastructure. Communications of the ACM, 38(11) :33–38.

[Lichtner, 1994] Lichtner, K. (1994). An automated internet resource discovery system. In CASCON '94 : Proceedings of the 1994 conference of the Centre for Advanced Studies on Collaborative research, page 40. IBM Press.

[Lindig, 2000] Lindig, C. (2000). Fast concept analysis. In Stumme, G., editor, Working with Conceptual Structures - Contributions to ICCS 2000, Aachen, Germany.

[Lord et al., 2004] Lord, P., Bechhofer, S., Wilkinson, M. D., Schiltz, G., Gessler, D., Hull, D., Goble, C., and Stein, L. (2004). Applying semantic web services to Bioinformatics : Experiences gained, lessons learnt. In Sheila A. McIlraith, Dimitris Plexousakis, F. v. H., editor, 3rd International Semantic Web Conference ISWC'04, Hiroshima, Japan, November 7–11, 2004. Proccedings, volume 3298, pages 350–364. Springer-Verlag GmbH.

[Lukose et al., 1997] Lukose, D., Delugach, H. S., Keeler, M., Searle, L., and Sowa, J. F., editors (1997). Conceptual Structures : Fulfilling Peirce's Dream, Fifth International Conference on Conceptual Structures, ICCS '97, Seattle, Washington, USA, August 3-8, 1997, Proceedings, volume 1257 of Lecture Notes in Computer Science. Springer.

[Ma et al., 2006] Ma, J., Zhang, W.-X., and Cai, S. (2006). Variable threshold concept lattice and dependence space. In [Wang et al., 2006], pages 109–118.

[Macdonald et al., 2008] Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., and White, R. W., editors (2008). Advances in Information Retrieval, 30th European Conference on IR Research, ECIR 2008, Glasgow, UK, March 30-April 3, 2008. Proceedings, volume 4956 of Lecture Notes in Computer Science. Springer.

[Maedche and Staab, 2001] Maedche, A. and Staab, S. (2001). Ontology learning for the semantic web. IEEE Intelligent Systems, 16(2) :72–79.

[Martin and Eklund, 2008] Martin, B. and Eklund, P. W. (2008). From concepts to concept lattice : A border algorithm for making covers explicit. In [Medina and Obiedkov, 2008], pages 78–89.

[Matuszek et al., 2006] Matuszek, C., Cabral, J., Witbrock, M., and DeOliveira, J. (2006). An introduction to the syntax and content of cyc. In Proceedings of the 2006 AAAI Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering.

[McIlraith et al., 2004] McIlraith, S. A., Plexousakis, D., and van Harmelen, F., editors (2004). The Semantic Web - ISWC 2004 : Third International Semantic Web Conference,Hiroshima, Japan, November 7-11, 2004. Proceedings, volume 3298 of Lecture Notes in Computer Science. Springer.

[Medina and Obiedkov, 2008] Medina, R. and Obiedkov, S. A., editors (2008). Formal Concept Analysis, 6th International Conference, ICFCA 2008, Montreal, Canada, February 25-28, 2008, Proceedings, volume 4933 of Lecture Notes in Computer Science. Springer.

[Messai, 2004] Messai, N. (2004). Treillis de Galois et ontologies de domaine pour la classification et la recherche de sources de données génomiques. Rapport de dea informatique de lorraine, UHP-Nancy 1.

[Messai et al., 2006a] Messai, N., Devignes, M.-D., Napoli, A., and Smail-Tabbone, M. (2006a). Treillis de concepts et ontologies pour interroger l'annuaire de sources de données biologiques BioRegistry. Ingénierie des Systèmes d'Information : Systèmes d'information spécialisés, 11(1) :39–60.

[Messai et al., 2008a] Messai, N., Devignes, M.-D., Napoli, A., and Smail-Tabbone, M. (2008a). Correction et complétude d'un algorithme de recherche d'information par treillis de concepts. Revue des Nouvelles Technologies de l'Information RNTI, Classification : points de vue croisés :147 – 158. Rédacteur invités : Mohamed Nadif et François-Xavier Jollois.

[Messai et al., 2008b] Messai, N., Devignes, M.-D., Napoli, A., and Smaïl-Tabbone, M. (2008b). Extending attribute dependencies for lattice-based querying and navigation. In [Eklund and Haemmerlé, 2008], pages 189–202.

[Messai et al., 2008c] Messai, N., Devignes, M.-D., Napoli, A., and Smaïl-Tabbone, M. (2008c). Many-valued concept lattices for conceptual clustering and information retrieval. In Ghallab, M., Spyropoulos, C. D., Fakotakis, N., and Avouris, N., editors, 18th biennial European Conference on Artificial Intelligence, ECAI 2008, 21-25 July, Patras, Greece, volume 178 of Frontiers in Artificial Intelligence and Applications, pages 127–131. IOS Press.

[Messai et al., 2005] Messai, N., Devignes, M.-D., Napoli, A., and Smaïl-Tabbone, M. (2005). Querying a bioinformatic data sources registry with concept lattices. In 13th International Conference on Conceptual Structures, ICCS 05, Kassel, Germany, July 18-22, 2005, pages 323–336.

[Messai et al., 2006b] Messai, N., Devignes, M.-D., Napoli, A., and Smaïl-Tabbone, M. (2006b). BR-Explorer : An FCA-based algorithm for Information Retrieval. In Fourth Internationnal

Conference on Concept Lattices and their Applications, CLA 2006, October 30th - November 1st, Yasmine Hammamet, Tunisia, pages 285–290.

[Messai et al., 2007] Messai, N., Devignes, M.-D., Napoli, A., and Smaïl-Tabbone, M. (2007). Traitement d'attributs inter-dépendants pour la recherche d'information par treillis. In 18e Journées Francophones d'Ingénierie des Connaissances, IC 2007, Plate-forme AFIA 2007, 2 - 6 juillet, Grenoble, France, pages 109–120.

[Mimouni and Slimani, 2006] Mimouni, N. and Slimani, Y. (2006). Indexing and Searching Video Sequences Using Concept Lattices. In Fourth International Conference on Concept Lattices and their Applications - CLA'06, pages 285–290, Yasmine Hammamet, Tunisia.

[Mougin, 2006] Mougin, F. (2006). Conception d'un modèle Web sémantique appliqué à la génomique fonctionnelle. Thèse, Université Renne 1.

[Mugnier and Chein, 1998] Mugnier, M.-L. and Chein, M., editors (1998). Conceptual Structures : Theory, Tools and Applications, 6th International Conference on Conceptual Structures, ICCS '98, Montpellier, France, August 10-12, 1998, Proceedings, volume 1453 of Lecture Notes in Computer Science. Springer.

[Nauer and Toussaint, 2007] Nauer, E. and Toussaint, Y. (2007). Dynamical modification of context for an iterative and interactive information retrieval process on the web. In [Eklund et al., 2008].

[Nguifo and Njiwoua, 1998] Nguifo, E. M. and Njiwoua, P. (1998). Using Lattice-based Feamework as a Tool for Feature Extraction, chapter 13, pages 205–216. Kluwer Academic Publishers.

[Norris, 1978] Norris, E. M. (1978). An algorithm for computing the maximal rectangles in a binary relation. Revue Roumaine de Mathématiques Pures et Appliquées, 23(2) :243–250.

[Nourine and Raynaud, 1999] Nourine, L. and Raynaud, O. (1999). A fast algorithm for building lattices. Information Processing Letters, 71(5-6) :199–204.

[Oinn et al., 2004] Oinn, T., Addis, M., Ferris, J., Marvin, D., Greenwood, M., Carver, T., Matthew, Pocock, Wipat, A., and Li, P. (2004). Taverna : a tool for the composition and enactment of bioinformatics workflows. Bioinformatics, 20 :3045–3054.

[Osman, 2004] Osman, S. (2004). Réalisation d'un annuaire de sources de données génomiques en vue de la collecte et de l'intégration de données sur le web. Rapport de master professionnel sciences et techniques mention informatique, spécialité bio-informatique, Université Bordeaux I, Université Victor Segalen, Bordeaux II.

[Pensa et al., 2004] Pensa, R. G., Leschi, C., Besson, J., and Boulicaut, J.-F. (2004). Assessment of discretization techniques for relevant pattern discovery from gene expression data. In [Zaki et al., 2004], pages 24–30.

[Pernelle et al., 2002] Pernelle, N., Rousset, M.-C., Soldano, H., and Ventos, V. (2002). ZooM : a nested Galois lattices-based system for conceptual clustering. Journal of Experimental and Theoretical Artifial Intelligence (JETAI), 14(2) :157–187.

[Perner, 2006] Perner, P., editor (2006). Advances in Data Mining, Applications in Medicine, Web Mining, Marketing, Image and Signal Mining, 6th Industrial Conference on Data Mining, ICDM 2006, Leipzig, Germany, July 14-15, 2006, Proceedings, volume 4065 of Lecture Notes in Computer Science. Springer.

[Pierra, 2002] Pierra, G. (2002). Un modèle formel d'ontologie pour l'ingénierie, le commerce électronique et le web sémantique : Le modèle de dictionnaire sémantique plib. In Journées Scientifiques WEB SEMANTIQUE, Paris.

[Pisanelli et al., 2002] Pisanelli, D. M., Gangemi, A., and Steve, G. (2002). Ontologies and information systems : the marriage of the century ? In Fujita, H. and Johannesson, P., editors, New Trends in Software Methodologies, Tools and Techniques, pages 125–133. IOS Press.

[Polaillon, 1998] Polaillon, G. (1998). Organisation et interprétation par les treillis de Glois de données de type multivalué, intervalle ou histogramme. Thèse de doctorat en informatique, Université Paris IX-Dauphine.

[Prediger, 1997] Prediger, S. (1997). Logical scaling in formal concept analysis. In [Lukose et al., 1997], pages 332–341.

[Prediger, 1998] Prediger, S. (1998). Simple concept graphs : A logic approach. In [Mugnier and Chein, 1998], pages 225–239.

[Prediger, 2003] Prediger, S. (2003). Formal concept analysis for general objects. Discrete Applied Mathematics, 127(2) :337–355.

[Prediger and Stumme, 1999] Prediger, S. and Stumme, G. (1999). Theory-driven logical scaling : Conceptual information systems meet description logics. In [Franconi and Kifer, 1999], pages 46–49.

[Prediger and Wille, 1999] Prediger, S. and Wille, R. (1999). The lattice of concept graphs of a relationally scaled context. In [Tepfenhart and Cyre, 1999], pages 401–414.

[Priss, 1996] Priss, U. (1996). Relational Concept Analysis : Semantic Structures in Dictionaries and Lexical Databases. PhD thesis, Technischen Universität Darmstadt, Darmstadt, Germany.

[Priss, 1997] Priss, U. (1997). A graphical interface for document retrieval based on formal concept analysis. In Proceedings of the 8 th Midwest Artificial Intelligence and Cognitive Science Conference, AAAI.

[Priss, 2000a] Priss, U. (2000a). Comparing classification systems using facets. In Clare Beghtol, Lynne C. Howarth, N. J. W., editor, Dynamism and stability in knowledge organization ; Proceedings of the sixth international ISKO conference, Toronto Canada. Wurtzburg : Ergon Verlag.

[Priss, 2000b] Priss, U. (2000b). Faceted knowledge representation. Electronic Transactions on Artificial Intelligence, 4(C) :21–33.

[Priss, 2000c] Priss, U. (2000c). Lattice-based Information Retrieval. Knowledge Organization, 27(3) :132–142.

[Priss, 2005] Priss, U. (2005). Linguistic applications of formal concept analysis. In [Ganter et al., 2005a], pages 149–160.

[Priss, 2008] Priss, U. (2008). Facet-like structures in computer science. Axiomathes, 18(2) :243–255.

[Quan et al., 2004] Quan, T. T., Hui, S. C., Fong, A. C. M., and Cao, T. H. (2004). Automatic generation of ontology for scholarly semantic web. In [McIlraith et al., 2004], pages 726–740.

[Rada et al., 1989] Rada, R., Mili, H., Bicknell, E., and Blettner, M. (1989). Development and applications of a metric on semantic nets. IEEE Transactions on Systems, Man, and Cybernetics, 19(1) :17–30.

[Resnik, 1999] Resnik, P. (1999). Semantic similarity in a taxonomy : An information-based measure and its application to problems of ambiguity in natural language. Journal of Artificial Intelligence Research (JAIR), 11 :95–130.

[Rouane-Hacene, 2006] Rouane-Hacene, M. (2006). Étude de l'analyse formelle dans les données relationnelles : Application à la restructuration des modèles structuraux UML. Thèse en informatique, Université de Montréal, Faculté des arts et des sciences, Département d'informatique et de recherche opérationnelle.

[Rousset et al., 2002] Rousset, M.-C., Bidault, A., Froidevaux, C., Gagliardi, H., Goasdoué, F., Reynaud, C., and Safar, B. (2002). Construction de médiateurs pour intégrer des sources d'information multiples et hétérogènes : le Projet PICSEL. Revue I3 : Information - Interaction - Intelligence, 2(1) :9–59.

[Safar et al., 2004] Safar, B., Kefi, H., and Reynaud, C. (2004). OntoRefiner, a user query refinement interface usable for Semantic Web Portals. In Proceedings of Application of Semantic Web technologies to Web Communities, Workshop ECAI'04, pages 65–79, Valencia, Spain.

[Saias and Quaresma, 2002] Saias, J. and Quaresma, P. (2002). Semantic enrichment of a web legal information retrieval system. In Bench-Capon, T., Daskalopulu, A., and Winkels, R., editors, Frontiers in AI and Applications, volume 89, pages 11–20, London UK. JURIX'2002, Fifteenth Annual International Conference on Legal Knowledge and Information Systems.

[Saias and Quaresma, 2003] Saias, J. and Quaresma, P. (2003). A methodology to create ontology-based information retrieval systems. In Moura-Pires, F. and Abreu, S., editors, Progress in Artificial Intelligence, 11th Portuguese Conference on Artificial Intelligence, EPIA, 2003, volume 2902 of Lecture Notes in Computer Science, pages 424–434, Beja Portugal.

[Salton and McGill, 1986] Salton, G. and McGill, M. J. (1986). Introduction to Modern Information Retrieval. McGraw-Hill, Inc., New York, NY, USA.

[Schuler et al., 1996] Schuler, G. D., Epstein, J. A., Ohkawa, H., and Kans, J. A. (1996). Entrez : molecular biology database and retrieval system. Methods in enzymology, 266 :141–162.

[Singhal, 2008] Singhal, A. (2008). Web search : Challenges and directions. In [Macdonald et al., 2008], page 2.

[Smaïl-Tabbone et al., 2005] Smaïl-Tabbone, M., Osman, S., Messai, N., Napoli, A., and Devignes, M.-D. (2005). Bioregistry : A structured metadata repository for bioinformatic databases. In Computational Life Sciences, First International Symposium, CompLife 05, Konstanz, Germany, September 25-27, 2005, pages 46–56.

[Sowa, 1992] Sowa, J. F. (1992). Conceptual graphs. Knowledge-Based Systems, 5(3) :171–172.

[Stumme, 2004] Stumme, G. (2004). Iceberg query lattices for datalog. In [Wolff et al., 2004], pages 109–125.

[Stumme et al., 2006] Stumme, G., Hotho, A., and Berendt, B. (2006). Semantic web mining : State of the art and future directions. Web Semantics : Science, Services and Agents on the World Wide Web, 4(2) :124–143.

[Stumme et al., 2001] Stumme, G., Taouil, R., Bastide, Y., and Lakhal, L. (2001). Conceptual Clustering with Iceberg Concept Lattices. In Proceeding GI-Fachgruppentreffen Maschinelles Lernen (FGML'01), Universitat Dortmund 763.

[Stumme et al., 2000] Stumme, G., Taouil, R., Bastide, Y., Pasquier, N., and Lakhal, L. (2000). Fast computation of concept lattices using data mining techniques. In [Bouzeghoub et al., 2000], pages 129–139.

[Stumme et al., 2002] Stumme, G., Taouil, R., Bastide, Y., Pasquier, N., and Lakhal, L. (2002). Computing iceberg concept lattices with titanic. Data and Knowledge Engineering, 42(2) :189–222.

[Szathmary et al., 2008] Szathmary, L., Valtchev, P., Napoli, A., and Godin, R. (2008). Constructing iceberg lattices from frequent closures using generators. In Boulicaut, J.-F., Berthold, M. R., and Horváth, T., editors, Discovery Science, 11th International Conference, DS 2008, Budapest, Hungary, October 13-16, 2008. Proceedings, volume 5255 of Lecture Notes in Computer Science, pages 136–147. Springer.

[Tepfenhart and Cyre, 1999] Tepfenhart, W. M. and Cyre, W. R., editors (1999). Conceptual Structures : Standards and Practices, 7th International Conference on Conceptual Structures, ICCS '99, Blacksburg, Virginia, USA, July 12-15, 1999, Proceedings, volume 1640 of Lecture Notes in Computer Science. Springer.

[Tilley et al., 2005] Tilley, T., Cole, R., 0002, P. B., and Eklund, P. W. (2005). A survey of formal concept analysis support for software engineering activities. In [Ganter et al., 2005a], pages 250–271.

[Valtchev and Missaoui, 2001] Valtchev, P. and Missaoui, R. (2001). Building concept (galois) lattices from parts : Generalizing the incremental methods. pages 290–303.

[Valtchev et al., 2002] Valtchev, P., Missaoui, R., and Lebrun, P. (2002). A partition-based approach towards constructing galois (concept) lattices. Discrete Math., 256(3) :801–829.

[Ventos and Soldano, 2005] Ventos, V. and Soldano, H. (2005). Alpha galois lattices : An overview. In [Ganter and Godin, 2005], pages 299–314.

[Wang et al., 2006] Wang, L., Jiao, L., Shi, G., Li, X., and Liu, J., editors (2006). Fuzzy Systems and Knowledge Discovery, Third International Conference, FSKD 2006, Xi'an, China, September 24-28, 2006, Proceedings, volume 4223 of Lecture Notes in Computer Science. Springer.

[Widom, 1995] Widom, J. (1995). Research problems in data warehousing. In CIKM '95 : Proceedings of the fourth international conference on Information and knowledge management, pages 25–30, New York, NY, USA. ACM.

[Wille, 1982] Wille, R. (1982). Restructuring lattice theory : an approach based on hierarchies of concepts. Ordered sets, pages 445–470.

[Wille, 1984] Wille, R. (1984). Line diagrams of hierarchical concept systems. International Classification, 2 :77–86.

[Wille, 1996] Wille, R. (1996). Restructuring mathematical logic : an approach based on peirce's pragmatism. In Ursini, A. and Agliano, P., editors, Logic and algebra, pages 267–281, New York. Marcel Dekker.

[Wolff et al., 2004] Wolff, K. E., Pfeiffer, H. D., and Delugach, H. S., editors (2004). Conceptual Structures at Work : 12th International Conference on Conceptual Structures, ICCS 2004, Huntsville, AL, USA, July 19-23, 2004. Proceedings, volume 3127 of Lecture Notes in Computer Science. Springer.

[Wroe et al., 2003] Wroe, C., Stevens, R., Goble, C., Roberts, A., and Greenwood, M. (2003). A suite of DAML+OIL Ontologies to Describe Bioinformatics Web Services and Data. International Journal of Cooperative Information Systems, 12(2) :197–224.

[Yahia and Jaoua, 2001] Yahia, S. B. and Jaoua, A. (2001). Discovering knowledge from fuzzy concept lattice. pages 167–190.

[Yevtushenko, 2000] Yevtushenko, S. A. (2000). System of data analysis "Concept Explorer". In Proceedings of the 7th national conference on Artificial Intelligence KII-2000, pages 127–134, Russia.

[Yuwono et al., 1995] Yuwono, B., Lam, S. L., Ying, J. H., and Lee, D. L. (1995). A world wide web resource discovery system. In The Fourth International WWW Conference, pages 145–158.

[Zadeh, 1965] Zadeh, L. A. (1965). Fuzzy sets. Information and Control, 8(3) :338–353.

[Zaki et al., 2004] Zaki, M. J., Morishita, S., and Rigoutsos, I., editors (2004). Proceedings of the 4th ACM SIGKDD Workshop on Data Mining in Bioinformatics (BIOKDD 2004), August 22th, 2004, Seattle, Washington, USA.

[Zhou et al., 2007] Zhou, W., Liu, Z., Zhao, Y., and Xie, Z. (2007). Clustering-based reduction algorithm on the structure of fuzzy concept lattices. In 5th International Conference Formal Concept Analysis, ICFCA07, Supplementary Volume, pages 131–145, Clermont-Ferrand, France.

Annexe A

Exécution du Système BR-Explorer

A.1 Visualisation du treillis de concepts

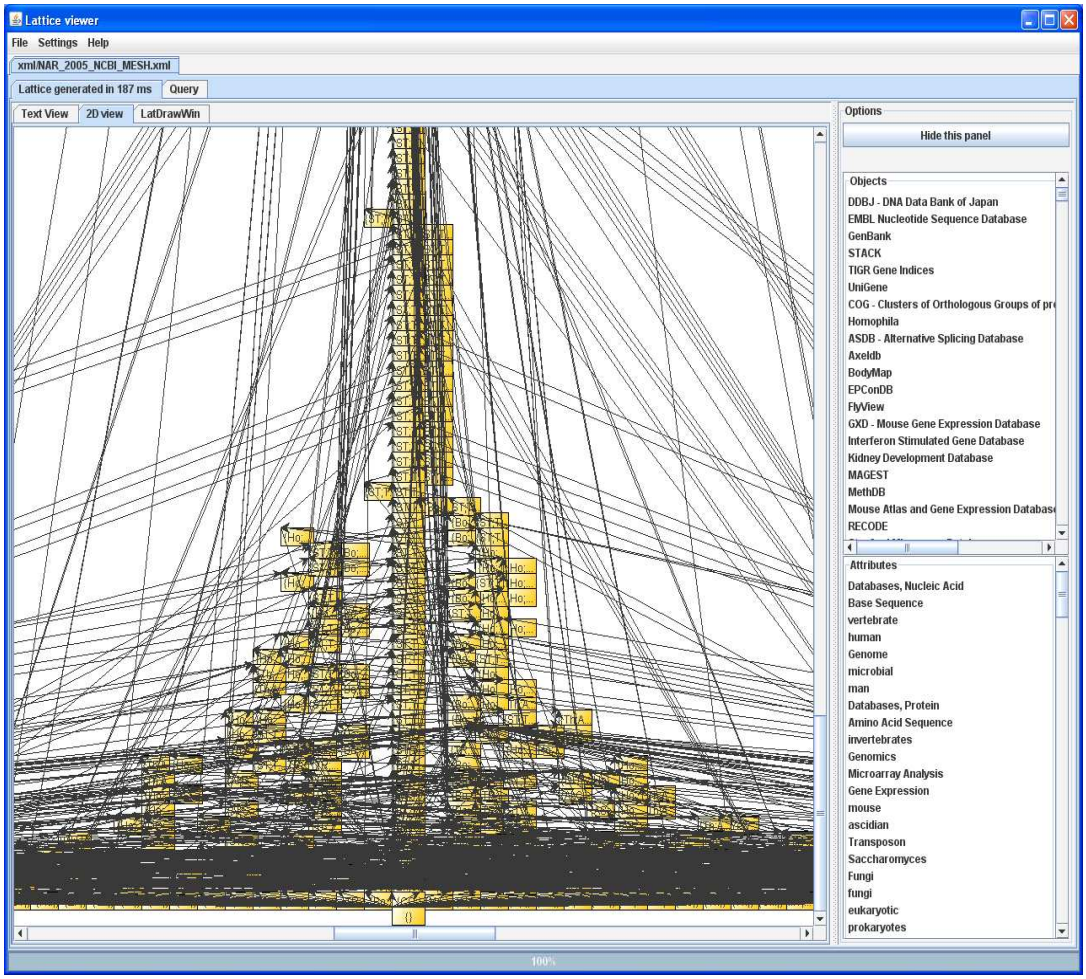


FIGURE A.1 – Visualisation du treillis de concepts correspondant au contexte formel extrait de BioRegistry.

A.2 Interface de requête de BR-Explorer

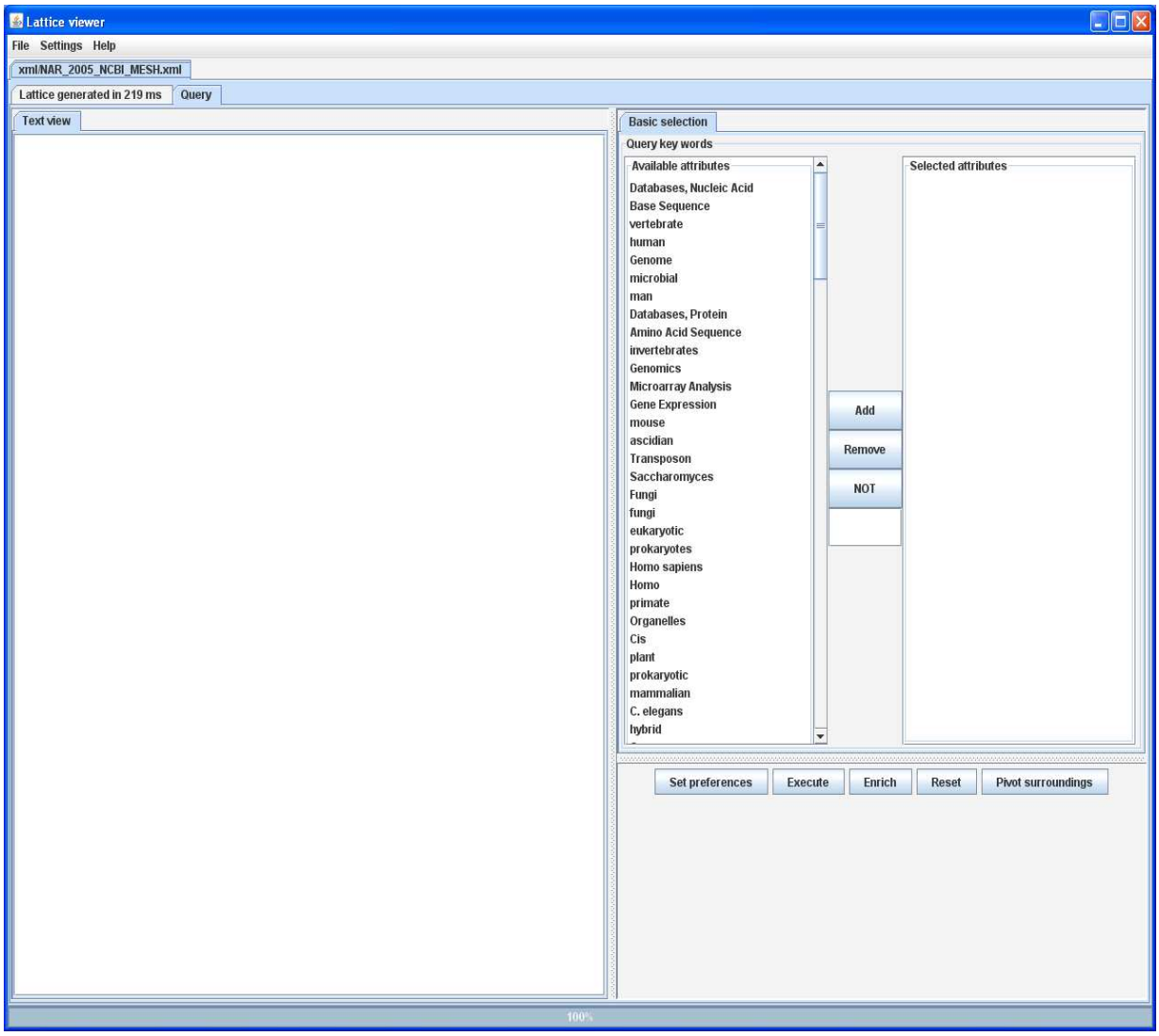


FIGURE A.2 – Interface de définition de requêtes de BR-Explorer.

A.3 Visualisation du résultat de l'exécution de BR-Explorer sur une requête

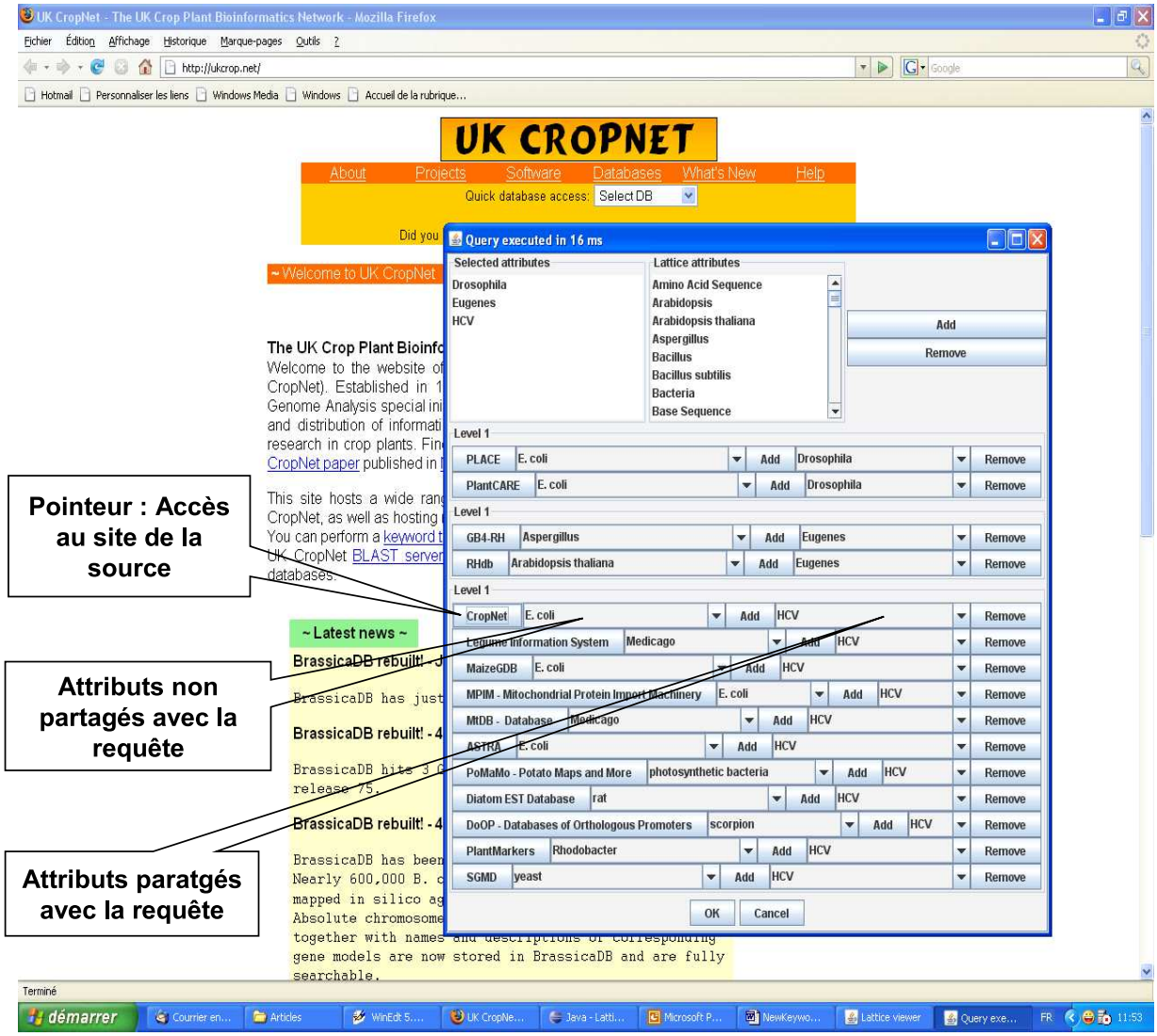


FIGURE A.3 – Visualisation du résultat de l'exécution de BR-Explorer sur la requête formée par l'ensemble de métadonnées {Drosophila, Eugenes, HCV}.

A.4 Visualisation du voisinage de la requête dans le treillis

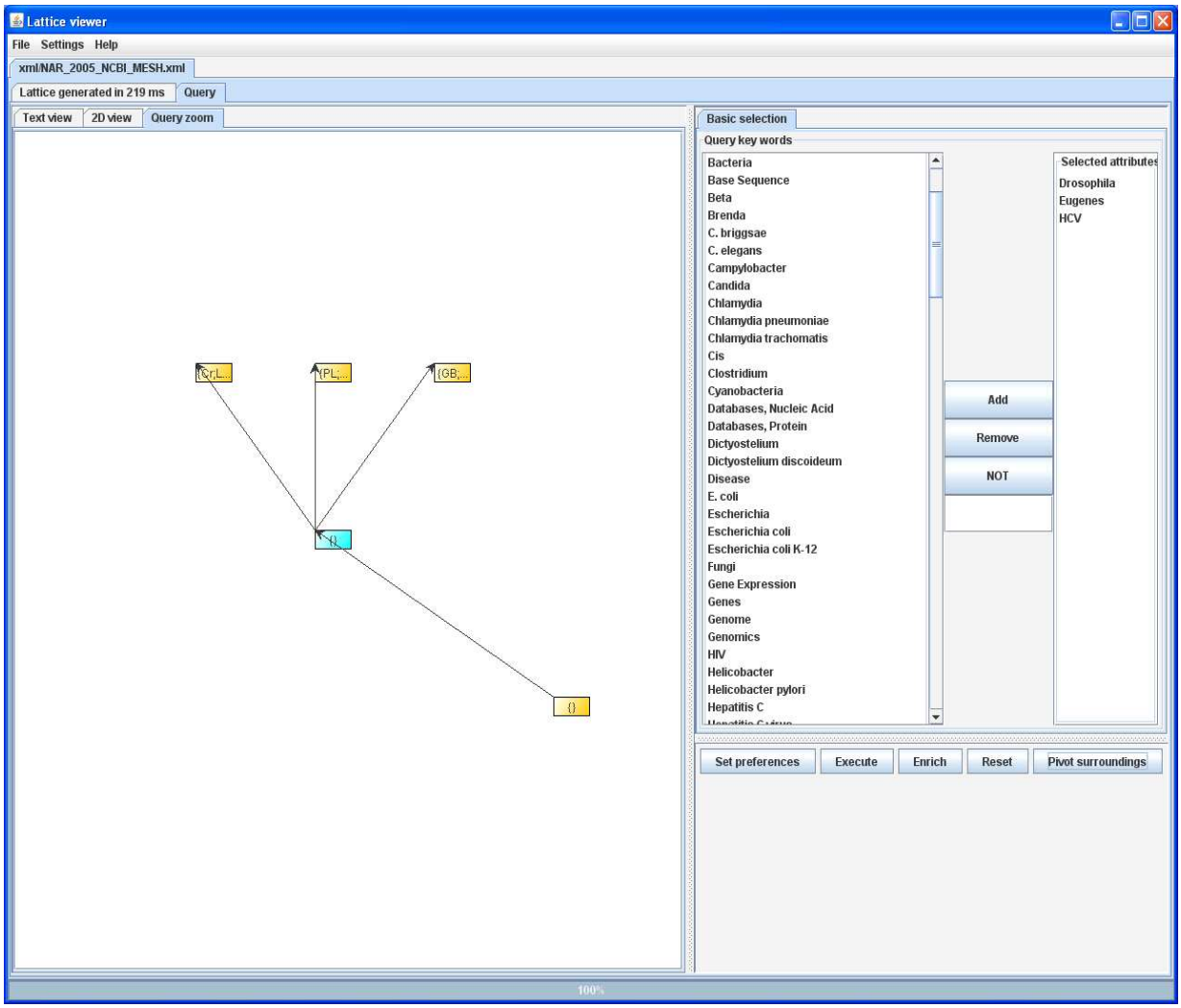


FIGURE A.4 – Visualisation du voisinage (à une distance égale à 1) de la requête formée par l'ensemble de métadonnées {Drosophila, Eugenes, HCV}.

A.5 Interface de définition de hiérarchies d'attributs dans BR-Explorer

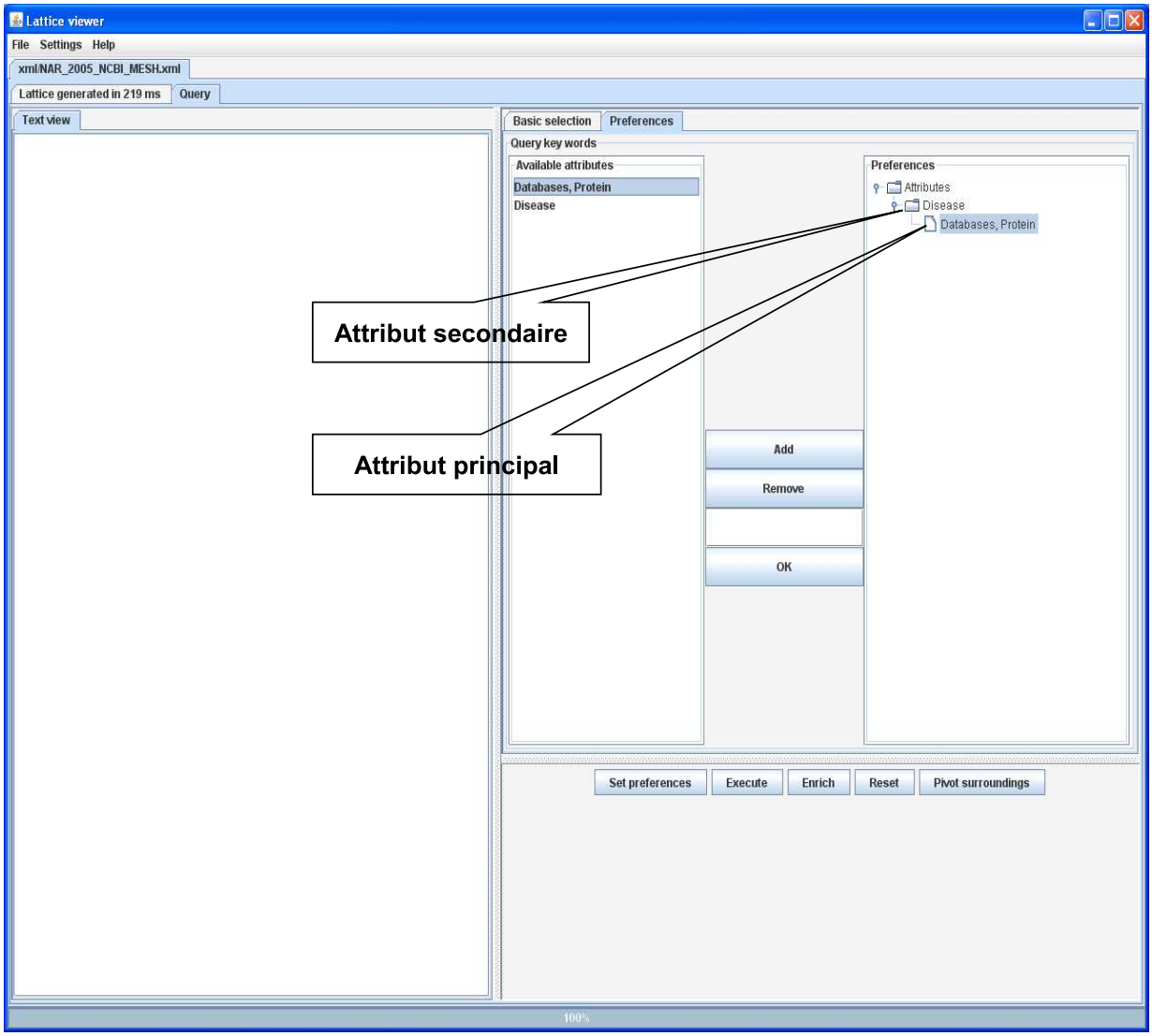


FIGURE A.5 – Définition de hiérarchies d'attributs dans BR-Explorer.

Résumé

Cette thèse porte sur l'exploitation des connaissances de domaine dans un processus de découvertes de sources de données biologiques sur le Web. Tout d'abord, des ontologies de domaine sont utilisées pour représenter un ensemble de connaissances qui reflètent le contenu et la qualité des sources de données. Ensuite, en s'appuyant sur ces connaissances, les sources sont organisées dans un treillis de concepts en fonction de leurs caractéristiques communes. Le treillis de concept constitue le support de la découverte qui peut être effectuée de deux manières différentes et complémentaires : par navigation et par interrogation. Dans les deux cas la découverte peut être guidée par des connaissances de domaines. Lors d'une découverte par navigation, les connaissances sont utilisées soit pour réduire l'espace de recherche soit pour orienter la navigation vers des concepts plutôt que d'autres. Lors d'une découverte par interrogation les connaissances de domaine sont soit exprimées sous la forme de préférences entre mots clés dans la requête soit utilisées pour l'enrichissement (ou reformulation) de la requête. Pour assurer une meilleure prise en compte des connaissances de domaine nous avons introduit les treillis de concepts multivalués. L'organisation des sources sous la forme d'un treillis de concepts multivalués permet de contrôler la taille de l'espace de recherche et d'augmenter la flexibilité et les performances du processus de découverte dans ses deux modes. La navigation peut être effectuée dans des treillis de différents niveaux de precision avec la possibilité d'effectuer des zooms dynamiques permettant le passage d'un treillis à l'autre. L'interrogation bénéficie d'une augmentation de l'expressivité dans les requêtes.

Mots-clés: Analyse de concepts formels, découverte de ressources, connaissances de domaine, ontologies, treillis de concepts multivalués, sources de données biologiques.

Abstract

This thesis deals with knowledge-based biological data sources discovery. First, domain ontologies are used for encoding metadata describing the content of biological data sources. Then the data sources are organized into a concept lattice according to their common metadata. The data source discovery process can be performed either by navigation into the obtained concept lattice or by defining queries to be inserted into the concept lattice. In both cases, domain knowledge can be used to guide the discovery. In the case of navigation, domain knowledge is used to reduce the search space and/or to guide the navigation to some concepts rather than others. In the case of querying, domain knowledge is used to express preferences between the query keywords or to refine the query. In order to take more advantage of domain knowledge, we introduce many-valued concept lattices. Several many-valued concept lattices with different levels of precision can be built from the data sources metadata set based on domain knowledge. The use of such many-valued concept lattices allows to improve the discovery process in its both forms. In the case of navigation, it is possible to consider more than one lattice and to dynamically switch form ont to another in a zooming operation. In the case of querying, more complex expressive queries can be defined and inserted into the many-valued concept lattice.

Keywords: Formal concept analysis, ressources discovery, domain knowledge, ontologies, many-valued concept lattices, biological data sources.